PROBLEM STATEMENT

Introduction

In today's competitive retail landscape, understanding customer behaviour is crucial for malls to thrive. Shopping malls cater to a diverse clientele with varying demographics, preferences, and spending habits. Utilizing advanced analytical techniques can provide valuable insights that enable targeted marketing strategies and enhance overall customer experience.

Objective

The objective of this project is to segment mall customers based on key demographic and behavioural attributes using K-Means clustering, a powerful unsupervised machine learning algorithm. By grouping customers into distinct clusters, the project aims to uncover hidden patterns in their shopping behaviour and preferences.

Scope

This project focuses on analysing customer data sourced from a public dataset containing information on customer demographics, annual income, and spending scores. The dataset consists of 200 unique entries, with no missing values or duplicates.

Deliverables

The project will deliver:

- Clearly defined customer segments with detailed profiles and characteristics.
- Visual representations of customer clusters using radar charts for easy interpretation.
- Growth strategies tailored for each customer segment aimed at increasing customer lifetime value and enhancing overall mall performance.

TABLE OF CONTENTS

FIGURES

CHAPTER - 1

INTRODUCTION

Customer understanding is vital for malls to thrive in today's competitive retail environment. Malls cater to a diverse clientele with varied shopping habits and preferences. This project utilizes K-Means clustering, an unsupervised machine learning technique, to segment mall customers into distinct groups based on five key features:

- **Customer ID:** This unique identifier will help track individual customers within their assigned segment.
- **Age:** Understanding customer age groups allows us to tailor marketing and product offerings to resonate with specific demographics.
- **Gender**: Segmenting by gender provides insights into shopping preferences that can be leveraged for targeted promotions and store layouts.
- **Annual Income:** Customer income level helps identify high-value segments and optimize product placement and promotions accordingly.
- **Spending Score:** This metric, assigned by the mall based on customer behavior, reflects spending habits and allows us to group customers with similar spending tendencies.

By segmenting customers based on these features, we can gain valuable insights into their behavior, preferences, and spending power. This information can then be used to:

- Develop targeted marketing campaigns catered to specific customer segments.
- Optimize store layout and product placement based on the preferences of different age groups, genders, and income levels.
- Design loyalty programs and promotions that resonate with specific customer spending habits.
- Improve the overall customer experience by catering to the needs of each distinct segment.

This project will employ K-Means clustering to analyze customer data and uncover hidden patterns in their shopping behavior. By grouping customers with similar characteristics, we can create distinct customer segments that can be effectively targeted by the mall's marketing and operational strategies.

CHAPTER-2

LITERATURE REVIEW

Customer segmentation is a crucial marketing strategy that allows businesses to target specific groups of consumers with tailored messaging and promotions. In the context of shopping malls, customer segmentation helps understand the diverse needs and behaviors of mall visitors, leading to improved marketing campaigns and resource allocation.

K-Means clustering, a popular unsupervised machine learning algorithm, has emerged as a valuable tool for customer segmentation in mall environments. This technique groups customers with similar characteristics into distinct clusters, enabling marketers to develop targeted strategies for each segment.

Several studies have successfully employed K-Means clustering for mall customer segmentation. For instance, Liston et al. (2018) utilize K-Means clustering on a dataset containing customer demographics and spending habits. Their findings demonstrate the effectiveness of K-Means in dividing customers into groups with distinct spending patterns [1]. Similarly, Suman et al. (2021) apply K-Means to segment customers based on annual income and spending scores, revealing valuable insights into customer behavior within the mall [2].

The key advantage of K-Means clustering lies in its simplicity and interpretability. The algorithm's ease of implementation allows marketers to gain customer insights without extensive technical expertise. Additionally, the resulting clusters provide clear profiles of distinct customer segments, facilitating the development of targeted marketing strategies.

However, it is essential to acknowledge limitations associated with K-Means clustering. The algorithm requires pre-defining the number of clusters (k), which can significantly impact the segmentation results. Techniques like the elbow method [2] can aid in determining the optimal number of clusters, but it remains a subjective decision. Furthermore, K-Means assumes spherical clusters, which might not always be the case with customer data.

In conclusion, K-Means clustering offers a valuable approach for mall customer segmentation. Its effectiveness in grouping customers with similar characteristics allows marketers to develop targeted strategies and optimize marketing campaigns.

CHAPTER-3

METHODOLOGY

- **CLUSTERING**

  Clustering is a data mining technique that groups similar data points together. Imagine sorting a basket of fruits: apples with apples, oranges with oranges. In data science, these fruits are data points, and their characteristics (features) determine how they're grouped. Clustering algorithms rely on distance metrics (like Euclidean distance) to identify similarities and group data points accordingly. This unsupervised learning technique helps uncover hidden patterns and structures within data, allowing for better understanding and segmentation. From customer segmentation in marketing to image segmentation in computer vision, clustering is a powerful tool for unlocking valuable insights from unlabeled data.

- **K-MEANS CLUSTERING**

  K-Means clustering, a popular unsupervised learning algorithm, sorts data points into pre-defined groups (clusters). Imagine sorting colored balls: reds with reds, blues with blues. It assigns each point to the nearest cluster center (centroid), minimizing the overall distance between points and their assigned centroid. Through an iterative process of reassigning points and recalculating centroids, K-Means forms distinct clusters. While simple and efficient, K-Means requires specifying the number of clusters beforehand and works best with spherical data distributions.

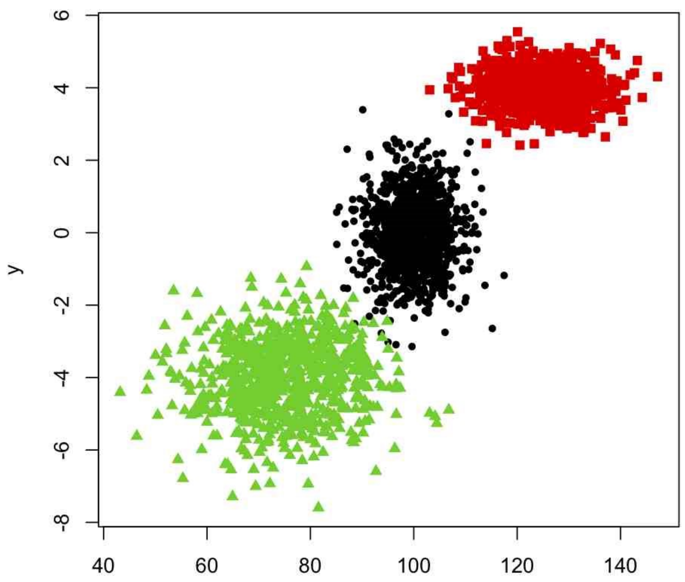


FIG-1. K-MEANS ALGORITHM

- **ELBOW METHOD**

  K-Means clustering, a popular unsupervised learning technique, sorts data points into pre-defined groups. Imagine sorting colored balls: reds with reds, blues with blues. But how many clusters (k) are ideal? Here's where the elbow method comes in as a visual aid. It plots the within-cluster sum of squares (WCSS) for different k values. WCSS represents the total distance between points and their assigned cluster center. As k increases, WCSS naturally decreases. The "elbow" on the curve indicates the point where adding more clusters yields diminishing returns. By choosing the k value at the elbow, we can avoid over-segmentation and achieve a good balance between the number of clusters and the compactness of those clusters in our mall customer segmentation project. However, the silhouette score provides a more robust measure for optimal k determination.

- **SILHOUETTE SCORE**

  The silhouette score (s) measures how well a data point fits within its assigned cluster. It considers two distances:

  1. Distance to the cluster center (centroid) of the point's assigned cluster.

  2. Average distance to the centroids of all other clusters (excluding the assigned)

  The formula for silhouette score is:

  s = (b - a) / max(a, b)

  Values closer to +1 indicate a good fit, with the point well-separated from other clusters. Scores near -1 suggest misplacement, as the point might be closer to another cluster's center. This score helps us evaluate and refine cluster assignments in data analysis.

- **RADAR CHARTS**

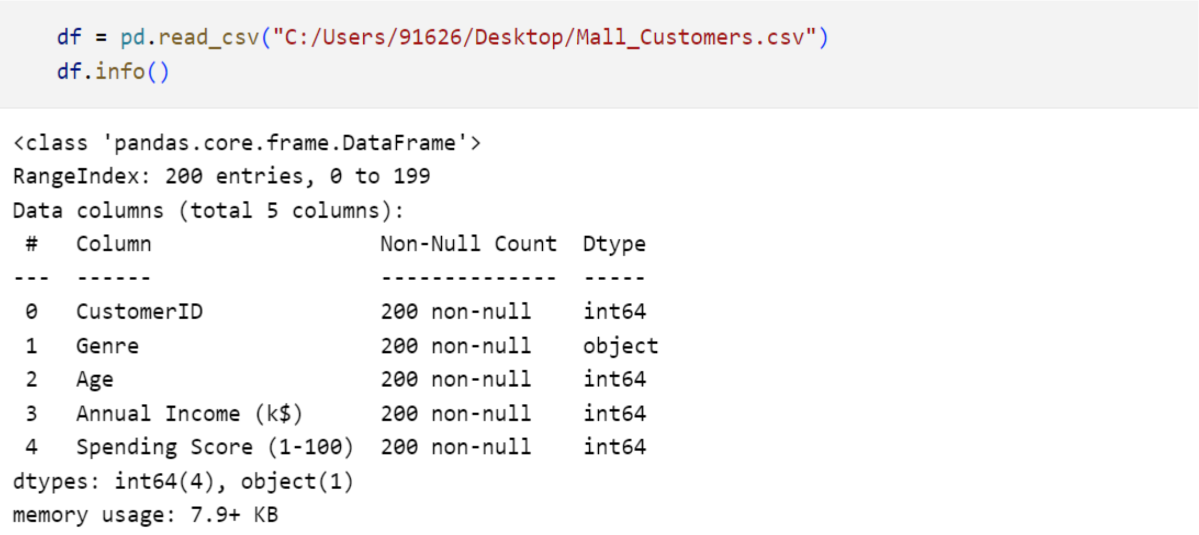
  Radar chart is a visual tool that displays data across multiple variables on axes radiating from a central point. In cluster analysis, such as with KMeans, it helps compare and understand the characteristics of different clusters. By plotting each cluster’s attributes on the chart, we can see how clusters differ or are similar, aiding in identifying unique customer segments and tailoring strategies for each.

