Mall Customer Segmentation

A MINI PROJECT REPORT SUBMITTED BY

Pranav Adiga P
4NM18CS114
VI Semester, C Section

Prathvik Shervegar
4NM18CS119
VI Semester, C section

UNDER THE GUIDANCE OF

Dr. Sarika Hegde
Associate professor
Department of Computer Science and Engineering

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF

Bachelor of Engineering in Computer Science & Engineering

from

Visvesvaraya Technological University, Belagavi

NITTE
EDUCATION TRUST

N.M.A.M. INSTITUTE OF TECHNOLOGY

(An Autonomous Institution under VTU, Belgaum)
AICTE approved, (ISO 9001:2015 Certified), Accredited with 'A' Grade by NAAC
NITTE -574 110, Udupi District, KARNATAKA.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

B.E. CSE Program Accredited by NBA, New Delhi from 1-7-2018 to 30-6-2021

May 2021

NITTE
EDUCATION TRUST

N.M.A.M. INSTITUTE OF TECHNOLOGY
(An Autonomous Institution affiliated to Visvesvaraya Technological University, Belagavi)
Nitte – 574 110, Karnataka, India
(ISO 9001:2015 Certified), Accredited with ‘A’ Grade by NAAC
☎: 08258 - 281039 – 281263, Fax: 08258 – 281265

Department of Computer Science and Engineering

B.E. CSE Program Accredited by NBA, New Delhi from 1-7-2018 to 30-6-2021

CERTIFICATE

“Mall Customer Segmentation” is a bonafide work carried out by Pranav Adiga P (4NM18CS114) and Prathvik Shervegar (4NM18CS119) in partial fulfilment of the requirements for the award of Bachelor of Engineering Degree in Computer Science and Engineering prescribed by Visvesvaraya Technological University, Belagavi during the academic year 2020-2021.

It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report. The Mini project report has been approved as it satisfies the academic requirements in respect of the project work prescribed for the Bachelor of Engineering Degree.

Abstract

The main objective of our project is to study the dataset of Mall_Customers, which is the data of customers who visit the mall and spend there. Then we solve our targets by using unsupervised learning methods.

By the end of this case study, you would be able to answer below questions.

1. How to achieve customer segmentation using machine learning algorithm (K-Means Clustering) in Python in simplest way.
2. Who are your target customers with whom you can start marketing strategy [easy to converse].
3. How the marketing strategy works in real world.

Table of Contents

Sl.No.| Title | Page No.

Introduction

Let's imagine you're owning a supermarket mall and through membership cards, you have some basic data about your customers like Customer ID, age, gender, annual income and spending score, which is something you assign to the customer based on your defined parameters like customer behavior and purchasing data.

The main aim of this problem is learning the purpose of the customer segmentation concepts, also known as market basket analysis, trying to understand customers and separate them in different groups according to their preferences, and once the division is done, this information can be given to marketing team so they can plan the strategy accordingly.

This dataset is composed by the following five features:

- ***CustomerID***: Unique ID assigned to the customer
- ***Gender***: Gender of the customer
- ***Age***: Age of the customer
- ***Annual Income (k$)***: Annual Income of the customer
- ***Spending Score (1-100)***: Score assigned by the mall based on customer behavior and spending nature.

Literature Survey

A. Customer Classification

Over the years, the commercial world has become more competitive, as organizations such as these have to meet the needs and desires of their customers, attract new customers, and thus improve their businesses. The task of identifying and meeting the needs and requirements of every customer in the business is very difficult. This is because customers can vary according to their needs, wants, demographics, size, taste and taste, features etc. As it is, it is a bad practice to treat all customers equally in business. This challenge has adopted the concept of customer segmentation or market segmentation, where consumers are divided into subgroups or segments, where members of each subcategory exhibit similar market behaviors or characteristics. Accordingly, customer segmentation is the process of dividing the market into indigenous groups.

B. Data Repository

Data collection is the process of collecting and measuring information against targeted changes in an established system, which enables one to answer relevant questions and evaluate the results. Data collection is part of research in all fields of study including physical and social sciences, humanities and business. The purpose of all data collection is to obtain quality evidence that leads the analysis to construct concrete and misleading answers to the questions presented. We collected data from the UCI machine learning repository.

C. Clustering data

Clustering is the process of grouping information into a dataset based on some commonalities. There are several algorithms, which can be applied to datasets based on the provided condition. However, no universal clustering algorithm exists, hence it becomes important to choose the appropriate clustering techniques. In this paper, we have implemented three clustering algorithms using the Python scalar library.

D. K-means Clustering Algorithm

K-means is one of the most popular classification algorithms. This clustering algorithm relies on centro, where each data point is placed in one of the overlapping ones, which is pre-sorted in the K-algorithm. Clusters are created that correspond to hidden patterns in the data that provide the necessary information to help decide execution. process. There are many ways to make assembling K-means, we will use the elbow method.

