GRT INSTITUTE OF

ENGINEERING AND

TECHNOLOGY, TIRUTTANI - 631209

Approved by AICTE, New Delhi Affiliated to Anna University, Chennai

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

PHASE 4

PROJECT TITLE

Customer Segmentation Using Data Science

COLLEGE CODE : 1103

Lokeshwar.k

3rd yr, 5th sem

Reg no. : 110321104027

lokeshwarnaidu2@gmail.com

PHASE 4: CUSTOMER SEGMENTATION USING DATASCIENCE

4.1 IN THIS TECHNOLOGY YOU WILL CONTINUE BUILDING YOUR PROJECT BY PREPROCESSING YOUR DATASET

4.1.1 PREPROCESS DATASET

1. IMPORT LIBRARIES:

• First, import the necessary libraries, including Pandas for data manipulation.

import pandas as pd

2. LOAD THE DATASET:

• Load the dataset from the CSV file. Make sure to download the dataset from Kaggle and place it in your working directory.

3. EXPLORE THE DATASET:

• Explore the dataset to understand its structure, check for missing values, and review data types.

4. HANDLE MISSING VALUES:

• In this dataset, it's possible that there are no missing values. However, if there were any missing values, you'd need to decide how to handle them. Options include dropping rows with missing values, filling them with a default value, or using more advanced imputation techniques.

5. ENCODE CATEGORICAL DATA (IF ANY):

• The Mall Customers dataset doesn't contain categorical variables that need encoding. However, if your dataset had categorical data (e.g., "Genre"), you'd need to encode it, typically using one-hot encoding or label encoding.

6. FEATURE SELECTION:

• Depending on your analysis goals, you may want to select a subset of features for segmentation.

7. STANDARDIZE/NORMALIZE DATA :

• If you're using clustering algorithms that rely on distances (e.g., K-Means), it's often a good practice to standardize or normalize the data to bring features to the same scale. This can be done using techniques like Min-Max scaling or Z-score standardization.

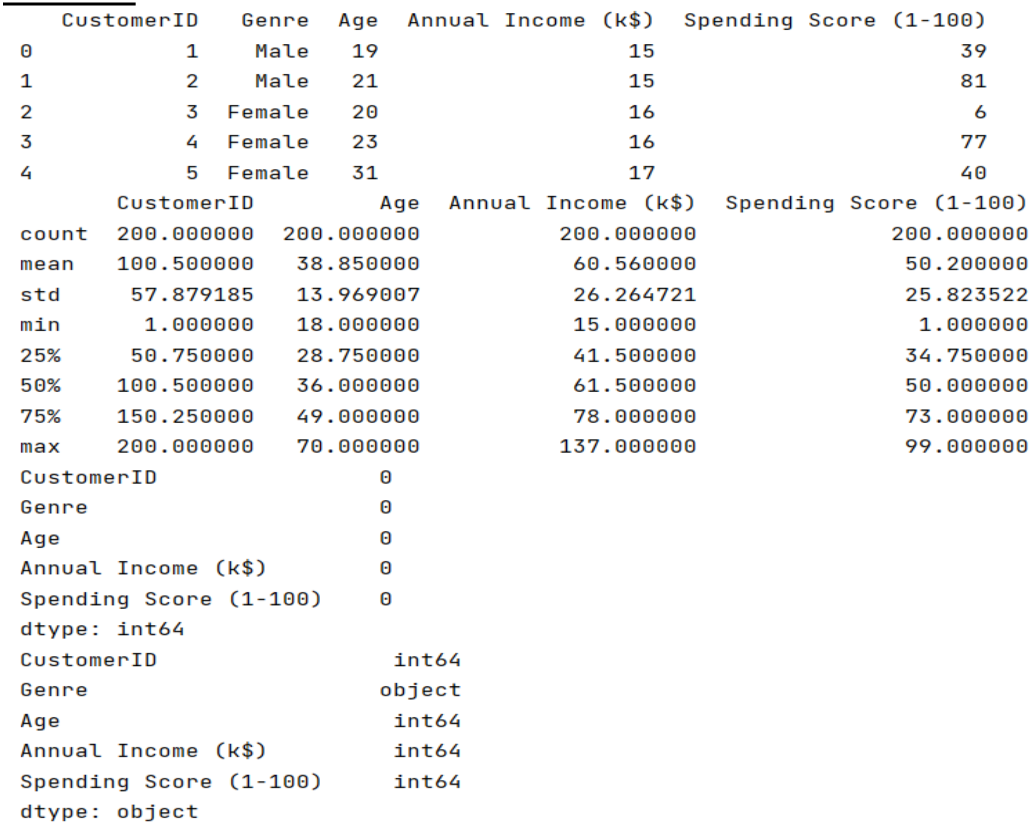
8. SAVE THE PREPROCESSED DATA :

- If you want to save the preprocessed data for future use, you can save it to a new CSV file.