CHAPTER-4

ANALYSIS PROCESS

1. DATA OVERVIEW

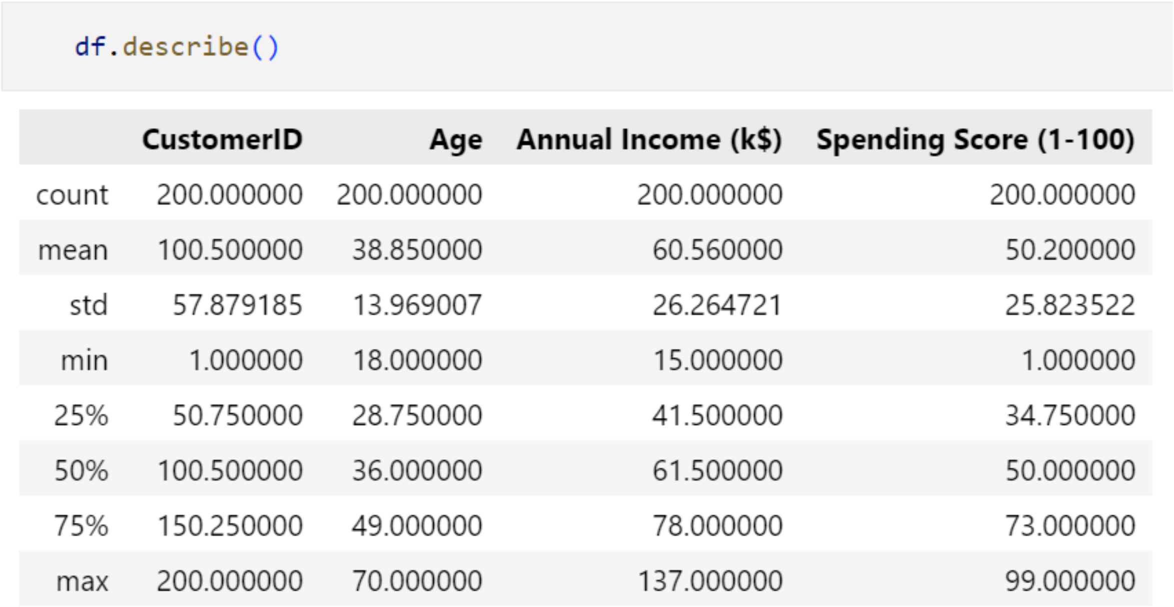
This project utilizes customer data from a public dataset link here: https://www.kaggle.com/datasets/shwetabh123/mall-customers to explore customer segmentation using K-Means clustering. Radar charts will be used to visualize the distinct characteristics of each identified customer segment. This will aid in developing targeted marketing strategies for the mall.



```
df = pd.read_csv("C:/Users/91626/Desktop/Mall_Customers.csv")
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   Genre                   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
```

This dataset has 5 columns, 200 rows, 0 duplicated rows, and 0 missing values.

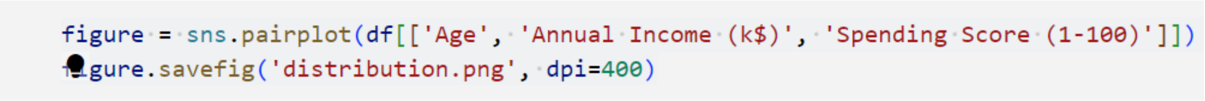


```
df.describe()
```

	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

- **Age**: The average age of customers is 38.85 years, with a standard deviation of 13.969, indicating a moderate spread around the mean age. The youngest customer is 18, and the oldest is 70 years old.
- **Annual Income (k$)**: The mean annual income is $60.56k, with a standard deviation of $26.265k, which points to a wide range of income levels among the customers. The minimum income is $15k, suggesting the presence of lower-income customers in the dataset.
- **Spending Score (1-100)**: The average spending score is 50.2, with a standard deviation of 25.824, showing a balanced distribution of spending behavior among the customers.

2. MULTIVARIATE ANALYSIS THROUGH PAIRPLOT



```
figure = sns.pairplot(df[['Age', 'Annual Income (k$)', 'Spending Score (1-100)']])
figure.savefig('distribution.png', dpi=400)
```

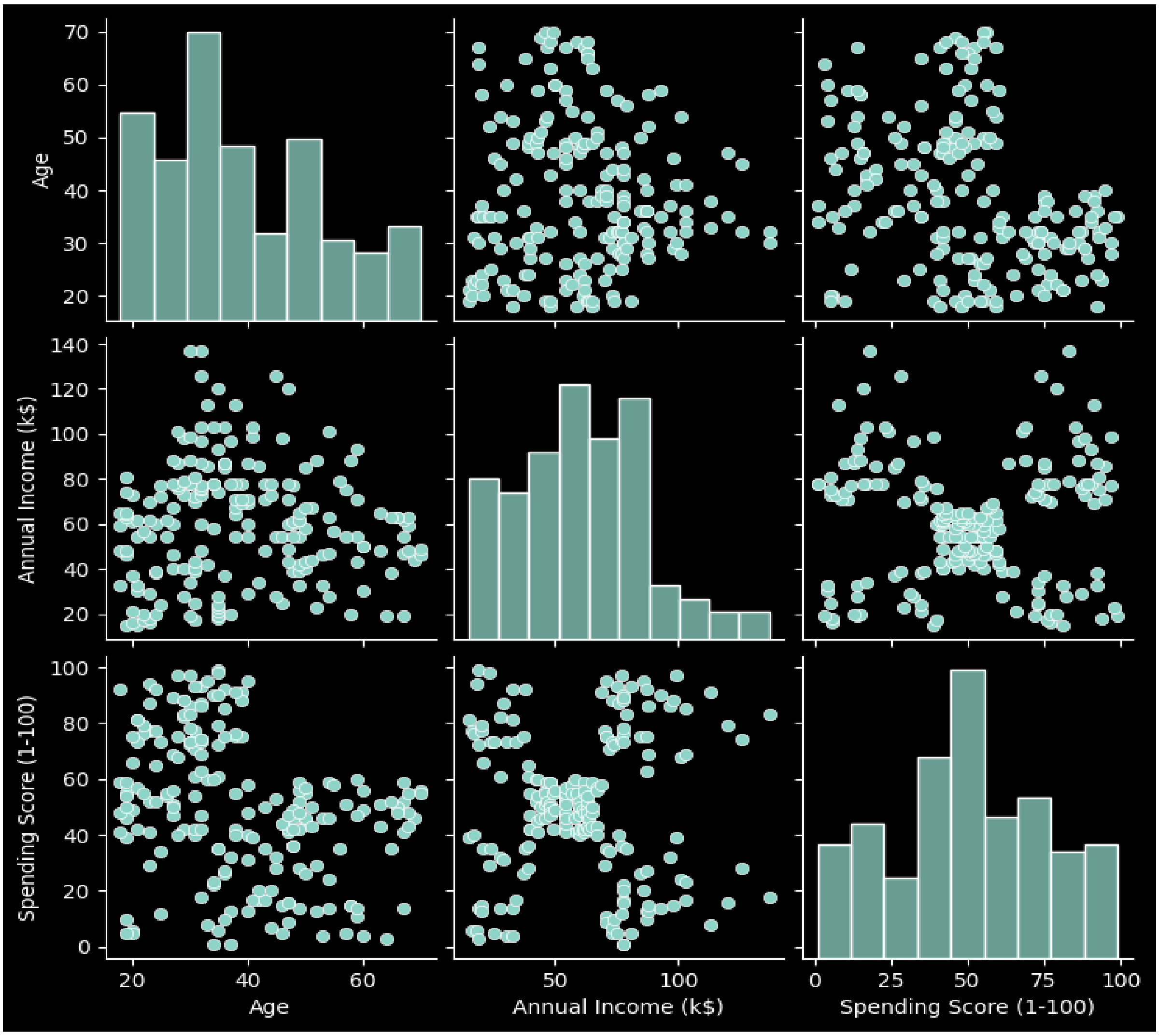


FIG-2. PAIR PLOT OF AGE, INCOME&SPENDING

Data Distribution

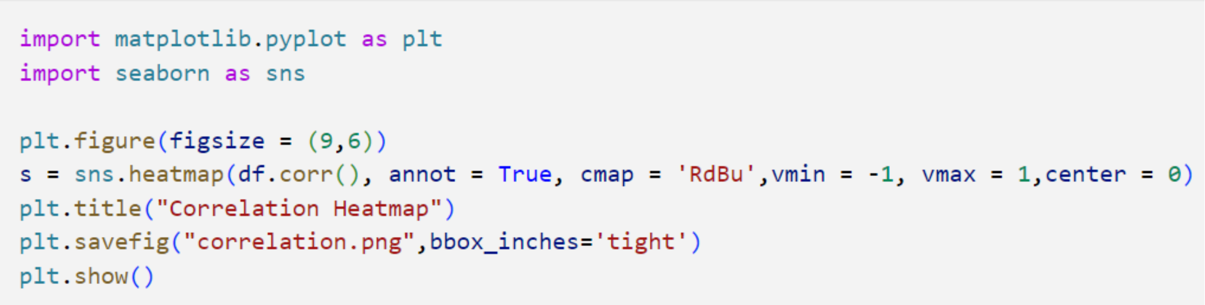
- **Age:** The right skewness in the age distribution suggests that a larger number of mall customers are younger, with fewer older customers. This skew indicate that the mall is more popular among the younger demographic, which might be due to the types of stores or entertainment options available that cater to younger preferences.
- **Annual Income:** Similar to age, the right skewness of annual income implies that there are more customers with incomes below the mean than above it. This could reflect a customer base that primarily consists of middle-income earners, with a smaller proportion of high-income earners. The mall's product and service offerings might be aligned with the spending power of this larger middle-income group.
- **Spending Score (1-100):** The near-normal distribution of the spending score indicates that customers are spread across the spectrum from low to high spenders, with most customers around the average spending score. This balanced distribution suggests that the mall attracts a wide variety of customers in terms of spending behavior.