Design and Analysis

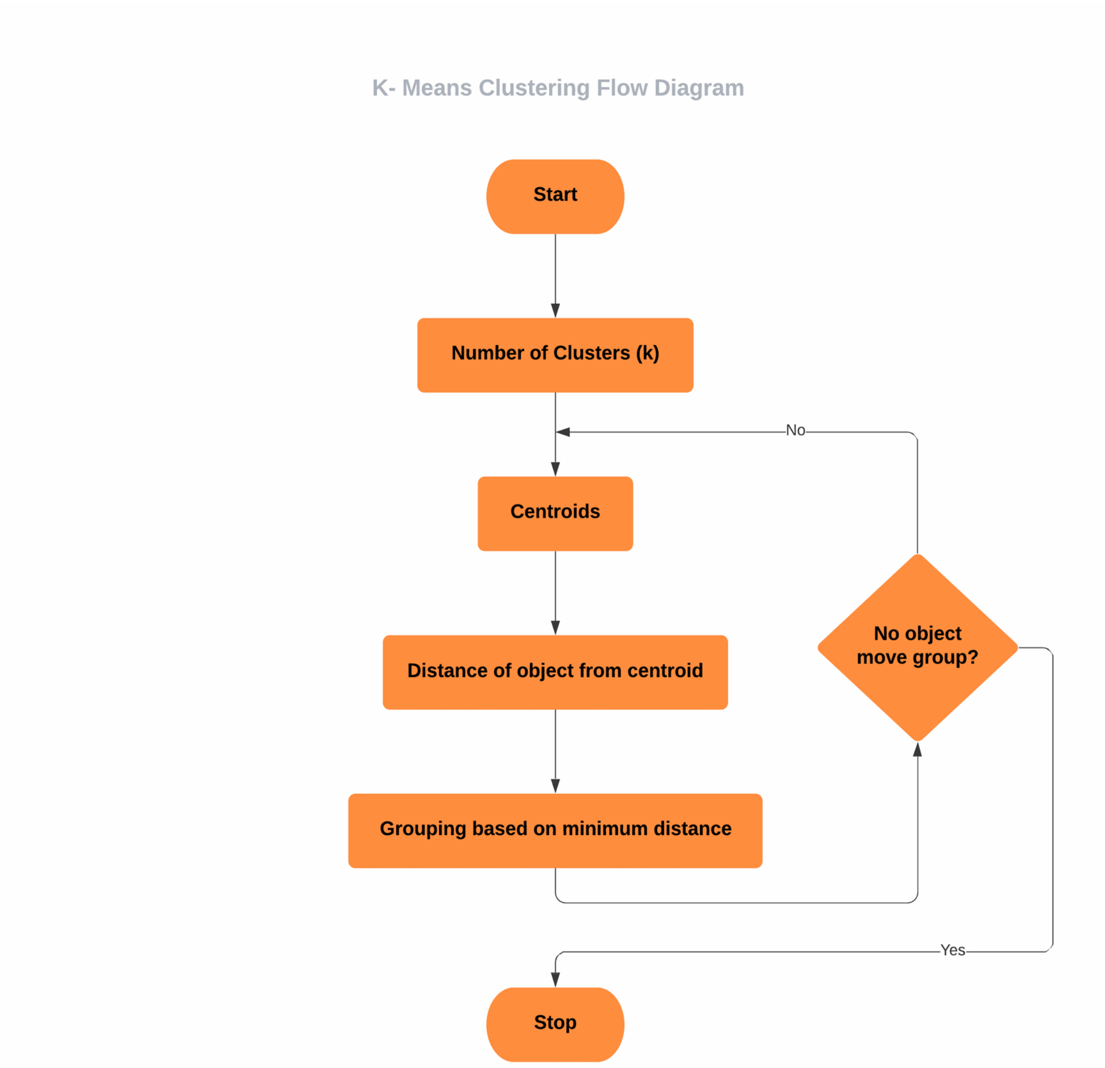

K- Means Clustering Flow Diagram
Start
Number of Clusters (k)
No
Centroids
Distance of object from centroid
No object move group?
Grouping based on minimum distance
Yes
Stop

Figure 1: Flow diagram of project

The working of the K-Means algorithm is explained in the below steps:

Step-1: Select the number K to decide the number of clusters.

Step-2: Select random K points or centroids. (It can be other from the input dataset).

Step-3: Assign each data point to their closest centroid, which will form the predefined K clusters.

Step-4: Calculate the variance and place a new centroid of each cluster.

Step-5: Repeat the third steps, which means reassign each datapoint to the new closest centroid of each cluster.

Step-6: If any reassignment occurs, then go to step-4 else go to FINISH.

Step-7: The model is ready.

In the k means clustering algorithm, the first parameter that needs to be specified is the value of K i.e. the number of clusters. After this value is determined, these k points are chosen as cluster centers. All instances that need to be classified are assigned to their closest cluster center, according to simple Euclidean distance metric. Next, the centroid or the mean of all instances in each cluster is calculated. These center or mean values are taken to be the new center values for their respective clusters. The process is then repeated iteratively until the same points are assigned to cluster centers in consecutive rounds, at which stage the cluster centers are stabilized and do not change after this point.

Choosing the value of "K number of clusters" in K-means Clustering:

The performance of the K-means clustering algorithm depends upon highly efficient clusters that it forms. But choosing the optimal number of clusters is a big task. There are some different ways to find the optimal number of clusters, but here we are using the most appropriate method to find the value of K.

The idea of the elbow method is to run k-means clustering on the dataset for a range of values of *k* (say, *k* from 1 to 10 in the examples above), and for each value of *k* calculate the sum of squared errors (SSE).

Implementation

The steps to be followed for the implementation are given below:

i. Data Pre-processing

ii. Finding the optimal number of clusters using the elbow method

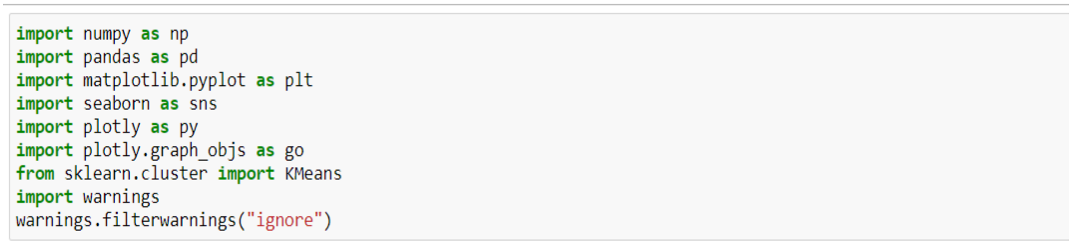
iii. Training the K-means algorithm on the training dataset

iv. Visualizing the clusters

Step-1: Data pre-processing

The first step will be the data pre-processing.