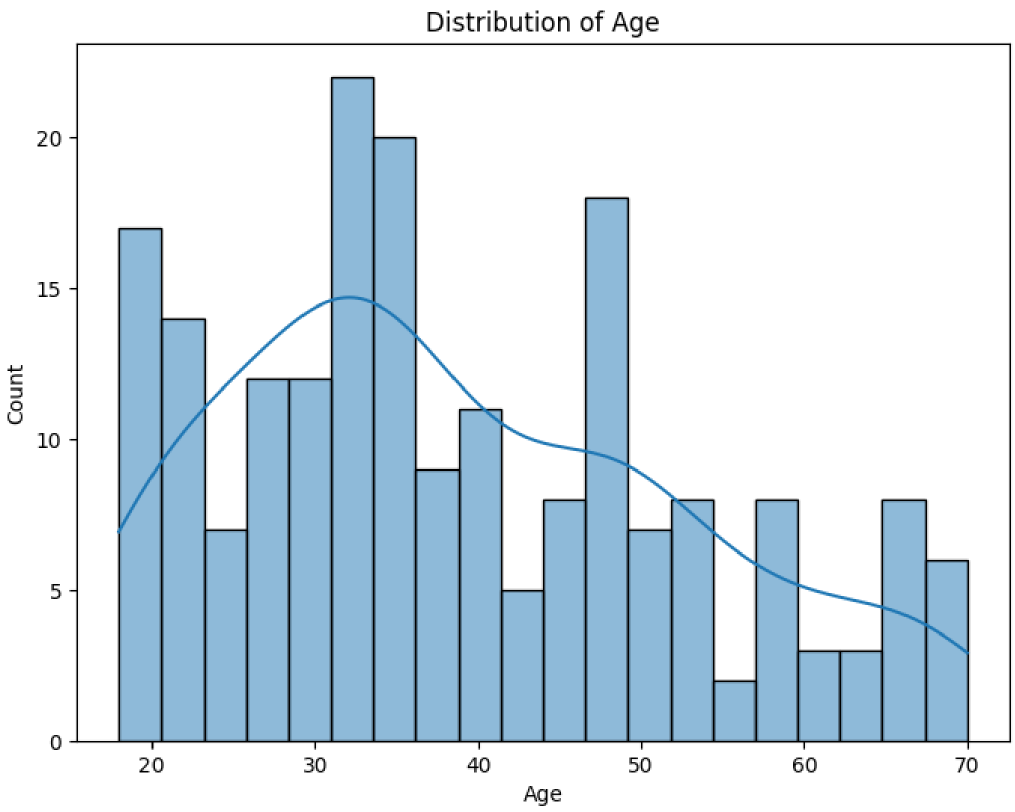
PROGRAM:

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
df = pd.read_csv("C:/Users/lokeshwar k/OneDrive/Documents/naan mudhalavan/Mall_Customers.csv")
print(df.head())
# Check the basic statistics of the dataset
print(df.describe())
# Check for missing values
print(df.isnull().sum())
# Check the data types of each column
print(df.dtypes)
plt.figure(figsize=(8, 6))
sns.histplot(df['Age'], bins=20, kde=True)
plt.title('Distribution of Age')
plt.xlabel('Age')
plt.ylabel('Count')
plt.show()
selected_columns = ['Annual Income (k$)', 'Spending Score (1-100)']
df = df[selected_columns]
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
df_scaled = scaler.fit_transform(df)
df_preprocessed = pd.DataFrame(df_scaled, columns=selected_columns)
df_preprocessed.to_csv('mall_customers_preprocessed.csv', index=False)
```

OUTPUT:



```
    CustomerID   Genre  Age  Annual Income (k$)  Spending Score (1-100)
0            1    Male   19                  15                      39
1            2    Male   21                  15                      81
2            3  Female   20                  16                       6
3            4  Female   23                  16                      77
4            5  Female   31                  17                      40
       CustomerID         Age  Annual Income (k$)  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
CustomerID                0
Genre                     0
Age                       0
Annual Income (k$)        0
Spending Score (1-100)    0
dtype: int64
CustomerID                 int64
Genre                     object
Age                        int64
Annual Income (k$)         int64
Spending Score (1-100)     int64
dtype: object
```



4.2: IN THIS TECHNOLOGY YOU WILL CONTINUE

BUILDING YOUR PROJECT BY PERFORMING FEATURE ENGINEERING

DATA COLLECTION:

- Start by collecting relevant data about mall customers.
- This data can come from sources such as customer surveys, transaction records, loyalty programs, and even demographic information.

DATA PREPROCESSING:

- Clean and preprocess the data to ensure its quality and consistency.
- This step includes handling missing values, outliers, and standardizing or normalizing the data.

FEATURE SELECTION/ENGINEERING:

- Identify the relevant features (attributes) that can be used for segmentation.
- These could include age, gender, income, shopping frequency, spending behavior, and more.
- You may also create new features if they are informative, such as customer age groups or spending categories.

EXPLORATORY DATA ANALYSIS (EDA):

- Use data visualization and summary statistics to gain insights into your customer data.
- Explore relationships between features and look for patterns and trends.

CHOOSE SEGMENTATION VARIABLES:

- Select the variables that you will use to segment customers.
- Common variables include:

- Demographics: Age, gender, income, marital status, etc.
- Behavioral: Purchase history, frequency of visits, average spending, etc.