Correlation

The scatter plots indicate a lack of strong linear relationships between the features, as the data points do not form distinct patterns or lines. This suggests that the variables, such as Age, Annual Income, and Spending Score, do not directly influence one another in a predictable manner. Such an observation is crucial for understanding the dynamics of the dataset and implies that more complex models or non-linear relationships may need to be considered for accurate customer segmentation.

3. CORRELATION ANALYSIS THROUGH HEATMAP

Correlation quantifies the linear relationship between variables, with values from -1 to 1 indicating negative to positive relationships. Heatmaps use colour gradients to represent these correlations, providing an intuitive visual summary. This combination is a powerful tool in data analysis for revealing and interpreting complex relationships efficiently.



```
import matplotlib.pyplot as plt
import seaborn as sns

plt.figure(figsize = (9,6))
s = sns.heatmap(df.corr(), annot = True, cmap = 'RdBu',vmin = -1, vmax = 1,center = 0)
plt.title("Correlation Heatmap")
plt.savefig("correlation.png",bbox_inches='tight')
plt.show()
```

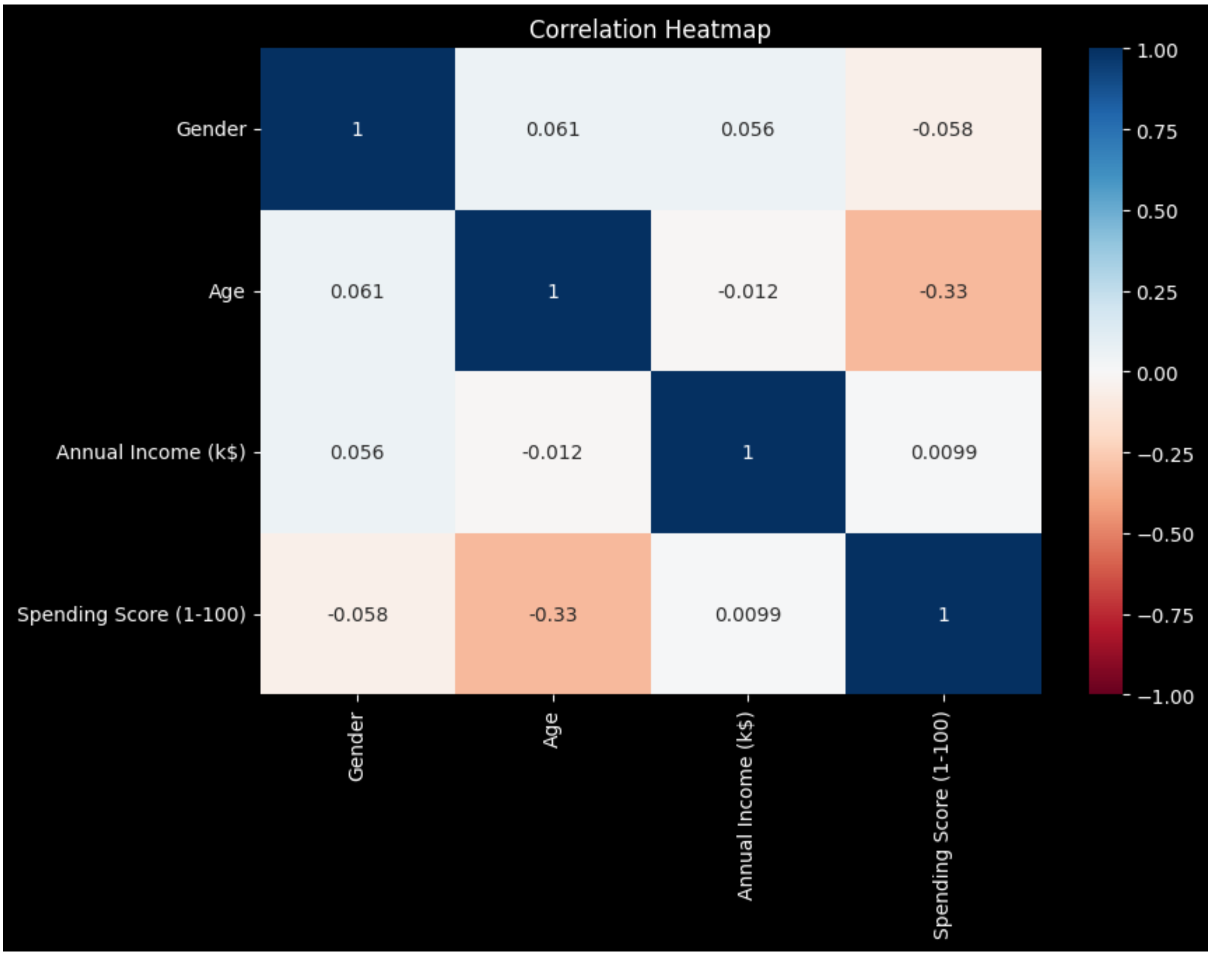


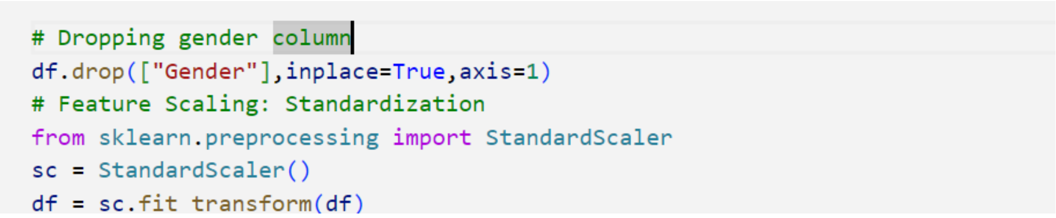
***FIG*-3. CORRELATION HEATMAP**

Interpretation

The heatmap analysis reveals a slight negative correlation of -0.33 between Spending Score and Age, suggesting that as customers get older, their spending score tends to decrease. This could imply that younger customers are more active in their purchases or more responsive to marketing strategies. While this correlation is not strong, it is the most significant among the variables presented, indicating a potential trend worth exploring for targeted customer engagement and retention strategies.

4. DATA PREPROCESSING

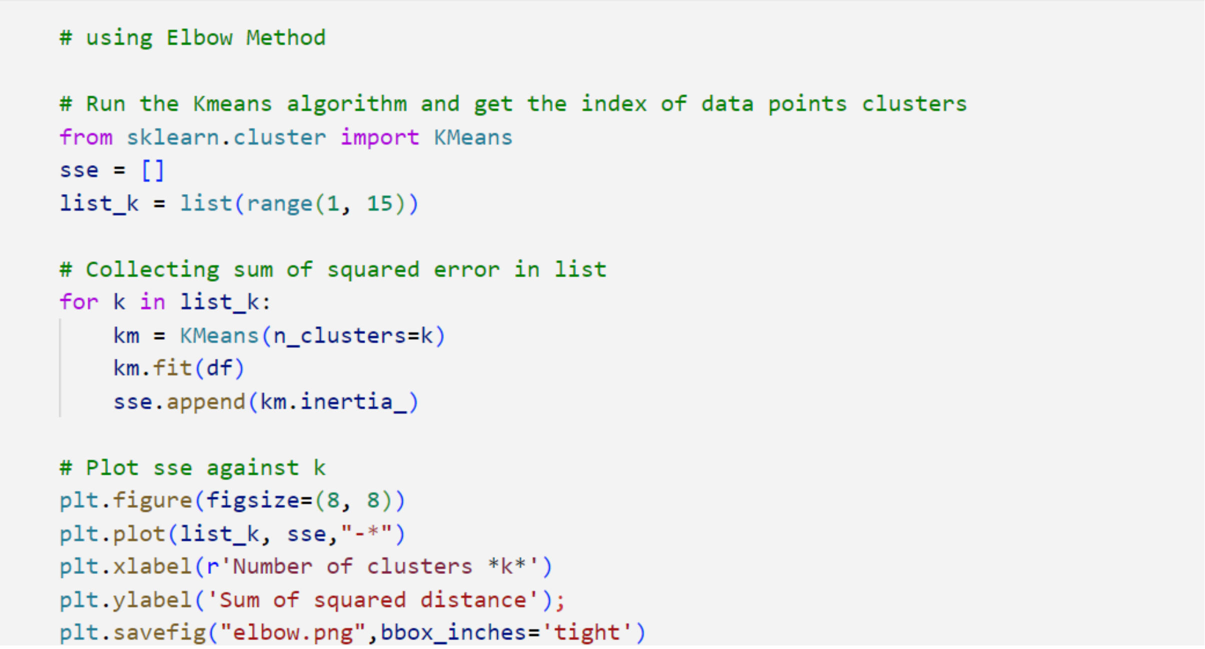
Given that the Age and Annual Income variables exhibit only minor right skewness, we are not normalizing them, next steps will be to omit the Gender variable from the analysis, as it is categorical and not continuous & standardize the data.