I. Importing Libraries: We will import the libraries for our model, which is part of data pre-processing. The code is given below:



```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import plotly as py
import plotly.graph_objs as go
from sklearn.cluster import KMeans
import warnings
warnings.filterwarnings("ignore")
```

numpy|pandas: Will help us treat and explore the data, and execute vector and matrix operations.

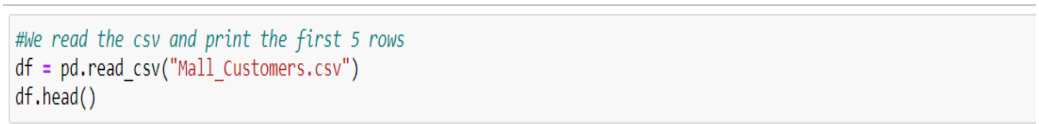
matplotlib|seaborn: Will help us plot the information so we can visualize it in different ways and have a better understanding of it.

plotly: Will also help us plotting data in a fancy way.

sklearn: Will provide all necessary tools to train our models and test them afterwards.

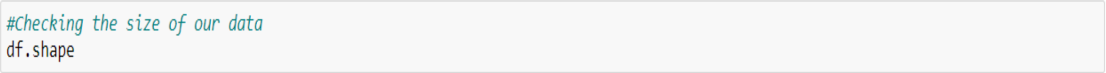
II. Data Exploration:

Next, we will import the dataset that we need to use. We will be using the Mall_Customer_data.csv dataset. It can be imported using the below code:

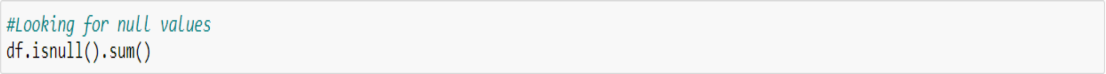


```
#We read the csv and print the first 5 rows
df = pd.read_csv("Mall_Customers.csv")
df.head()
```

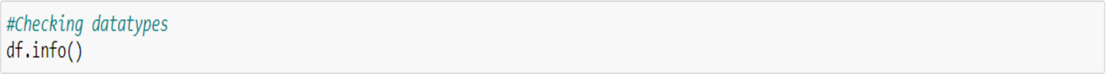
Checking for null values, object data types and other things we might consider in order to keep our data clean and well structured.



```
#Checking the size of our data
df.shape
```

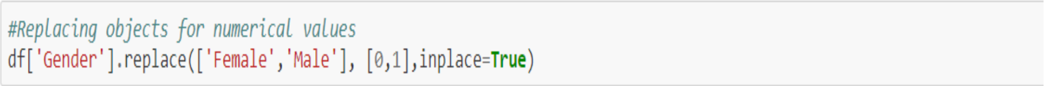


```
#Looking for null values
df.isnull().sum()
```



```
#Checking datatypes
df.info()
```

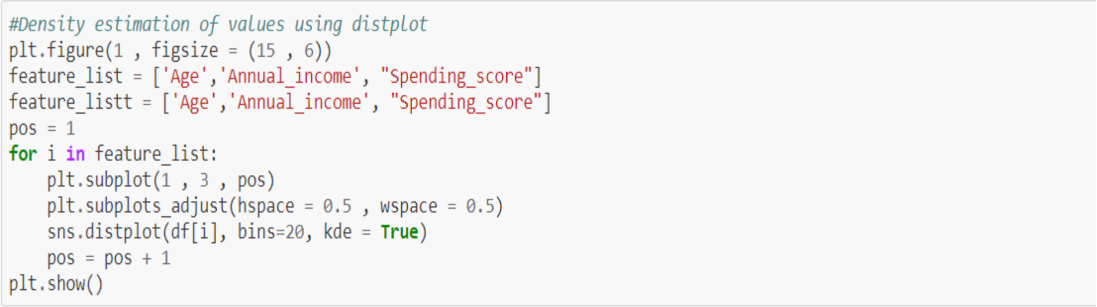
Since Gender is not a numerical value but an object, we are going to replace these values. Female will be 0 and Male will be 1 from now on.



```
#Replacing objects for numerical values
df['Gender'].replace(['Female','Male'], [0,1],inplace=True)
```

III. Data Visualization

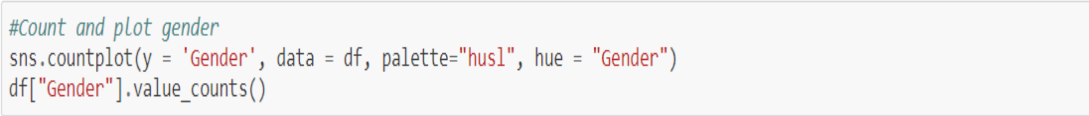
To begin with, we are plotting the histograms for each of the three features we said we would look into:



```
#Density estimation of values using distplot
plt.figure(1 , figsize = (15 , 6))
feature_list = ['Age','Annual_income', "Spending_score"]
feature_listt = ['Age','Annual_income', "Spending_score"]
pos = 1
for i in feature_list:
    plt.subplot(1 , 3 , pos)
    plt.subplots_adjust(hspace = 0.5 , wspace = 0.5)
    sns.distplot(df[i], bins=20, kde = True)
    pos = pos + 1
plt.show()
```

In these histograms we can observe that the distribution of these values resembles a Gaussian distribution, where the vast majority of the values lay in the middle with some exceptions in the extremes.