- Psychographics: Lifestyle, preferences, and interests.
- Geographic: Location or proximity to the mall.
-

SELECT A SEGMENTATION METHOD:

- Choose an appropriate segmentation technique based on your data and objectives.
- Common methods include:
- K-Means Clustering: Groups customers into clusters based on similarity.
- Hierarchical Clustering: Builds a tree-like structure of customer segments.
- DBSCAN: Identifies dense regions of data points as clusters.
- PCA (Principal Component Analysis): Reduces dimensionality for better visualization and interpretation.
- Machine Learning Algorithms: Use supervised learning to predict customer segments.

SEGMENTATION MODELING:

- Implement the chosen segmentation method.
- This process assigns each customer to a specific segment or cluster based on the variables you selected.

EVALUATE AND INTERPRET SEGMENTS:

- Analyze the characteristics of each segment.
- What distinguishes one segment from another?
- Do they have unique preferences, behaviors, or needs?
- Use visualizations and descriptive statistics to understand the segments.

VALIDATION AND REFINEMENT:

- Validate the quality of your segments using techniques like silhouette score (for clustering) or cross-validation (for machine learning models).
- If necessary, refine your segmentation based on validation results.

APPLICATION:

- Apply the customer segments to marketing strategies, product recommendations, store layout, and other aspects of mall management.
- Tailor your approach to the specific needs and preferences of each segment.

CONTINUOUS MONITORING:

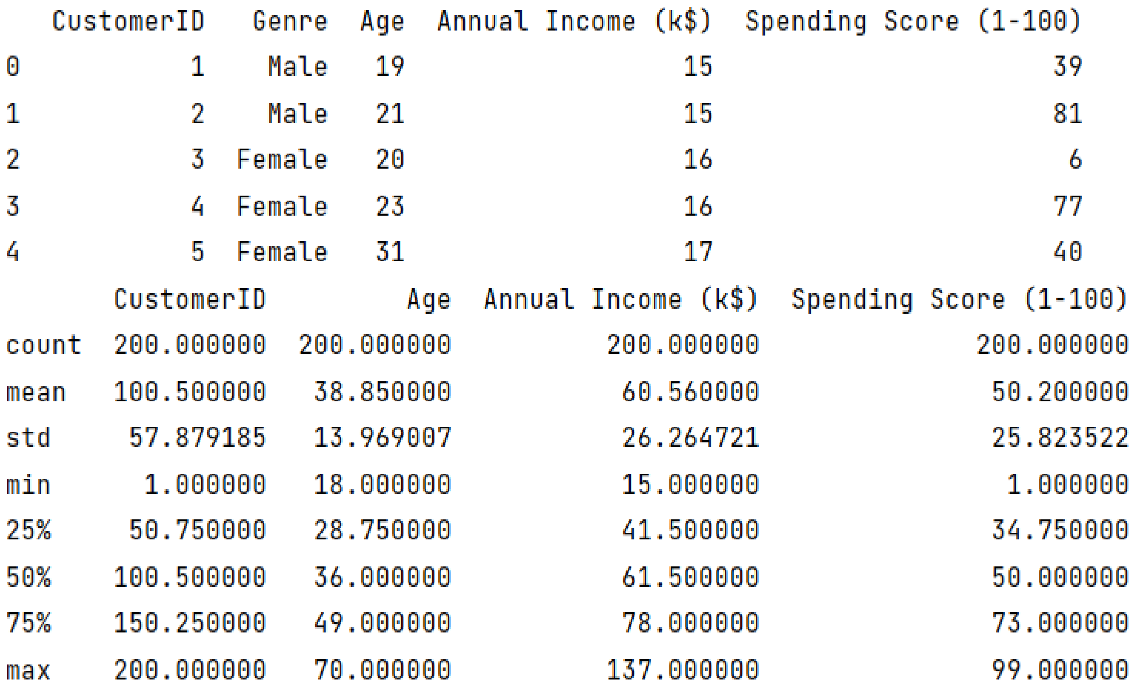
- Customer preferences and behaviors may change over time.
- Continuously monitor and update your segmentation as needed to ensure its relevance.

Customer segmentation in mall management is an iterative process that can yield valuable insights and improve the overall shopping experience for customers while increasing the mall's profitability. Data science and machine learning techniques help make this process more data-driven and effective.

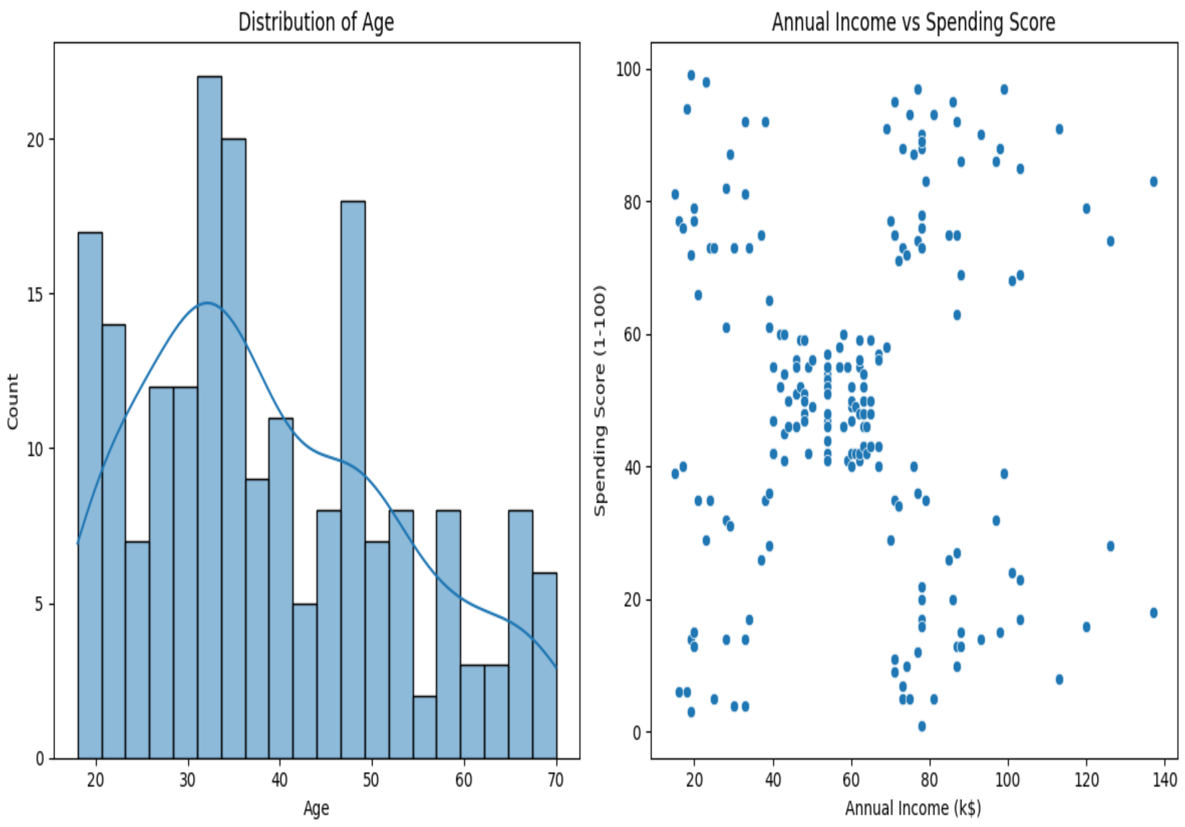
PROGRAM:

```
import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

# Load the dataset

df = pd.read_csv("C:/Users/lokeshwar k/OneDrive/Documents/naan mudhalavan/Mall_Customers.csv")

# Basic data exploration

print(df.head())
print(df.describe())
# Data visualization
plt.figure(figsize=(12, 6))
# Plot a histogram of 'Age' with a density curve
plt.subplot(1, 2, 1)
sns.histplot(df['Age'], bins=20, kde=True)
plt.title('Distribution of Age')
plt.xlabel('Age')
# Plot a scatterplot of 'Annual Income' vs 'Spending Score'
plt.subplot(1, 2, 2)
sns.scatterplot(data=df, x='Annual Income (k$)', y='Spending Score (1-100)')
plt.title('Annual Income vs Spending Score')
plt.tight_layout()
plt.show()
```

OUTPUT:



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



4.3:MODEL TRAINING AND EVALUATION

To build a model for the Mall Customers dataset from Kaggle, you can follow these general steps:

DATA PREPARATION:

- Download the dataset from the Kaggle link you provided.
- Load the dataset using a library like Pandas.
- Explore the dataset to understand its structure, including the columns and data types.

DATA CLEANING:

- Handle missing values if any.
- Convert categorical variables to numerical format through techniques like one-hot encoding.

FEATURE ENGINEERING:

- Create relevant features that can enhance the model's performance, as discussed in the previous response.

DATA SPLITTING:

- Split the dataset into training and testing sets. A common split is 70-30 or 80-20 for training and testing, respectively.

MODEL SELECTION:

- Choose a machine learning model suitable for the problem.
- For a mall customer dataset, you might consider clustering techniques like K-Means, hierarchical clustering, or even regression/classification models depending on the specific problem you want to solve.

MODEL TRAINING:

- Train the selected model on the training dataset.
- Use libraries like Scikit-Learn to implement the model.

MODEL EVALUATION:

- Evaluate the model's performance using appropriate metrics. For clustering models, you can use metrics like Silhouette Score or Davies-Bouldin Index. For regression or classification models, use metrics like Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), accuracy, precision, recall, F1-score, etc.

HYPERPARAMETER TUNING (IF APPLICABLE):

- Optimize model hyperparameters to improve performance.
- You can use techniques like grid search or random search.

MODEL DEPLOYMENT (IF NEEDED):

- If the model performs well and is ready for production use, deploy it in your desired environment