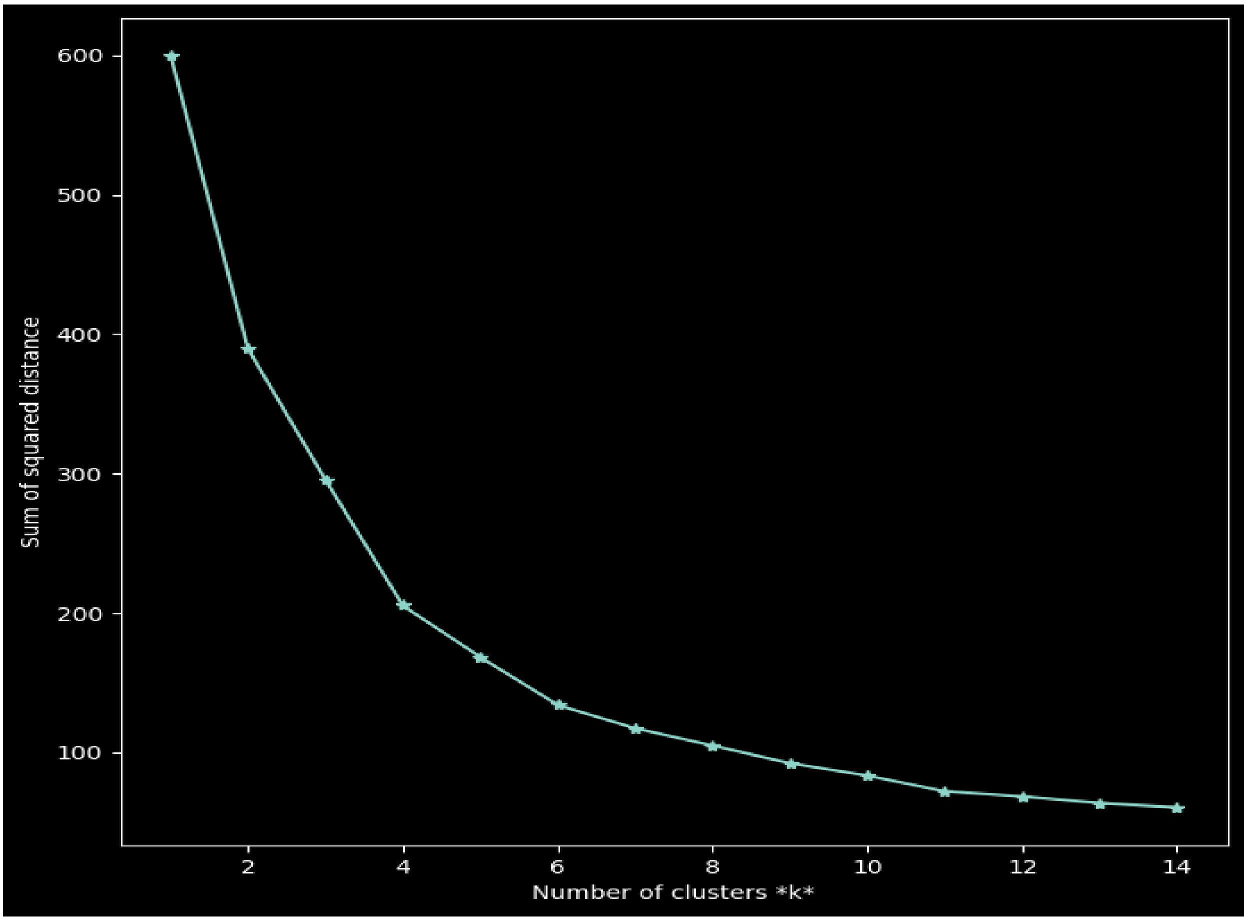
```
# Dropping gender column
df.drop(["Gender"],inplace=True,axis=1)
# Feature Scaling: Standardization
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
df = sc.fit_transform(df)
```

5. OPTIMIZING CLUSTER COUNT AND MODEL SELECTION

- **Elbow Method:** In the elbow method, we create a plot of the number of clusters on the x-axis vs. the average distance of the cluster's center to each point on the y-axis. This plot is called a scree plot. The average distance will always decrease with each additional cluster center. And, with fewer clusters, those decreases will be more substantial. At some point, adding new clusters will no longer create a substantial decrease in the average distance. This point is known as the elbow.



```
# using Elbow Method

# Run the Kmeans algorithm and get the index of data points clusters
from sklearn.cluster import KMeans
sse = []
list_k = list(range(1, 15))

# Collecting sum of squared error in list
for k in list_k:
    km = KMeans(n_clusters=k)
    km.fit(df)
    sse.append(km.inertia_)

# Plot sse against k
plt.figure(figsize=(8, 8))
plt.plot(list_k, sse,"-*")
plt.xlabel(r'Number of clusters *k*')
plt.ylabel('Sum of squared distance');
plt.savefig("elbow.png",bbox_inches='tight')
```

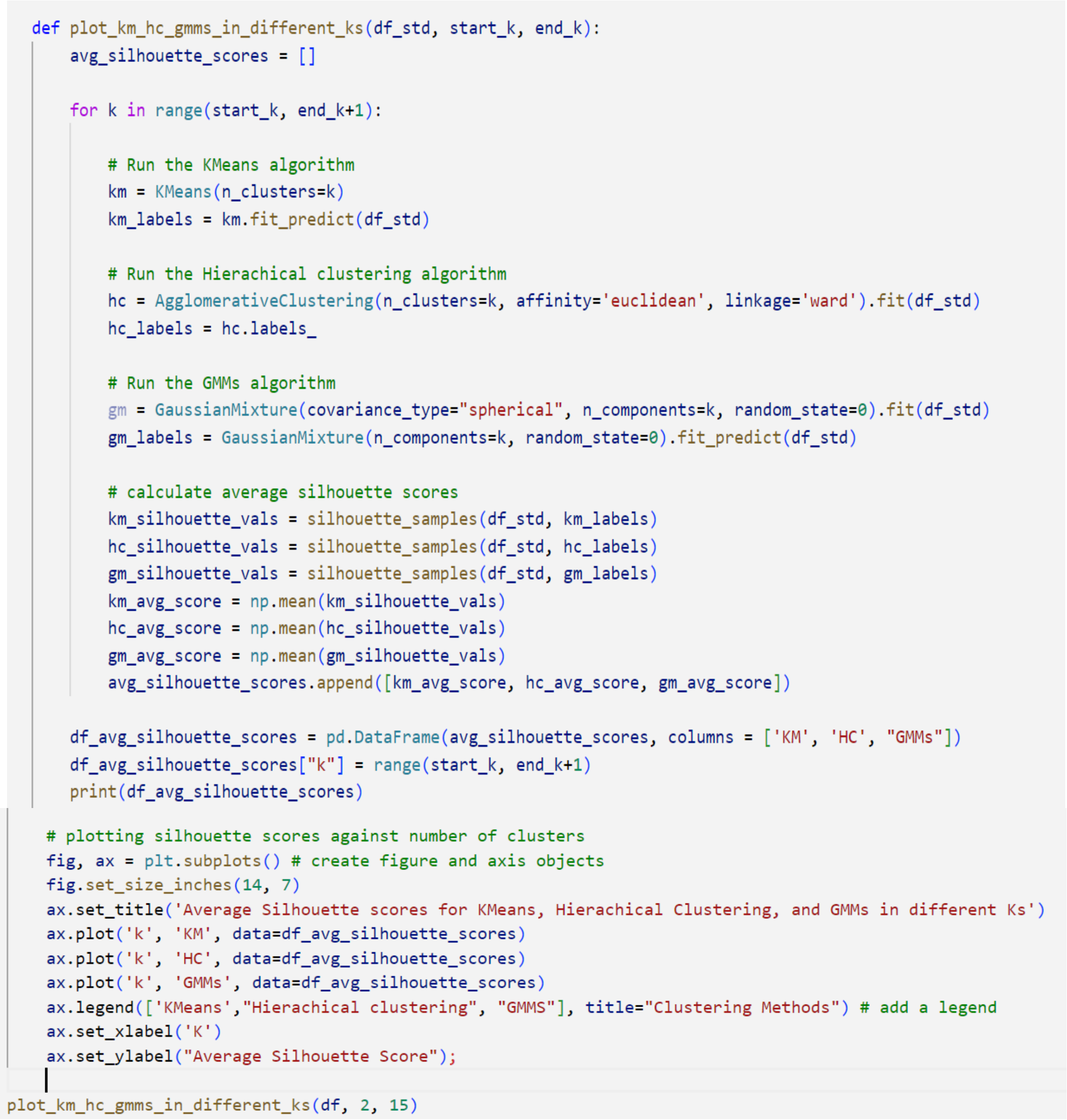


Interpretation

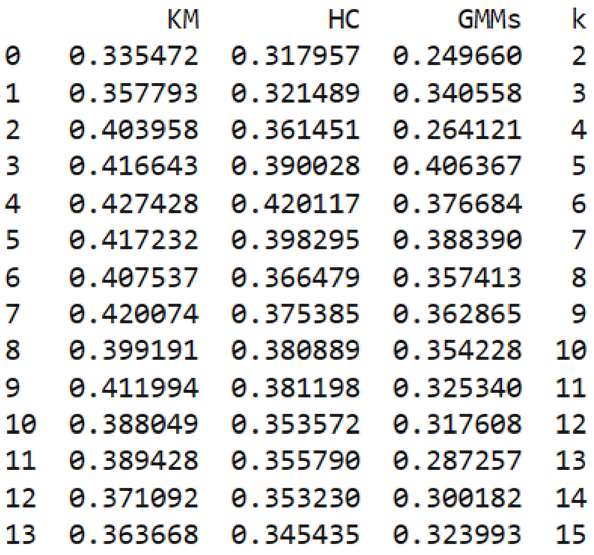
For an unclear elbow in K-means, picking K between 4 to 6 is sensible. It reflects the data's subtle clustering structure and warrants further validation with other metrics like silhouette scores.

- **Silhouette Score:** Silhouette scores will compute the average distance from all data points in the same cluster, let's say A. The average distance from all data points in the closest cluster, let's say B. Compute the coefficient, (B -A) divided by the max of a or b if a is bigger than it will be the denominator, and vice versa. And the value will be between -1 to 1. The higher the number, the better the k is.