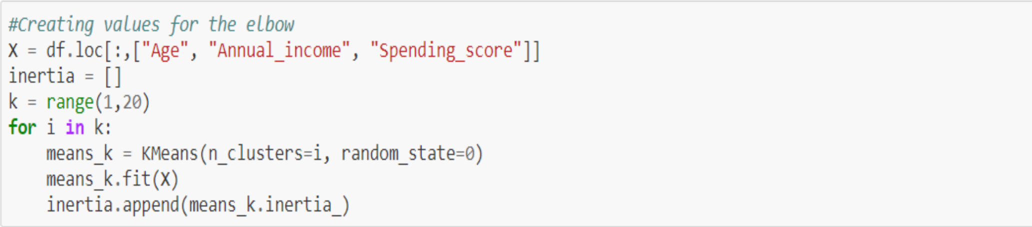
Then check how many women and men there are in our data



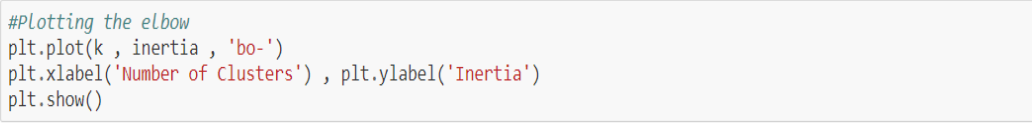
```
#Count and plot gender
sns.countplot(y = 'Gender', data = df, palette="husl", hue = "Gender")
df["Gender"].value_counts()
```

Step-2: Finding the optimal number of clusters using the elbow method

Now that we have already understood this dataset a little bit it's time to decide the amount of clusters we want to divide our data in. To do so, we are going to use the Elbow Method.



```
#Creating values for the elbow
X = df.loc[:,["Age", "Annual_income", "Spending_score"]]
inertia = []
k = range(1,20)
for i in k:
    means_k = KMeans(n_clusters=i, random_state=0)
    means_k.fit(X)
    inertia.append(means_k.inertia_)
```



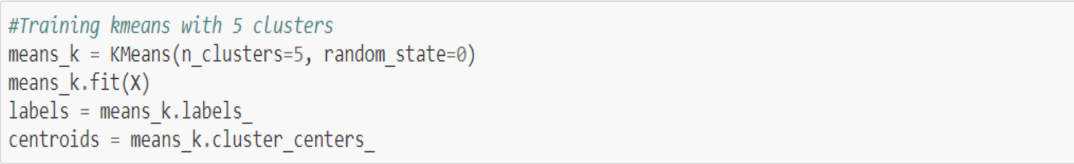
```
#Plotting the elbow
plt.plot(k , inertia , 'bo-')
plt.xlabel('Number of Clusters') , plt.ylabel('Inertia')
plt.show()
```

The elbow method is used to determine the optimal number of clusters in k-means clustering. The elbow method plots the value of the cost function produced by different values of k and one should choose a number of clusters so that adding another cluster doesn't give much better modeling of the data. In this problem, we are using the inertia as cost function in order to identify the sum of squared distances of samples to the nearest cluster center.

Looking at this particular example, if we imagine the line in the graphic is an arm, the elbow can be found, approximately, where the number of clusters is equal to 5. Therefore we are selecting 5 as the number of clusters to divide our data in.

Step- 3: Training the K-means algorithm on the training dataset

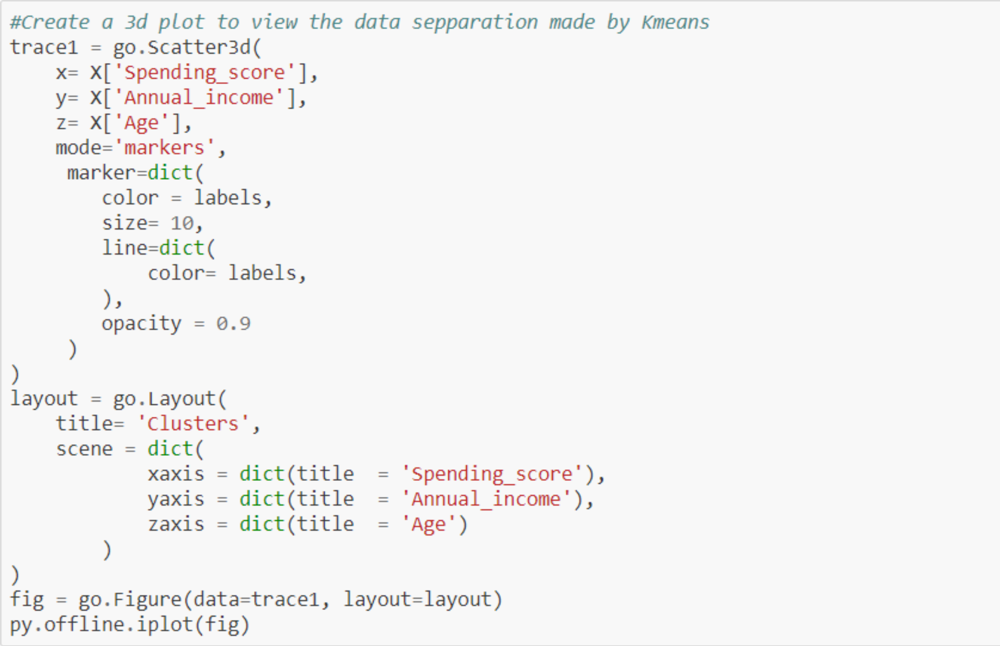
In the process of clustering we will not be considering the gender factor anymore. The main reason of why we do take this approach is because the difference between male and female in this data is not particularly high and making a gender differentiaton won't provide any further information.



```
#Training kmeans with 5 clusters
means_k = KMeans(n_clusters=5, random_state=0)
means_k.fit(X)
labels = means_k.labels_
centroids = means_k.cluster_centers_
```

Step-4: Visualizing the Clusters

As we can observe, the K-means algorithm has already finished its work and now it's time to plot the results we obtained by it so we can visualize the different clusters and analyze them.