Remember that the choice of the model and specific steps will depend on the problem you want to solve with this dataset. You might want to do further data analysis and consider different types of models, such as regression or classification, depending on your goals.

To perform different analyses on the Mall Customers dataset from Kaggle, you can use various data analysis techniques and tools. Here's a step-by-step guide for conducting different types of analysis on this dataset:

HERE'S A GENERAL EXAMPLE IN PYTHON FOR K-MEANS CLUSTERING:

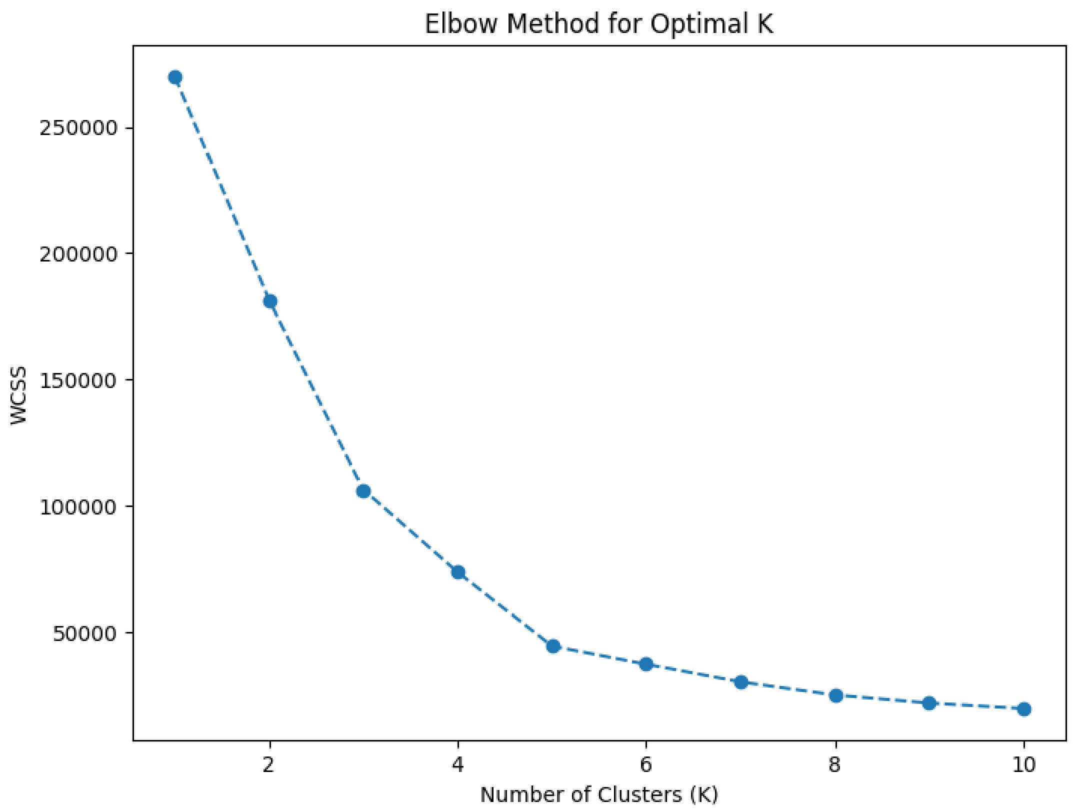
Remember that the choice of the model and specific steps will depend on the problem you want to solve with this dataset. You might want to do further data analysis and consider different types of models, such as regression or classification, depending on your goals.

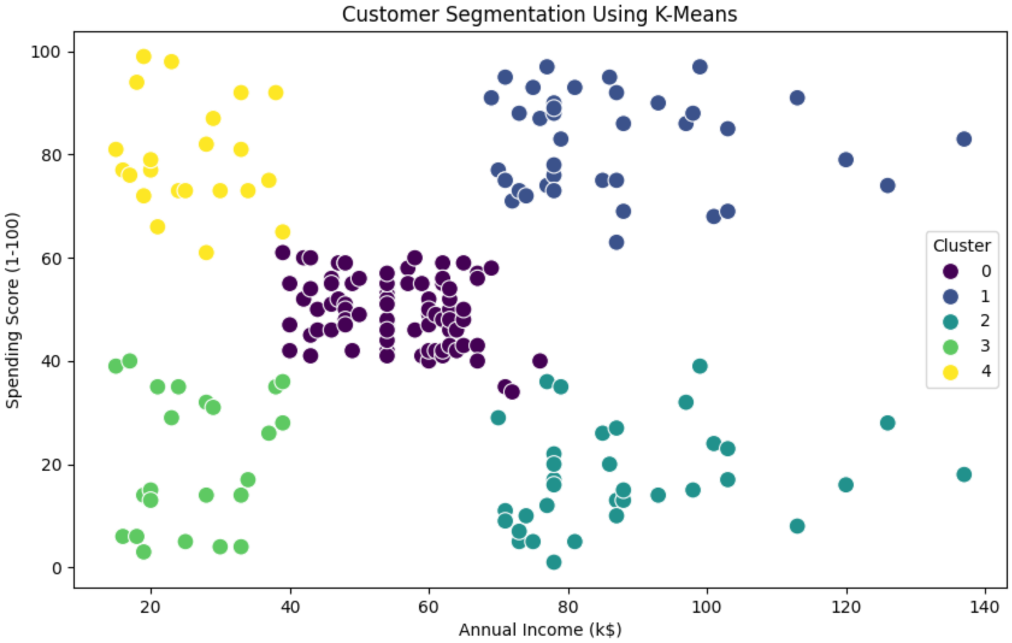
To perform different analyses on the Mall Customers dataset from Kaggle, you can use various data analysis techniques and tools. Here's a step-by-step guide for conducting different types of analysis on this dataset:

PROGRAM:

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_score
# Load the dataset
df = pd.read_csv("C:/Users/lokeshwar k/OneDrive/Documents/naan mudhalavan/Mall_Customers.csv")
# Select the relevant features for clustering
X = df[['Annual Income (k$)', 'Spending Score (1-100)']]
# Determine the optimal number of clusters (K) using the Elbow Method
wcss = []  # Within-Cluster-Sum-of-Squares
for k in range(1, 11):
    kmeans = KMeans(n_clusters=k, random_state=0)
    kmeans.fit(X)
    wcss.append(kmeans.inertia_)
# Plot the Elbow Method graph to find the optimal K
plt.figure(figsize=(8, 6))
plt.plot(range(1, 11), wcss, marker='o', linestyle='--')
plt.title('Elbow Method for Optimal K')
plt.xlabel('Number of Clusters (K)')
plt.ylabel('WCSS')
plt.show()
# Based on the Elbow Method, let's choose K=5
n_clusters = 5
# Train the K-Means model
kmeans = KMeans(n_clusters=n_clusters, random_state=0)
df['Cluster'] = kmeans.fit_predict(X)
# Visualize the clusters
plt.figure(figsize=(10, 6))
sns.scatterplot(data=df, x='Annual Income (k$)', y='Spending Score (1-100)', hue='Cluster', palette='viridis', s=100)
plt.title('Customer Segmentation Using K-Means')
plt.xlabel('Annual Income (k$)')
plt.ylabel('Spending Score (1-100)')
plt.show()
# Evaluate the clustering using silhouette score
silhouette_avg = silhouette_score(X, df['Cluster'])
print(f'Silhouette Score: {silhouette_avg:.2f}')
```