  Silhouette Score not only can be used to select the right k but it can also be used to choose the model the performs the best.



```python
def plot_km_hc_gmms_in_different_ks(df_std, start_k, end_k):
    avg_silhouette_scores = []

    for k in range(start_k, end_k+1):

        # Run the KMeans algorithm
        km = KMeans(n_clusters=k)
        km_labels = km.fit_predict(df_std)

        # Run the Hierachical clustering algorithm
        hc = AgglomerativeClustering(n_clusters=k, affinity='euclidean', linkage='ward').fit(df_std)
        hc_labels = hc.labels_

        # Run the GMMs algorithm
        gm = GaussianMixture(covariance_type="spherical", n_components=k, random_state=0).fit(df_std)
        gm_labels = GaussianMixture(n_components=k, random_state=0).fit_predict(df_std)

        # calculate average silhouette scores
        km_silhouette_vals = silhouette_samples(df_std, km_labels)
        hc_silhouette_vals = silhouette_samples(df_std, hc_labels)
        gm_silhouette_vals = silhouette_samples(df_std, gm_labels)
        km_avg_score = np.mean(km_silhouette_vals)
        hc_avg_score = np.mean(hc_silhouette_vals)
        gm_avg_score = np.mean(gm_silhouette_vals)
        avg_silhouette_scores.append([km_avg_score, hc_avg_score, gm_avg_score])

    df_avg_silhouette_scores = pd.DataFrame(avg_silhouette_scores, columns = ['KM', 'HC', "GMMs"])
    df_avg_silhouette_scores["k"] = range(start_k, end_k+1)
    print(df_avg_silhouette_scores)

    # plotting silhouette scores against number of clusters
    fig, ax = plt.subplots() # create figure and axis objects
    fig.set_size_inches(14, 7)
    ax.set_title('Average Silhouette scores for KMeans, Hierachical Clustering, and GMMs in different Ks')
    ax.plot('k', 'KM', data=df_avg_silhouette_scores)
    ax.plot('k', 'HC', data=df_avg_silhouette_scores)
    ax.plot('k', 'GMMs', data=df_avg_silhouette_scores)
    ax.legend(['KMeans',"Hierachical clustering", "GMMS"], title="Clustering Methods") # add a legend
    ax.set_xlabel('K')
    ax.set_ylabel("Average Silhouette Score");

plot_km_hc_gmms_in_different_ks(df, 2, 15)
```



```
          KM        HC      GMMs   k
0   0.335472  0.317957  0.249660   2
1   0.357793  0.321489  0.340558   3
2   0.403958  0.361451  0.264121   4
3   0.416643  0.390028  0.406367   5
4   0.427428  0.420117  0.376684   6
5   0.417232  0.398295  0.388390   7
6   0.407537  0.366479  0.357413   8
7   0.420074  0.375385  0.362865   9
8   0.399191  0.380889  0.354228  10
9   0.411994  0.381198  0.325340  11
10  0.388049  0.353572  0.317608  12
11  0.389428  0.355790  0.287257  13
12  0.371092  0.353230  0.300182  14
13  0.363668  0.345435  0.323993  15
```

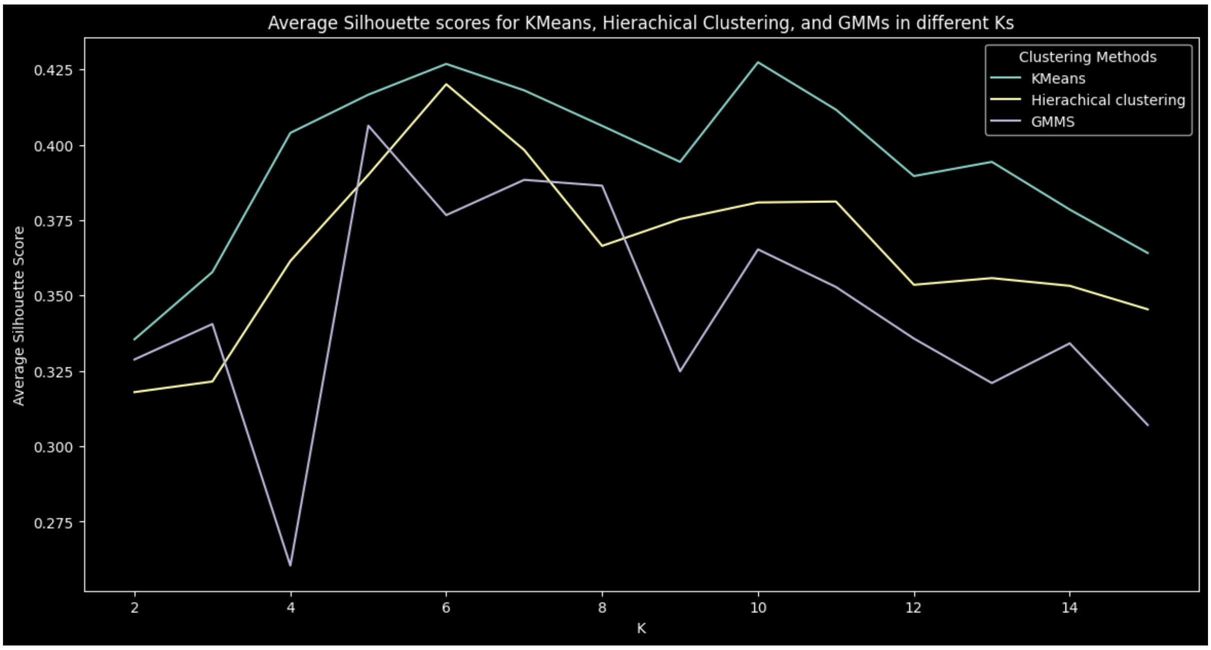


FIG-4. SILHOUETTE SCORE TABLE & CHART FOR DIFFERENT CLUSTERING MODELS

Interpretation

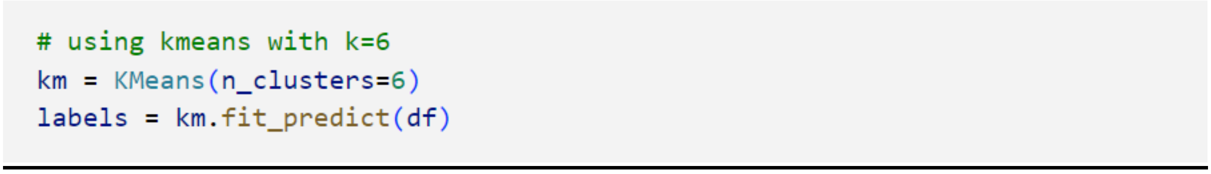
- **K-Means Clustering:** When the number of clusters (k) is set to 6, K-Means achieves a silhouette score of **0.428**. This suggests that the clusters are relatively well separated and that the data points within each cluster are closer to each other than to points in other clusters. A score closer to 1 indicates better-defined clusters.
- **Hierarchical Clustering:** Similarly, with k=6, Hierarchical clustering achieves a score of **0.42**. This is slightly lower than K-Means, indicating that while the clusters are reasonably distinct and compact, K-Means may be producing slightly tighter clusters

- **Gaussian Mixture Models (GMMs):** With k=5, GMMs achieve a score of **0.406**. This is the lowest score among the three, suggesting that the clusters formed by GMMs are less distinct compared to K-Means and Hierarchical clustering.

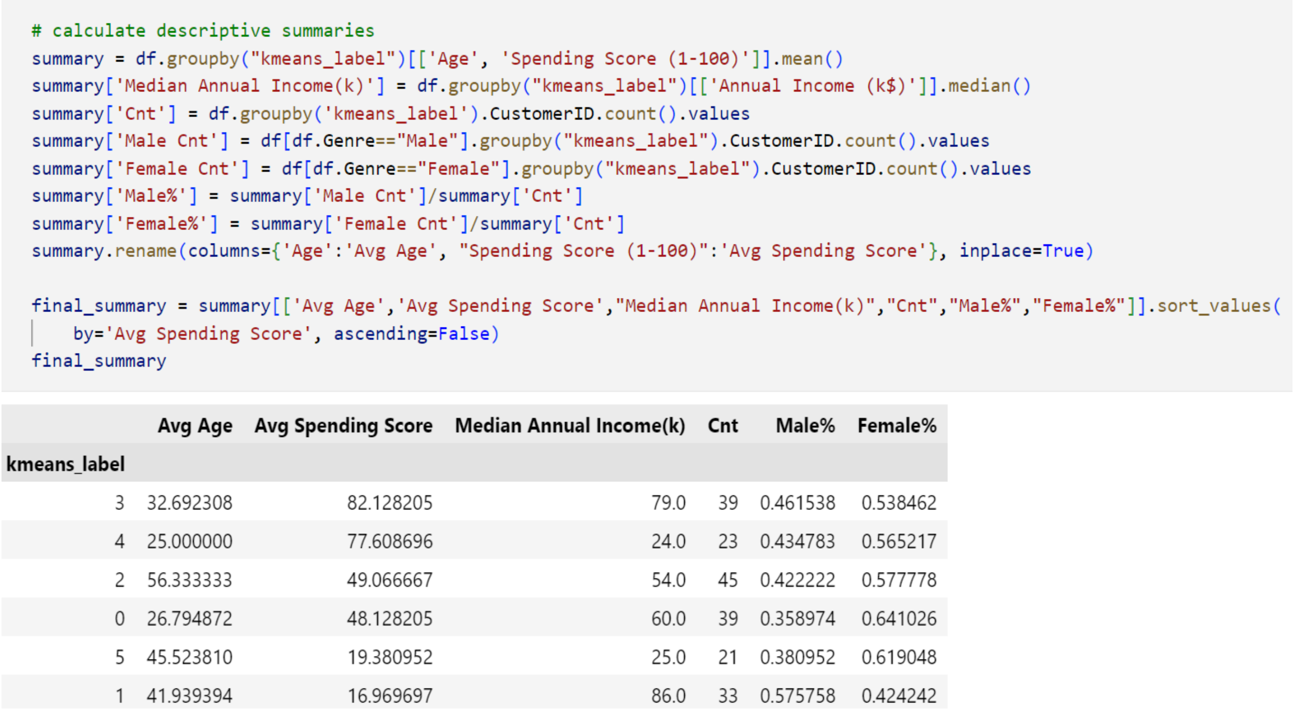
Conclusion

After evaluating the performance of K-Means, Hierarchical, and GMM clustering algorithms, **K-Means** is chosen for its highest silhouette score of **0.428**, indicating the most distinct and cohesive clusters for our dataset. This makes K-Means the optimal choice for Segmentation.