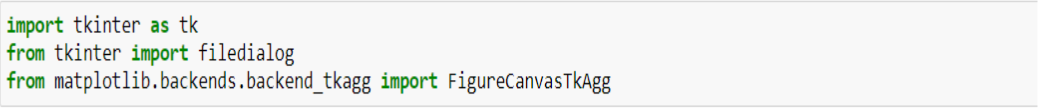


```
#Create a 3d plot to view the data sepparation made by Kmeans
trace1 = go.Scatter3d(
    x= X['Spending_score'],
    y= X['Annual_income'],
    z= X['Age'],
    mode='markers',
     marker=dict(
        color = labels,
        size= 10,
        line=dict(
            color= labels,
        ),
        opacity = 0.9
     )
)
layout = go.Layout(
    title= 'Clusters',
    scene = dict(
            xaxis = dict(title  = 'Spending_score'),
            yaxis = dict(title  = 'Annual_income'),
            zaxis = dict(title  = 'Age')
        )
)
fig = go.Figure(data=trace1, layout=layout)
py.offline.iplot(fig)
```

Front-End:

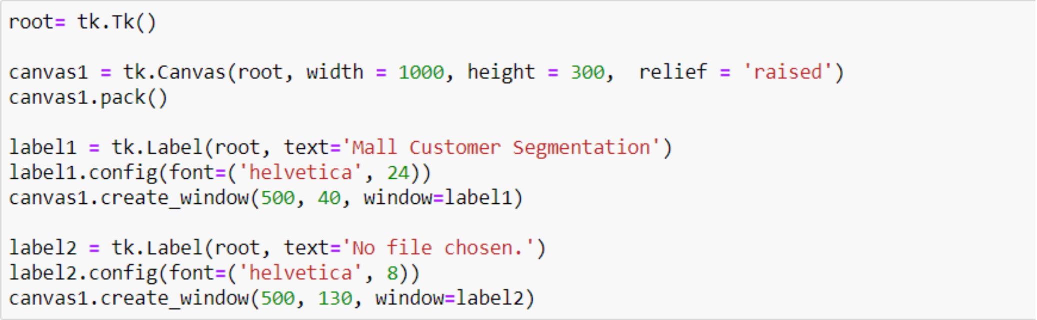
We used Tkinter to deploy our model to user on a GUI, where we user can provide their dataset and will be able to visualize the clusters.

I. Additional Libraries:

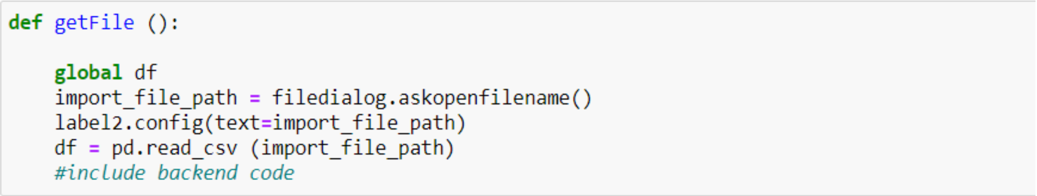


```
import tkinter as tk
from tkinter import filedialog
from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg
```

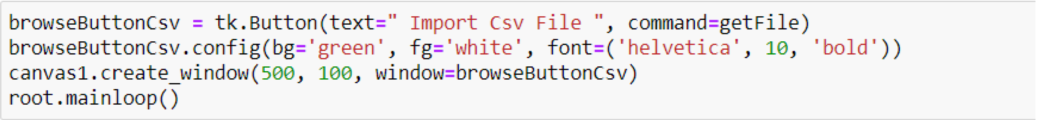
II. Creating GUI:



```
root= tk.Tk()

canvas1 = tk.Canvas(root, width = 1000, height = 300,  relief = 'raised')
canvas1.pack()

label1 = tk.Label(root, text='Mall Customer Segmentation')
label1.config(font=('helvetica', 24))
canvas1.create_window(500, 40, window=label1)

label2 = tk.Label(root, text='No file chosen.')
label2.config(font=('helvetica', 8))
canvas1.create_window(500, 130, window=label2)
```



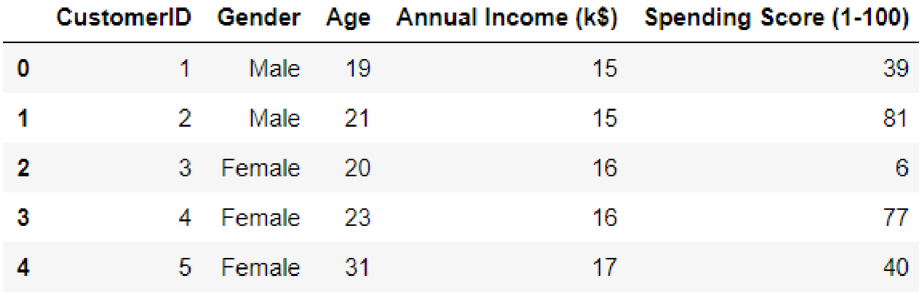
```
def getFile ():

    global df
    import_file_path = filedialog.askopenfilename()
    label2.config(text=import_file_path)
    df = pd.read_csv (import_file_path)
    #include backend code
```



```
browseButtonCsv = tk.Button(text=" Import Csv File ", command=getFile)
browseButtonCsv.config(bg='green', fg='white', font=('helvetica', 10, 'bold'))
canvas1.create_window(500, 100, window=browseButtonCsv)
root.mainloop()
```

Dataset is taken as input and it is processed, then an elbow graph is plotted, based on elbow point k-means model is plotted in 3-dimension through which visualizing characteristics of customer is easier.

