PREDICTING CUSTOMER SEGMENTS USING MACHINE LEARNING

VANAPALLI DIMPLE SATYA DEEPAK

DATE:`30-11-2023

Abstract:

Artificial Intelligence (AI) has been growing considerably over the last ten years. Machine Learning (ML) is probably the most popular branch of AI to date. Most systems that use ML methods use them to perform predictive analysis.

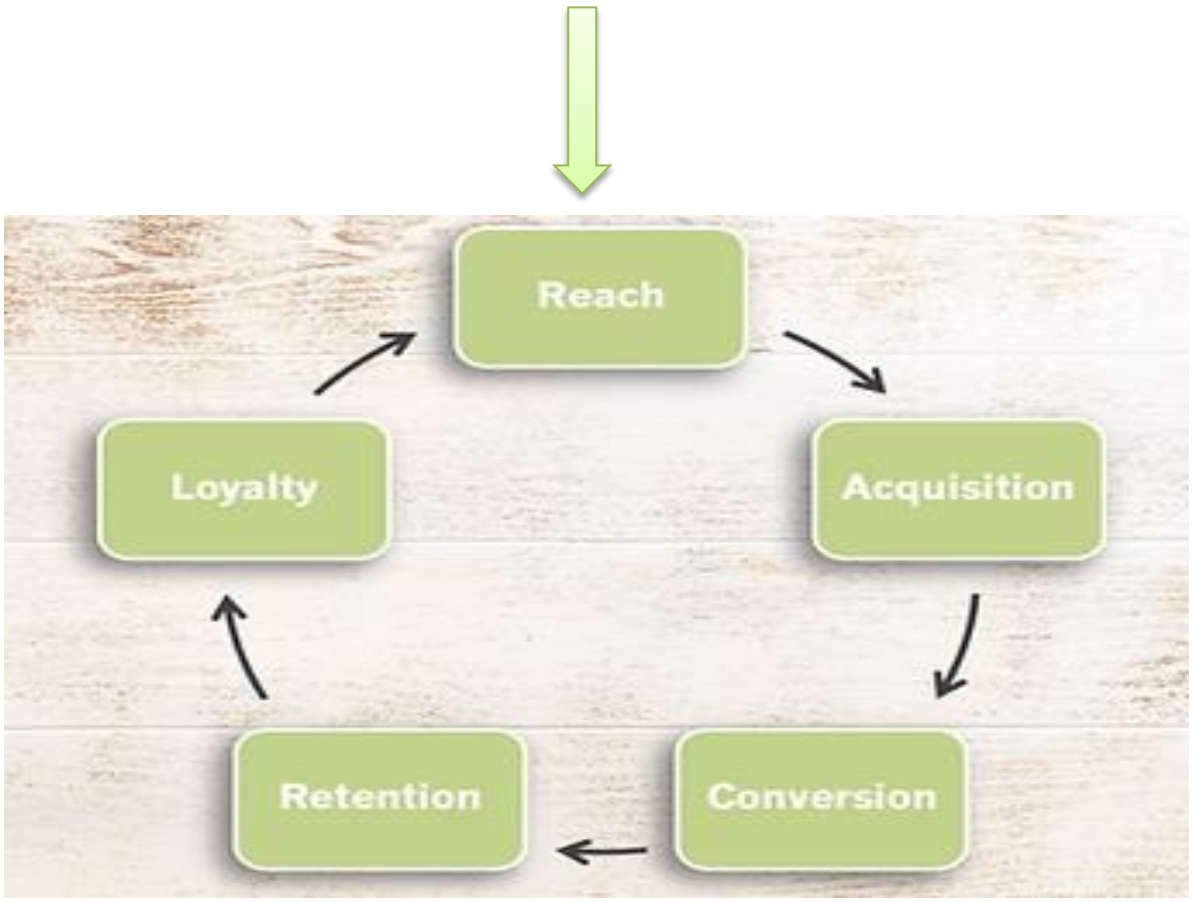
➡ In this report, I have proposed the idea of Predicting Customer Segmentation, an Machine Learning technique used in marketing to identify and create customer segments based on the high probability of occurrences of certain behaviours, events or conditions in the future, It's a way for organizations to understand their customers, knowing the differences between customer groups, it's easier to make strategic decisions regarding product growth and marketing.

➡ Machine learning methodologies are a great tool for analyzing customer data and finding insights and patterns. There are many machine learning algorithms, each suitable for a specific type of problem. One very common machine learning algorithm that's suitable for customer segmentation is the k-means clustering algorithm.

1. Problem Statement:

The goal of **customer segmentation** is to reach out to customers more effectively, thereby leading to more sales or customer conversions. Companies also hope to gain a deeper understanding of their customers' preferences and needs by discovering what each segment finds most valuable and more accurately tailoring marketing materials toward that segment. The customer segmentation helps to find the customers who are about to churn. Manual customer segmentation is time-consuming. It takes months, even years to analyze data and find patterns manually. Also if done heuristically, it may not have the accuracy to be useful as expected.The performance of the model is far better when we use machine learning.