OUTPUT:





4.4: PERFORM DIFFERENT ANALYSIS AS NEEDED

DATA EXPLORATION:

- Load the dataset and examine its structure.
- Explore the summary statistics of numeric columns (e.g., age, income, and spending score) using Pandas.
- Visualize the data to gain insights, e.g., use histograms, box plots, or scatter plots to understand the distribution and relationships among variables.

CUSTOMER SEGMENTATION:

- Perform customer segmentation using clustering techniques like K-Means or Hierarchical Clustering to group customers based on similar characteristics.
- Analyze the resulting clusters to understand customer behavior.

DESCRIPTIVE STATISTICS:

- Calculate and analyze descriptive statistics for various customer groups. For example, compute the mean, median, and standard deviation of spending scores for different segments.

CUSTOMER PROFILING:

- Create customer profiles or personas by summarizing the characteristics of each customer group, such as average age, income, and spending score.

DATA VISUALIZATION:

- Use data visualization libraries like Matplotlib or Seaborn to create informative charts and graphs.

- Visualize the distribution of spending scores, income, and other relevant features across customer segments.

CORRELATION ANALYSIS:

- Explore the correlations between different features. Determine whether there are any strong relationships between variables like age, income, and spending score.

HYPOTHESIS TESTING:

- Formulate hypotheses and conduct statistical tests to confirm or reject these hypotheses. For example, you could test whether there's a significant difference in spending scores between male and female customers.

TIME SERIES ANALYSIS (IF APPLICABLE):

- If the dataset contains timestamp data, perform time series analysis to identify patterns and trends in customer behavior over time.

MACHINE LEARNING PREDICTIVE MODELING (IF NEEDED):

- Build regression or classification models to predict customer behavior, such as spending score, based on other variables.
- Evaluate model performance using appropriate metrics like Mean Squared Error (MSE) for regression or accuracy for classification.

CUSTOMER CHURN ANALYSIS (IF APPLICABLE):

- Analyze customer churn by identifying customers who have stopped shopping at the mall.
- Create churn prediction models to identify customers at risk of leaving.

MARKET BASKET ANALYSIS (IF APPLICABLE):

- If you have transaction data, conduct market basket analysis to identify which products or services are frequently purchased together.

CUSTOMER RETENTION STRATEGIES:

- Based on your analyses, develop customer retention strategies to improve customer satisfaction and loyalty.

To conduct these analyses, you can use Python libraries like Pandas, Matplotlib, Seaborn, Scikit-Learn, and statsmodels for data analysis, visualization, and statistical testing. Additionally, you may use Jupyter notebooks to document your analysis steps and findings.

Keep in mind that the specific analyses you perform should align with your business objectives and the questions you aim to answer using the dataset. The results of your analyses can inform marketing strategies, customer targeting, and business decisions for the mall.