6. EXECUTING K-MEANS CLUSTERING WITH SIX CLUSTERS



```
# using kmeans with k=6
km = KMeans(n_clusters=6)
labels = km.fit_predict(df)
```



```
# calculate descriptive summaries
summary = df.groupby("kmeans_label")[['Age', 'Spending Score (1-100)']].mean()
summary['Median Annual Income(k)'] = df.groupby("kmeans_label")[['Annual Income (k$)']].median()
summary['Cnt'] = df.groupby('kmeans_label').CustomerID.count().values
summary['Male Cnt'] = df[df.Genre=="Male"].groupby("kmeans_label").CustomerID.count().values
summary['Female Cnt'] = df[df.Genre=="Female"].groupby("kmeans_label").CustomerID.count().values
summary['Male%'] = summary['Male Cnt']/summary['Cnt']
summary['Female%'] = summary['Female Cnt']/summary['Cnt']
summary.rename(columns={'Age':'Avg Age', "Spending Score (1-100)":'Avg Spending Score'}, inplace=True)

final_summary = summary[['Avg Age','Avg Spending Score',"Median Annual Income(k)","Cnt","Male%","Female%"]].sort_values(
    by='Avg Spending Score', ascending=False)
final_summary
```

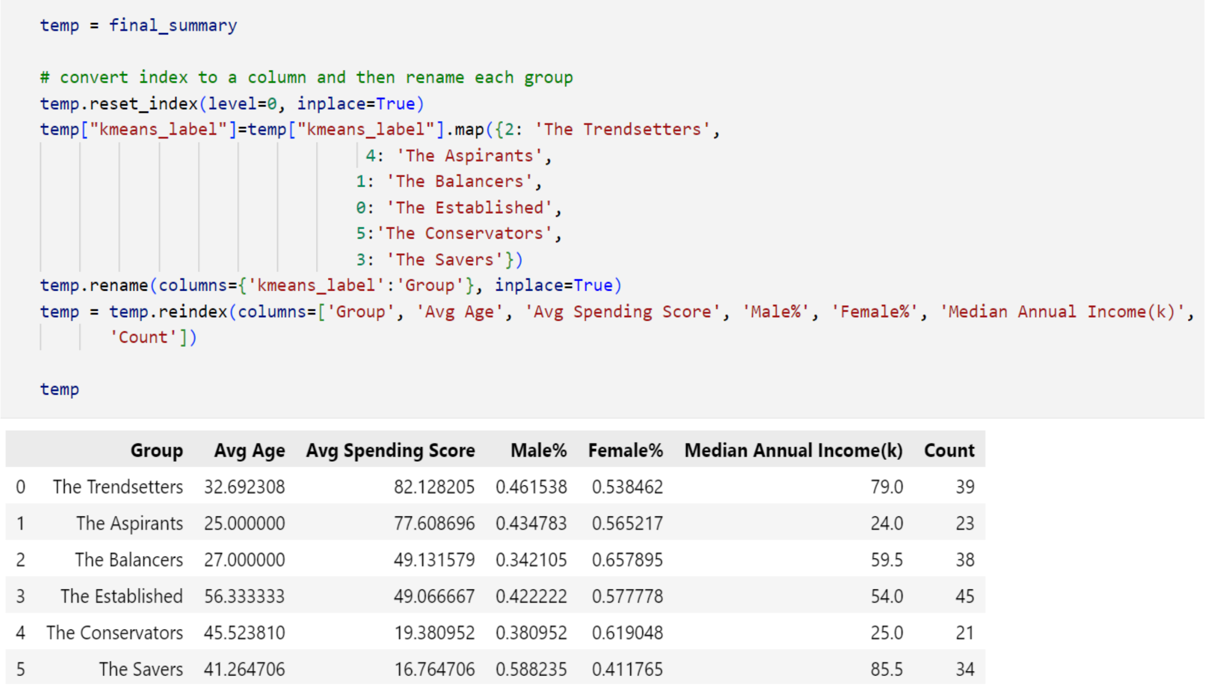
	Avg Age	Avg Spending Score	Median Annual Income(k)	Cnt	Male%	Female%
kmeans_label						
3	32.692308	82.128205	79.0	39	0.461538	0.538462
4	25.000000	77.608696	24.0	23	0.434783	0.565217
2	56.333333	49.066667	54.0	45	0.422222	0.577778
0	26.794872	48.128205	60.0	39	0.358974	0.641026
5	45.523810	19.380952	25.0	21	0.380952	0.619048
1	41.939394	16.969697	86.0	33	0.575758	0.424242

Naming the Segments

Segment names have been assigned based on their distinct characteristics :

- **Group 2- The Trendsetters**: This group is characterized by young individuals who not only have a high income but also tend to spend lavishly. They are the trendsetters, often leading the way in lifestyle and consumption.

- **Group 4 - The Aspirants:** Young in age and high in spirit, the members of this group have lower incomes but do not shy away from spending significantly. They aspire to reach higher echelons and their spending reflects their ambitions.
- **Group 0 - The Established:** With age comes wisdom and balance. This group consists of older individuals who have established themselves with moderate income and spending habits that reflect their stable and balanced lifestyle.
- **Group 1 - The Balancers:** Young and practical, this group includes individuals with middle-level incomes who spend wisely. They balance their earnings and expenditures well, ensuring a comfortable yet prudent lifestyle.
- **Group 5 - The Conservators:** Middle-aged and cautious, this group has low income and correspondingly low expenditure. They conserve their resources, focusing on essential spending and saving for the future.
- **Group 3 - The Savers:** This group is made up of middle-aged people who earn well but choose to save rather than spend. They prioritize financial security and are frugal in their spending habits.



```
temp = final_summary

# convert index to a column and then rename each group
temp.reset_index(level=0, inplace=True)
temp["kmeans_label"]=temp["kmeans_label"].map({2: 'The Trendsetters',
                                                4: 'The Aspirants',
                                               1: 'The Balancers',
                                               0: 'The Established',
                                               5:'The Conservators',
                                               3: 'The Savers'})
temp.rename(columns={'kmeans_label':'Group'}, inplace=True)
temp = temp.reindex(columns=['Group', 'Avg Age', 'Avg Spending Score', 'Male%', 'Female%', 'Median Annual Income(k)',
       'Count'])

temp
```

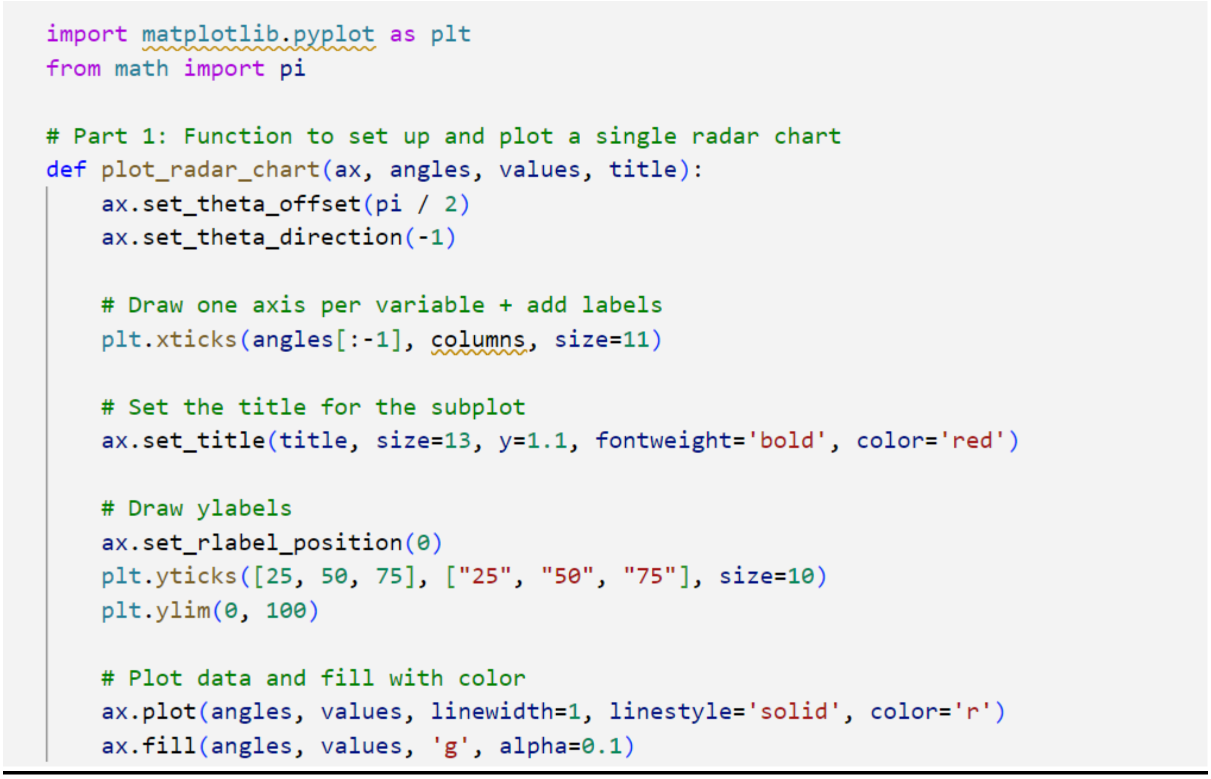
	Group	Avg Age	Avg Spending Score	Male%	Female%	Median Annual Income(k)	Count
0	The Trendsetters	32.692308	82.128205	0.461538	0.538462	79.0	39
1	The Aspirants	25.000000	77.608696	0.434783	0.565217	24.0	23
2	The Balancers	27.000000	49.131579	0.342105	0.657895	59.5	38
3	The Established	56.333333	49.066667	0.422222	0.577778	54.0	45
4	The Conservators	45.523810	19.380952	0.380952	0.619048	25.0	21
5	The Savers	41.264706	16.764706	0.588235	0.411765	85.5	34