Results



	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

Figure 2: First 5 rows of dataset

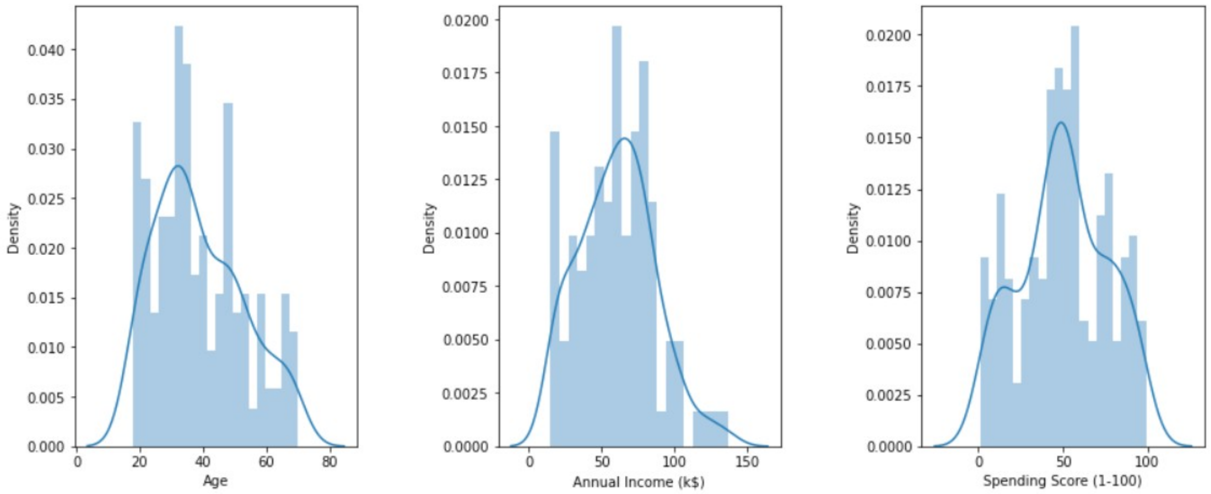


Figure 3: Density estimation of values using distplot

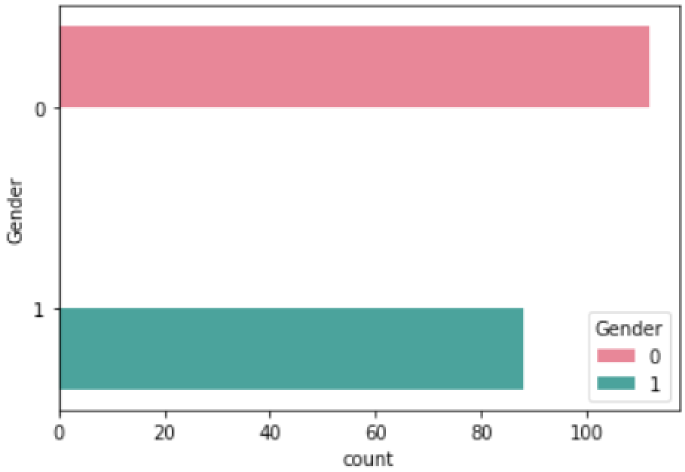


Figure 4: Count and plot gender

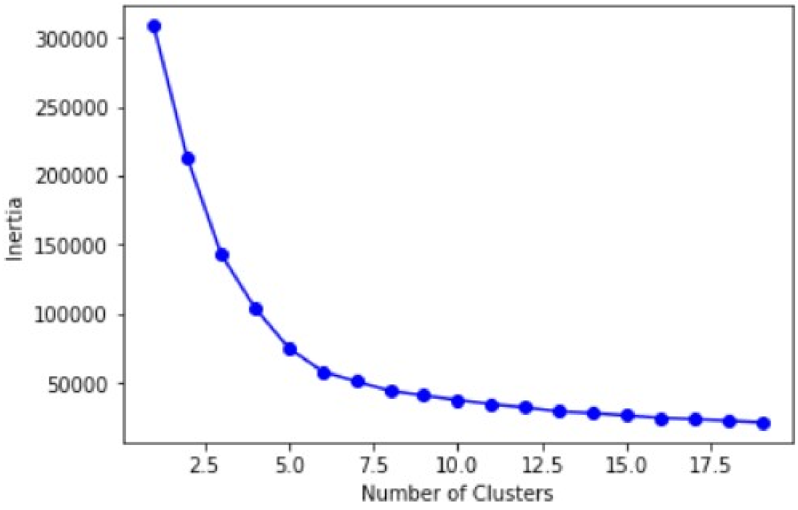
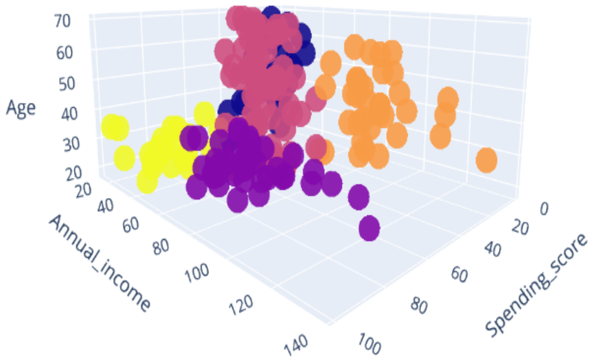


Figure 5: Plotting the elbow



After plotting the results obtained by K-means on this 3D graphic, it's our job now to identify and describe the five clusters that have been created:

1) Yellow Cluster - The yellow cluster groups young people with moderate to low annual income who actually spend a lot.
2) **Purple Cluster** - The purple cluster groups reasonably young people with pretty decent salaries who spend a lot.
3) Pink Cluster - The pink cluster basically groups people of all ages whose salary isn't pretty high and their spending score is moderate.
4) **Orange Cluster** - The orange cluster groups people who actually have pretty good salaries and barely spend money, their age usually lays between thirty and sixty years.
5) **Blue Cluster** - The blue cluster groups whose salary is pretty low and don't spend much money in stores, they are people of all ages.

