CUSTOMER SEGMENTATION USING DATASCIENCE

TEAM MEMBERS

T.TAMIL SELVAN

V.MUTHUMARIYAPPAN

K.RAHUL

R.PANDIYARAJAN

A.ARUNKUMAR

PHASE 3 SUBMISSION DOCUMENT

PHASE 3: DEVELOPMENT PART 1

INTRODUCTION

Today many of the businesses are going online and, in this case, online marketing is becoming essential to hold customers, but during this, considering all customers as same and targeting all of them with similar marketing strategy is not very efficient way rather it's also annoys the customers by neglecting his or her individuality, so customer segmentation is becoming very popular and also became the efficient solution for this existing problem. Customer segmentation is defined as dividing company's customers on the basis of demographic (age, gender, marital status) and behavioral (types of products ordered, annual income) aspects. Since demographic characteristics does not emphasize on individuality of customer because same age groups may have different interests so behavioral aspects is a better approach for customer segmentation as its focus on individuality and we can do proper segmentation with the help of it.

NECESSARY STEPS TO FOLLOW:

Start by importing libraries:

Pandas:

Pandas is essential for data manipulation and analysis, particularly for loading and handling datasets.

Program:

import pandas as pd

NumPy:

NumPy is used for numerical computations, and it complements Pandas for handling arrays and mathematical operations.

Program:

import NumPy as np

Scikit-Learn (sklearn):

Scikit-Learn provides tools for machine learning, including dataset splitting, preprocessing, and model evaluation. You'll import specific modules as needed for your analysis.

Program:

from sklearn. model selection import train\_test\_split

from sklearn. Preprocessing import StandardScaler # For data scaling (if needed)

Matplotlib and Seaborn:

These libraries are useful for data visualization, which is crucial for understanding your dataset.

Program:

import matplotlib.pyplot as plt

import seaborn as sns

LOAD THE DATASET:

To load a dataset for credit card fraud detection, We can use the Pandas library in Python. Here's how we can load a dataset from a CSV file, which is a common data format:

Program:

import pandas as pd

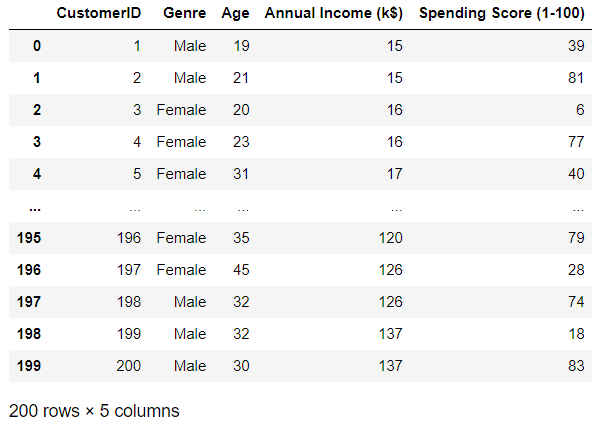
# Specify the file path to your dataset

file\_path = 'Mall\_Customer.csv'

# Use Pandas to read the CSV file into a DataFrame

df = pd.read\_csv(C:\Users\mithu\Documents\Mani\tamil\archive)

# Now, 'df' contains your dataset, and you can start working with it.



In the code above:

Import the Pandas library to work with data.

Replace 'your\_dataset.csv' with the actual file path to our dataset. Make sure that the CSV file is in the same directory as our Python script, or provide the full path to the file if it's located elsewhere.

The pd.read\_csv(C:\Users\mithu\Documents\Mani\tamil\archive) function reads the CSV file and stores its contents in a Pandas DataFrame called df. This DataFrame is a two-dimensional table-like data structure that you can manipulate and analyze.

After loading the dataset into a Data Frame, we can perform various data analysis tasks, such as data exploration, preprocessing, and modelling, depending on your specific objectives in customer segmentation using data science.

DATA EXPLORATION:

Data exploration is an iterative process, and the specific analyses. It's essential to gain a deep understanding of the data to make informed decisions on preprocessing and model selection for customer segmentation using data science.

Basic Summary Statistics:

Use Pandas to obtain summary statistics of the dataset, which can give a quick overview of the data, including counts, and percentiles.

Program:

print(df.describe())

Data Visualization:

Create visualizations to explore the data. Common types of plots include histograms, box plots, and scatter plots. We can use libraries like Matplotlib and Seaborn for this purpose.

Program:

import matplotlib.pyplot as plt

import seaborn as sns

# Example: Annual Income of Male and Female

plt.figure(figsize=(8, 5))

sns.histplot(df['Amount'], bins=50, kde=True)

plt.title(‘Annual Income of Male and Female’)

plt.xlabel('Amount')

plt.ylabel('Count')

plt.show()

Class Distribution:

In customer segmentation using data science, it's essential to understand spending score. An imbalanced dataset might require special handling during modeling.

Program:

df.shape

(200, 5)

x=df[['Genre','Age','Annual Income (k$)']]

Correlation Analysis:

Examine correlations between features. This can help identify relationships between variables and reveal potential patterns or anomalies.

Program:

from sklearn.preprocessing import LabelEncoder

from sklearn.preprocessing import LabelEncoder

x[:,0]=label\_encode\_x.fit\_transform(x[:,0])

Outlier Detection:

Identify and investigate potential outliers, which could be indicative of customer segmentation activities.

Program:

from sklearn.preprocessing import StandardScaler

sc\_x=StandardScaler()

x\_train=sc\_x.fit\_transform(x\_train)

Feature Importance:

To use machine learning models, analyse feature importance to understand which attributes play a significant role in customer segmentation.

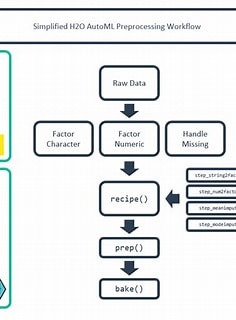
Program:

from sklearn.model\_selection import train\_test\_split

# Fit a model to assess feature importance

x\_train,x\_test,y\_train,y\_test=train\_test\_split(x,y,test\_size=0.2,random\_state=0)

y\_train



CONCLUSION

Customer segmentation is performed on the company's customers data and with the help of K-means clustering machine learning algorithm customers are divided using features like total spending and annual income, this study also proves that the dividing customers on the basis of behavioral characteristics is a better solution for existing customer segmentation problem and K-means clustering algorithm is identified as a good choice for this approach.