FIG-5. DIFFERENT SEGMENTS AD THEIR CHARACTERSTICS

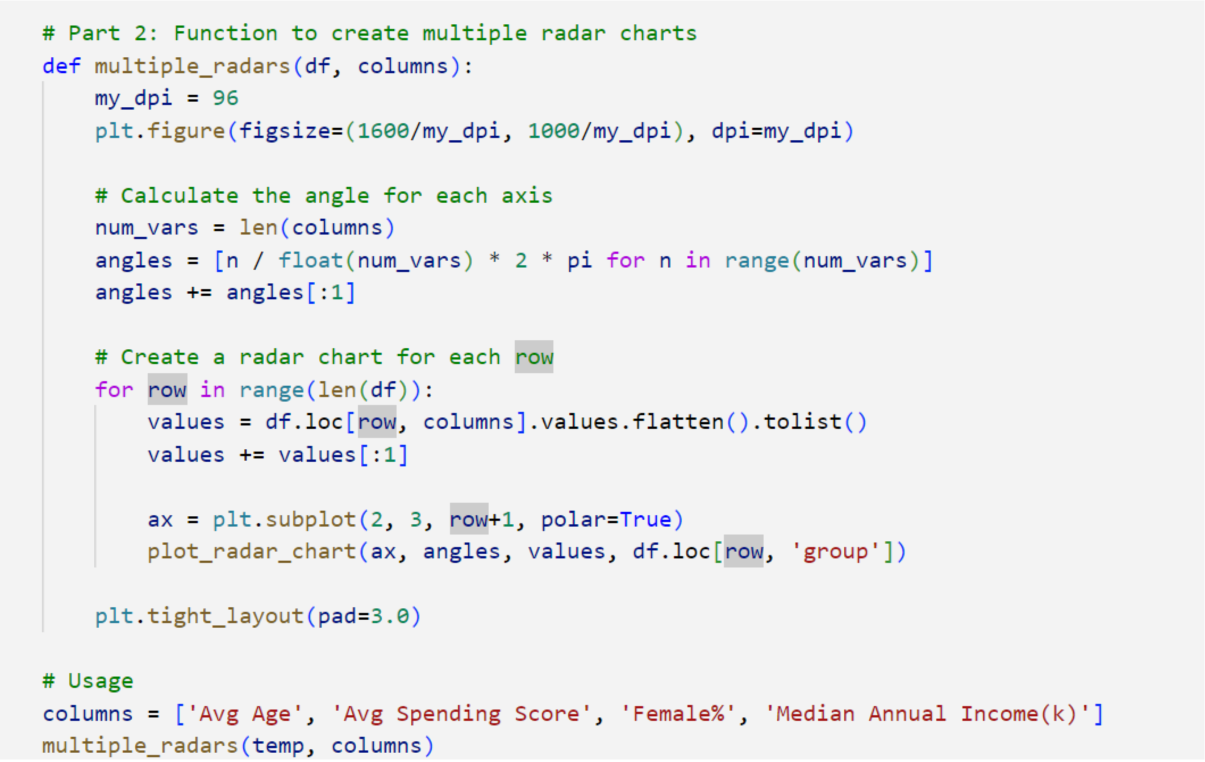
7. <u>VISUALIZING THE SEGMENTS USING RADAR CHARTS</u>

- **Radar Chart**

  In customer segmentation, a radar chart visualizes the attributes of different customer groups, allowing for easy comparison of their behaviors and preferences.



```python
import matplotlib.pyplot as plt
from math import pi

# Part 1: Function to set up and plot a single radar chart
def plot_radar_chart(ax, angles, values, title):
    ax.set_theta_offset(pi / 2)
    ax.set_theta_direction(-1)

    # Draw one axis per variable + add labels
    plt.xticks(angles[:-1], columns, size=11)

    # Set the title for the subplot
    ax.set_title(title, size=13, y=1.1, fontweight='bold', color='red')

    # Draw ylabels
    ax.set_rlabel_position(0)
    plt.yticks([25, 50, 75], ["25", "50", "75"], size=10)
    plt.ylim(0, 100)

    # Plot data and fill with color
    ax.plot(angles, values, linewidth=1, linestyle='solid', color='r')
    ax.fill(angles, values, 'g', alpha=0.1)
```



```python
# Part 2: Function to create multiple radar charts
def multiple_radars(df, columns):
    my_dpi = 96
    plt.figure(figsize=(1600/my_dpi, 1000/my_dpi), dpi=my_dpi)

    # Calculate the angle for each axis
    num_vars = len(columns)
    angles = [n / float(num_vars) * 2 * pi for n in range(num_vars)]
    angles += angles[:1]

    # Create a radar chart for each row
    for row in range(len(df)):
        values = df.loc[row, columns].values.flatten().tolist()
        values += values[:1]

        ax = plt.subplot(2, 3, row+1, polar=True)
        plot_radar_chart(ax, angles, values, df.loc[row, 'group'])

    plt.tight_layout(pad=3.0)

# Usage
columns = ['Avg Age', 'Avg Spending Score', 'Female%', 'Median Annual Income(k)']
multiple_radars(temp, columns)
```

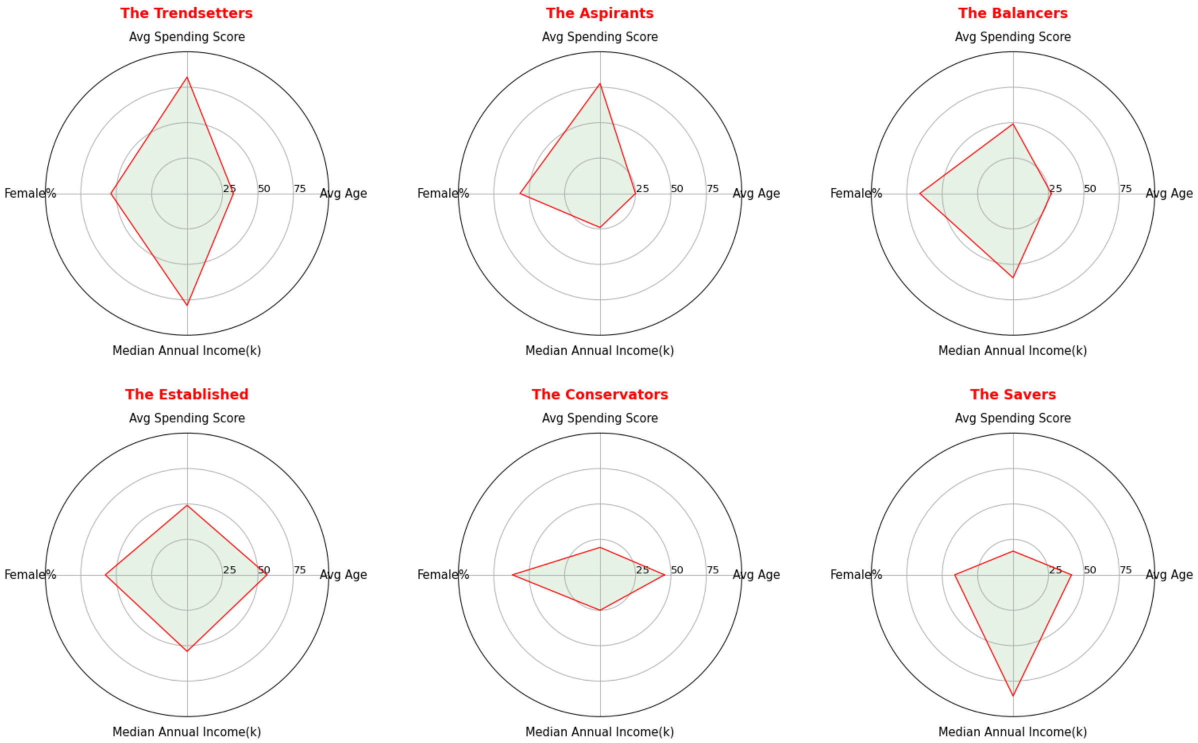


FIG-6. RADAR CHARTS REPRESENTING DIFFERENT CLUSTERS

It is clearer to see the characteristics of each group through radar charts. The next step is forming growth strategies.

8. FORMING GROWTH STRATEGIES FOR EACH CUSTOMER SEGMENT

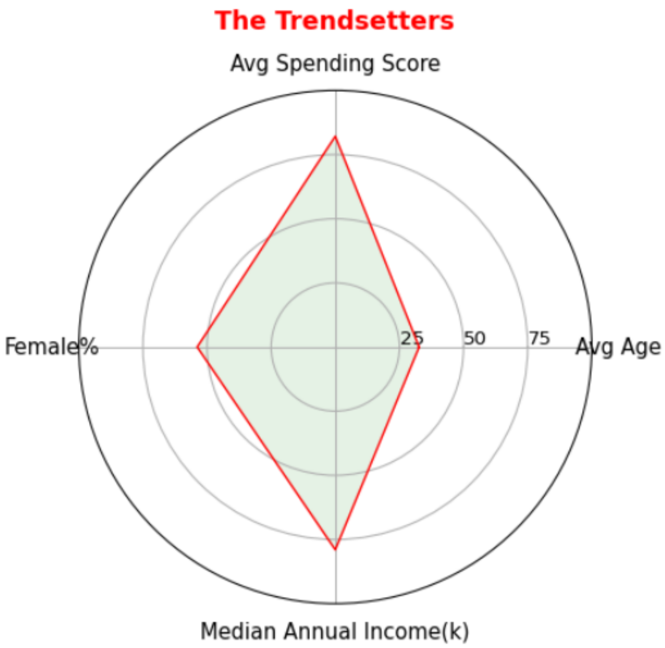
Some common growth tactics to increase customer lifetime value that can be applied to a mall business:

1. **New-customer programs**
2. **Loyalty programs**
3. **Upselling / recommending new or high-priced brands**
4. **Referral programs**
5. **Incentive program for ready-to-churn customers**
6. **Incentive program for winning back lost customers**

Group 1- The Trendsetters

Description

This group consists of rich people spending a lot on the mall. As they are very young, their income can increase even more in the future, so keeping them loyal would be the main strategy.



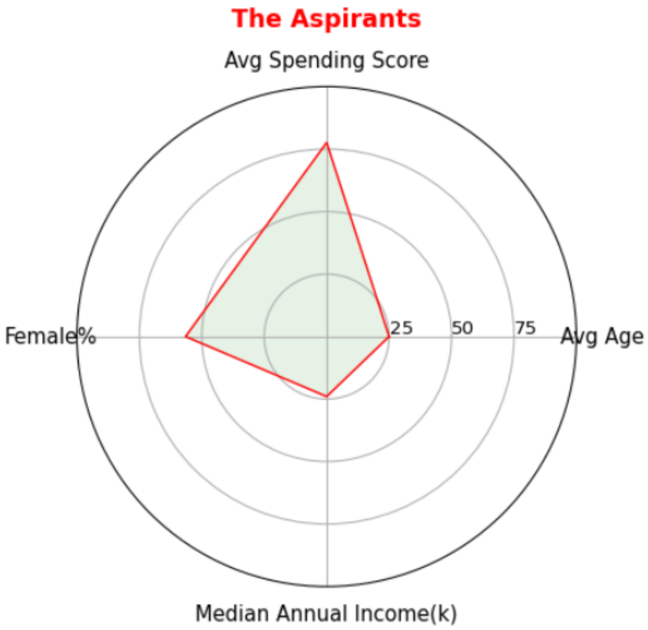
Possible Growth Strategies

1. Loyalty program
2. Up-selling
3. Referral program

As they have the ability to spend money, we can also try to sell more high-end brands or new products that they don't know they need.