Figure 6: Create a 3d plot to view the data separation made by Kmeans

Front-End:

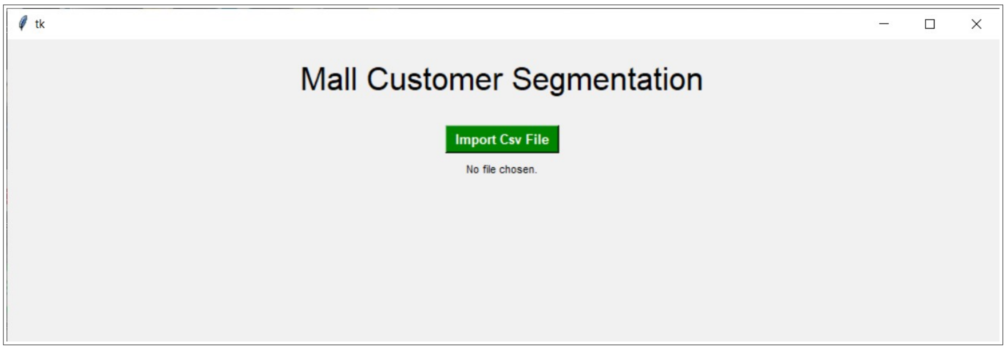


Figure 7: Launch window

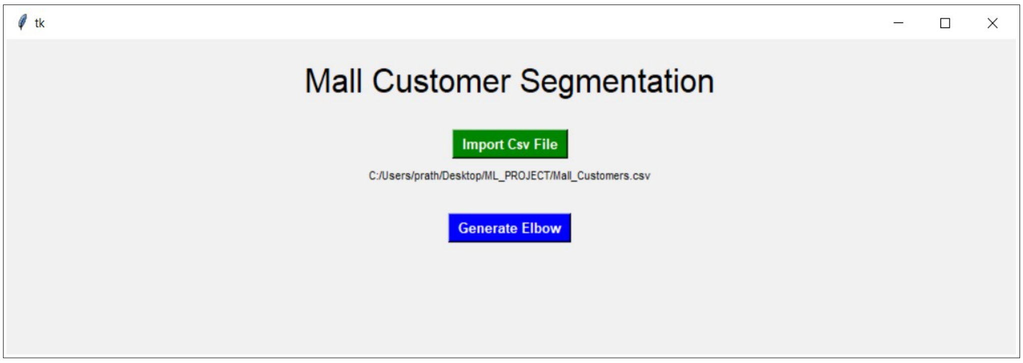

tk
Mall Customer Segmentation
Import Csv File
C:/Users/prath/Desktop/ML_PROJECT/Mall_Customers.csv
Generate Elbow

Figure 8: Choosing csv file

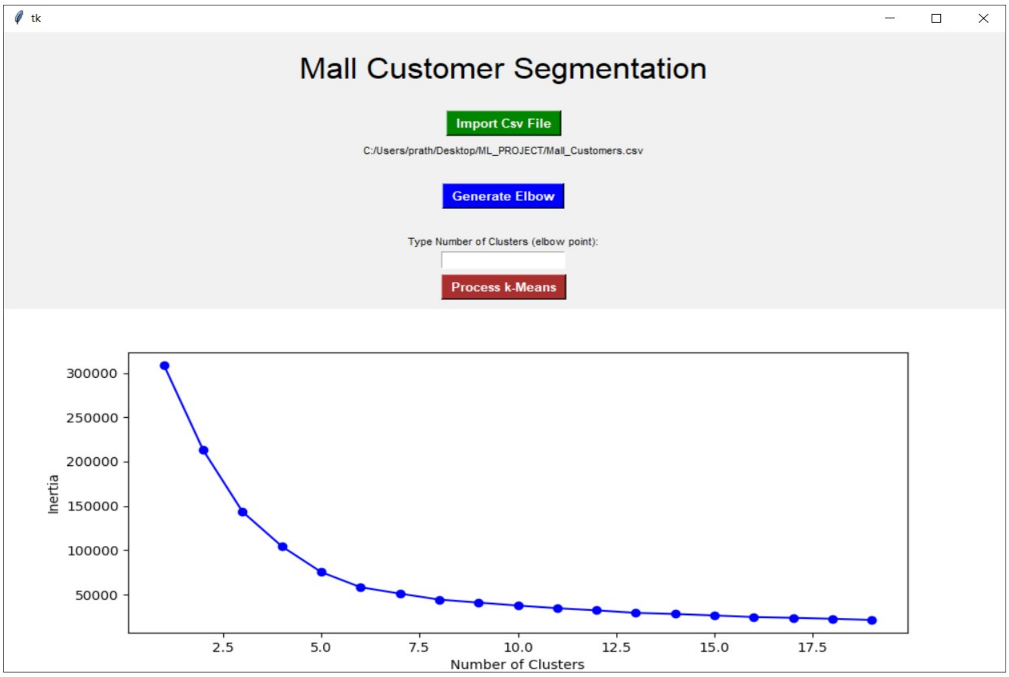
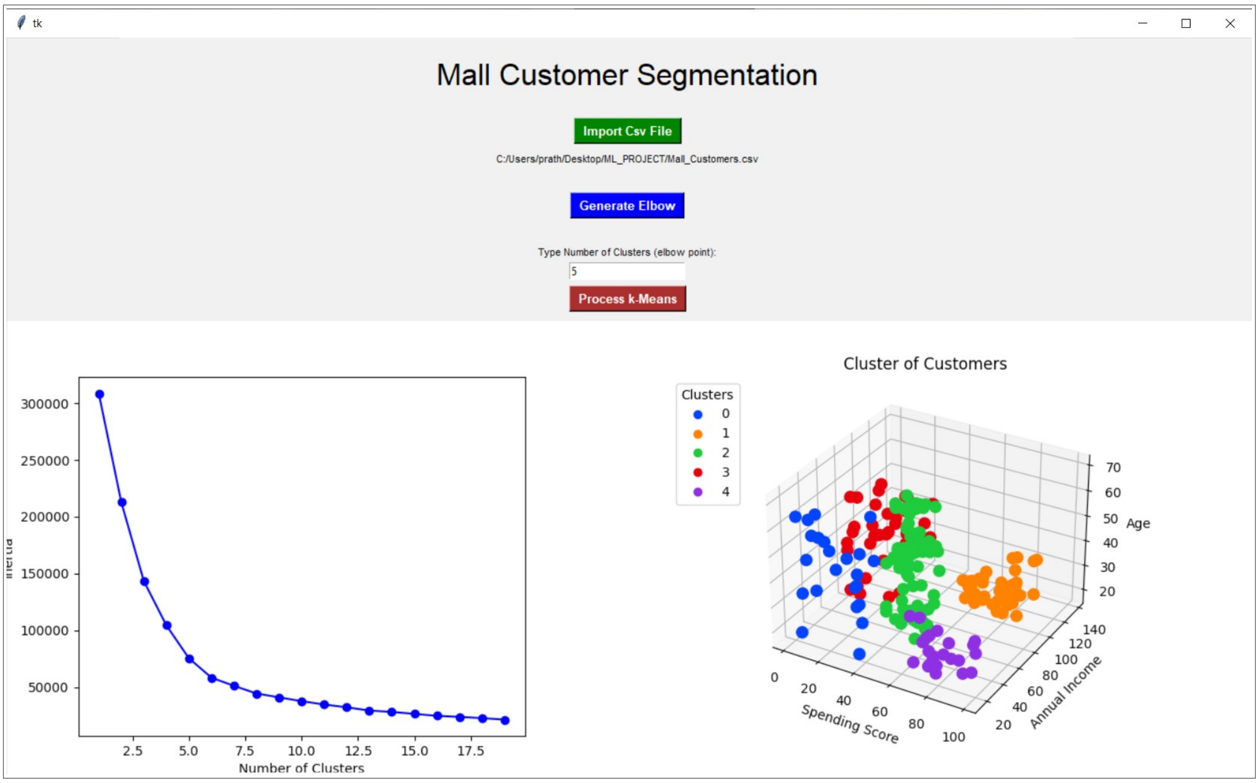


Figure 9: Generating Elbow Graph



tk

Mall Customer Segmentation

C:/Users/prath/Desktop/ML_PROJECT/Mall_Customers.csv

Type Number of Clusters (elbow point): 5

Figure 10: Processing kmeans to visualize clusters

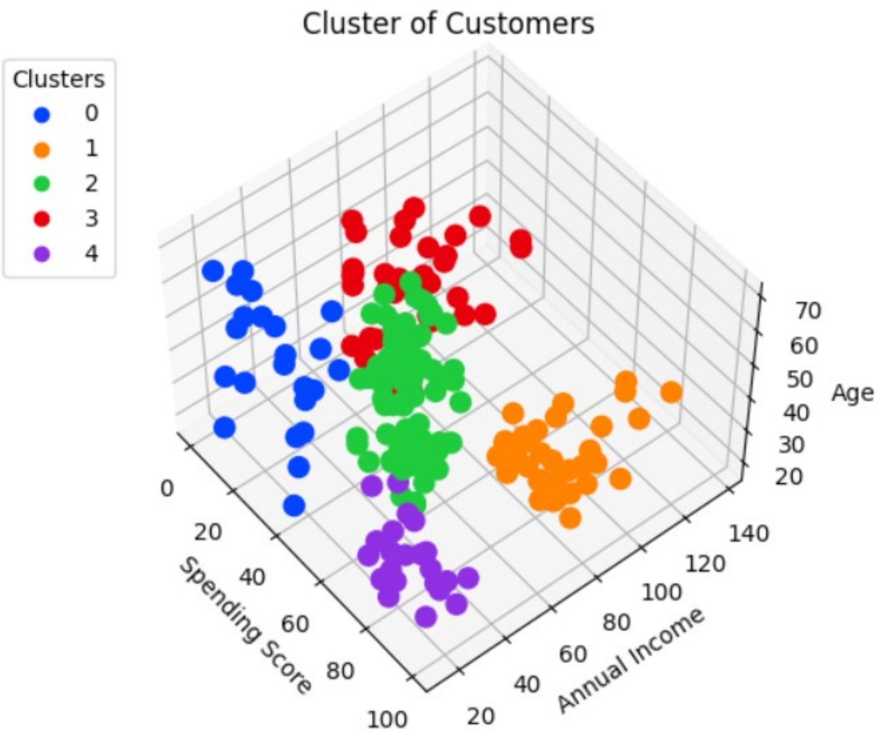


Figure 11: 3D clusters

Conclusion

After developing a solution for this problem, we have come to the following conclusions:

- KMeans Clustering is a powerful technique in order to achieve a decent customer segmentation.
- Customer segmentation is a good way to understand the behaviour of different customers and plan a good marketing strategy accordingly.
- There isn't much difference between the spending score of women and men, which leads us to think that our behaviour when it comes to shopping is pretty similar.
- Observing the clustering graphic, it can be clearly observed that the ones who spend more money in malls are young people. That is to say they are the main target when it comes to marketing, so doing deeper studies about what they are interested in may lead to higher profits.
- Although younglings seem to be the ones spending the most, we can't forget there are more people we have to consider, like people who belong to the pink cluster, they are what we would commonly name after "middle class" and it seems to be the biggest cluster.
- Promoting discounts on some shops can be something of interest to those who don't actually spend a lot and they may end up spending more.

References

[1] Stack Overflow – www.stackoverflow.com

[2] Github – www.github.com

[3] GeeksforGeeks – www.geeksforgeeks.org

[4] Dataset – https://www.kaggle.com/vjchoudhary7/customer-segmentation-tutorial-in-python