Lastly, since they had spent a lot in the mall, We assume they really enjoy their shopping experience at our mall. So, asking them to refer their friends to our mall might be a good idea. Besides, their friends might also be rich and high-spend.

Group 2- The Aspirants



Description

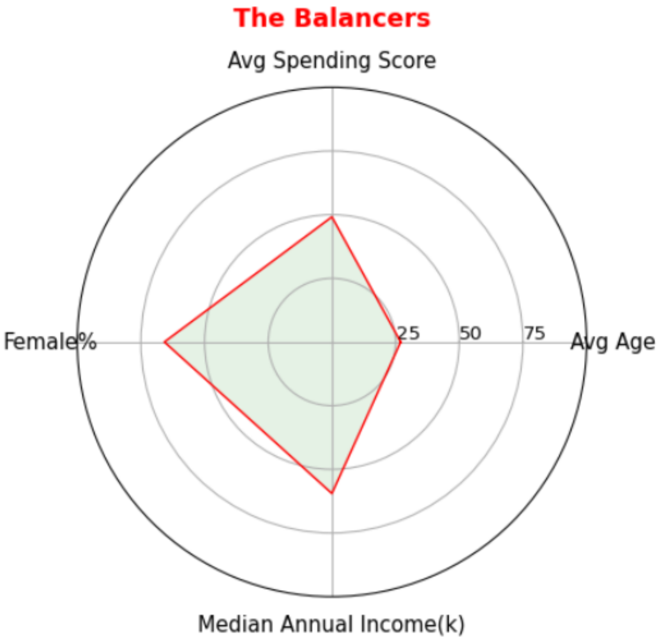
Group 2 consists of the Aspirants i.e customers with low income but have spent a lot. And they are the youngest among all groups.

Possible Growth Strategies

1. Referral program
2. Loyalty program

The main strategy for this group is a referral program, and there are three reasons for this: First, we can tell they really love our mall as they are low-income, but spent a lot on us. Second, they might value discounts more than other groups, given their financial situation. And third, they are at a young age, meaning that they are more likely to share things on social media or invite their friends to a product.

Group 3- The Balancers



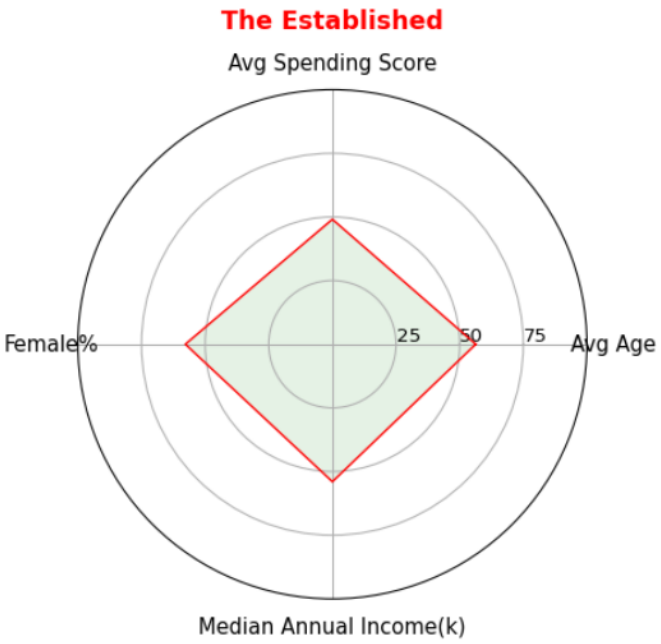
Description

Group 3 mainly consist of youngsters the average age is 27 and female customers are more, accounting for 63% of customers in this group. The main strategy is a loyalty program, but with a focus on female products.

Possible Growth Strategies

1. Loyalty program focusing on female products.

Group 4- The Established



Description

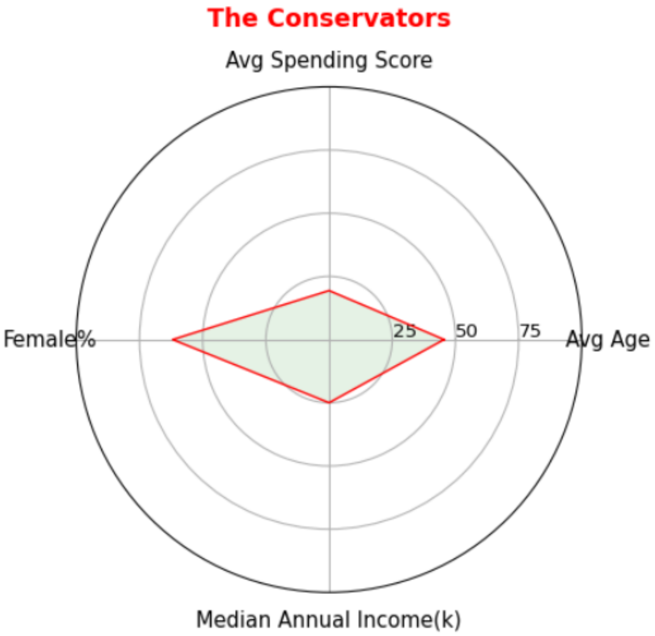
Group 4 is similar to group 3, except the average age is way higher and female customers are less. The main strategy is also a loyalty program.

Possible Growth Strategies

1. Loyalty program

Given the limited amount of information, We assume they are long-time customers already, so their purchasing behavior won’t be changed that much in the future, so keeping them loyal is this group's main strategy.

Group 5- The Conservators



Description

Group 5 is the least desirable group as they are low-spending and low-income.

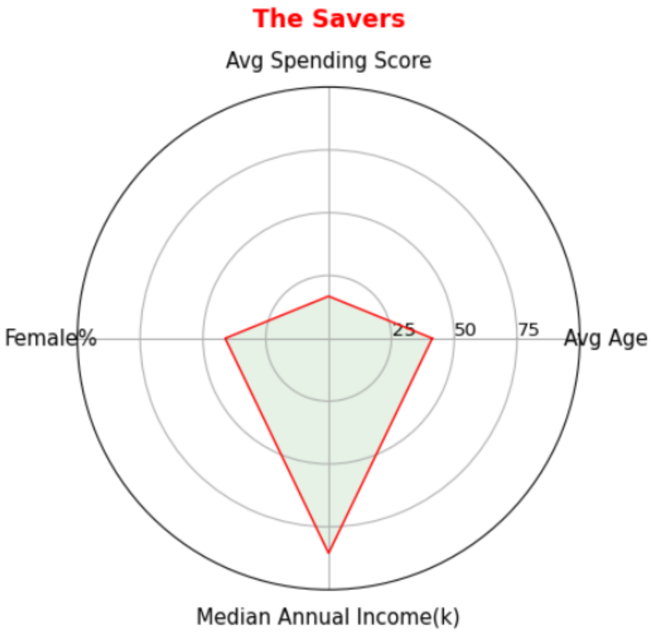
Possible Growth Strategies

None

Further Exploration Direction

Investigating why they spend so little, is it because they're low-income, staying single, or we face new competitors or other reasons?

Group 6- The Savers



Description

Group 6 is a high-income but low-spending group with a male ratio of 58%.

Possible Growth Strategies

1. New-customer program
2. Incentive program for winning back lost customers
3. Recommending high-pricing products

Possible Reasons for Spending Low:

1. They are just new customers
2. They are **lost** customers
3. They mostly shop at a more high-end mall. The reason is worth further investigation.

CHAPTER-5

CONSOLIDATED STRATEGY OUTLOOK

Loyalty Initiative: Targeting Groups 1, 2, 3, and 4

To bolster customer fidelity, it's advisable for the mall to focus on Groups 1 through 4. A tiered loyalty scheme, with 3 to 5 membership levels, could be implemented where higher tiers offer greater perks and discounts, determined by the customer's spending over time.

Such a program is expected to encourage patrons in these segments to either maintain their current membership status or aspire to higher tiers, ensuring a steady revenue stream for the mall and solidifying their allegiance.

Referral Incentive: Concentrating on Groups 1 and 2

Groups 1 and 2, characterized by their higher expenditure and youthful demographic, are ideal for a referral initiative. They are more likely to introduce peers to the mall, expanding the customer base. A structured referral strategy can be developed, potentially guided by a detailed article on the subject, offering rewards like discounts or complimentary products for successful referrals.

Upselling Strategy: Focusing on Groups 1 and 6

Groups 1 and 6, known for their substantial income, are prime candidates for upselling. Communication through emails or texts, after analyzing their purchase patterns, can be effective. Further segmentation can identify subgroups for targeted promotions, offering premium versions of favored products at a discount.

Engagement Program for New and Inactive Customers: Group 6

Group 6 presents a unique opportunity; despite their high earnings, their spending is minimal. Delving into the reasons behind their limited expenditure can reveal strategies to enhance the Gross Merchandise Value (GMV). For those who have lapsed, re-engagement efforts via emails or texts with discounts on previously purchased items could reignite their interest.

For newcomers, an introductory program that quickly acquaints them with the mall's value proposition is crucial. Marketing popular items at discounted rates or temporarily granting them top-tier membership benefits could swiftly integrate them into the mall's ecosystem, experiencing firsthand the advantages of being a valued customer.

CHAPTER – 6

REFERENCES

1. Customer Segmentation in Online Retail Using K-Means Clustering Classification and Principal Component Biplot | SpringerLink
2. Customer Segmentation Using K-Means Clustering | SpringerLink
3. Mall Customer Segmentation Data (kaggle.com)
4. https://www.kaggle.com/code/abhishekyadav5/kmeans-clustering-with-elbow-method-and-silhouette
5. https://scikit-learn.org/stable/index.html
6. https://matplotlib.org/
7. https://medium.com/data-and-beyond/customer-segmentation-using-k-means-clustering-with-pyspark-unveiling-insights-for-business-8c729f110fab
8. https://www.kaggle.com/code/heeraldedhia/hierarchical-clustering-for-customer-data
9. Copilot (microsoft.com)
10. Gemini (google.com)