The Customer Life Cycle



2. Customer Need Assessment:

- A Customer needs assessment is a detailed look at the needs and expectations of your customers. If you don't know what is most important to your customers, it is difficult to fulfill their needs and meet their expectations. It is easy to assume you know what your customers want and what is important to them.
- We have to make sure that the proposed method is able to examine the customer needs and fulfill the needs of the customer.
- Segmenting customers into groups based on their **Age, Annual Income, Spending Time**.
- The Advantages of Customer Segmentation includes Promotion and Marketing
- **Promotion**- Properly implemented customer segmentation helps you plan special offers and deals. If you reach a customer with just the right offer, at the right time, there's a huge chance they're going to buy. Customer segmentation will help you tailor your special offers perfectly.
- **Marketing**- The marketing strategy can be directly improved with segmentation because you can plan personalized marketing campaigns for different customer segments.
- Customer Segmentation uses Machine Learning Models to get better Accuracy and customer needs .

3. Target Specifications And Characterization:

- In this customer segmentation project, the used features from Customer Mall Data are:
    1) Age- The age of customer visited the mall.
    2) Annual Income- The annual income of customer visited the mall in (k$).
    3) Spending Score- The spending score in the mall(1-100).

4. External Search (information sources/references):

- The data set used for customer segmentation is Mall Customer Dataset.
- The dataset can be found in Kaggle (https://www.kaggle.com).
- The dataset contains 281 rows and columns of (Customer ID, Gender, Age, Annual Income, Spending Time).
- Link for dataset https://github.com/VANAPALLI-DIMPLE-SATYA-DEEPAK/customer-segmentation/blob/main/Mall_Customers_Data.csv

❖ Exploring dataset and its features

➢ Importing the required libraries

```
import pandas as pd
import numpy as np
from sklearn.cluster import KMeans
import plotly.express as px
import plotly.graph_objects as go
import matplotlib.pyplot as plt
```

➢ Loading the dataset

```
#Load customers data
customersdata = pd.read_csv("Mall_Customers_Data.csv")
```

➢ Analysing the dataset

```
#viewing the dataset
customersdata
```

	Mall Customer ID	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
...	...	...	...	...	...
195	196	Female	35	120	79
196	197	Female	45	126	28
197	198	Male	32	126	74
198	199	Male	32	137	18
199	200	Male	30	137	83

200 rows × 5 columns

➢ Pre-processing of dataset

We need to pre-process the dataset.

```
[15] customersdata.shape
```

```
(200, 6)
```

```
# Information about dataset
customersdata.info()
```

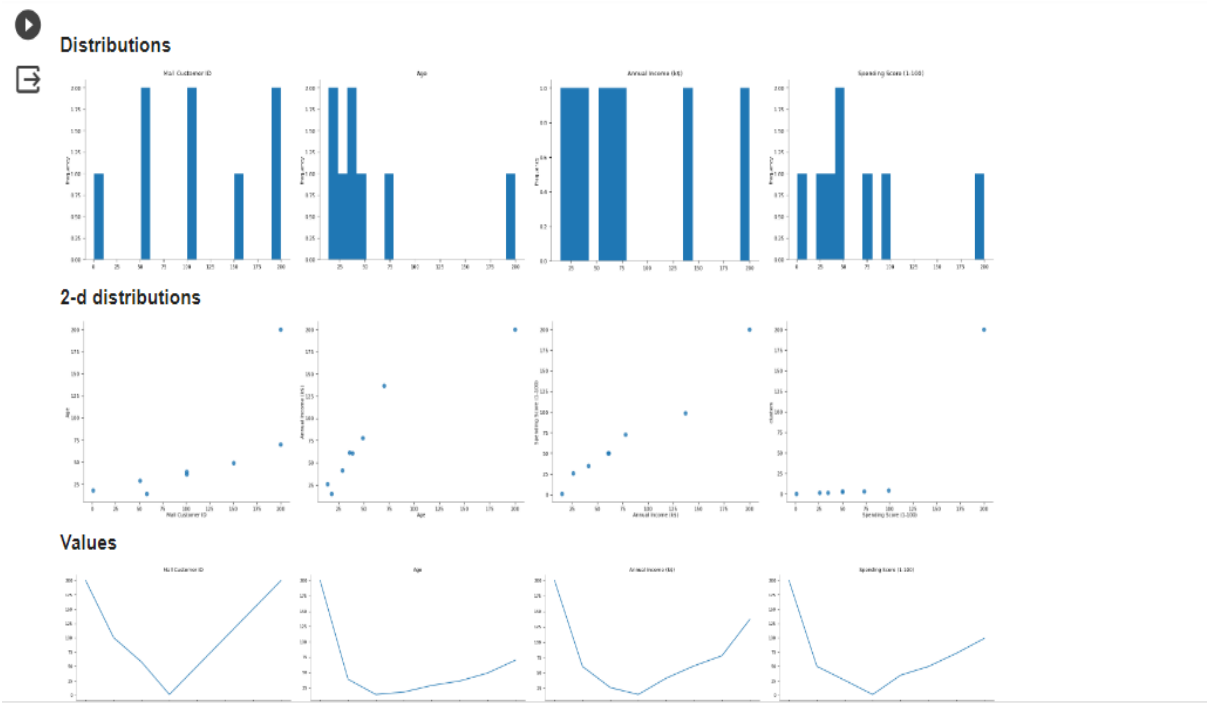
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 6 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   Mall Customer ID        200 non-null    int64
 1   Gender                  200 non-null    object
 2   Age                     200 non-null    int64
 3   Annual Income (k$)      200 non-null    int64
 4   Spending Score (1-100)  200 non-null    int64
 5   clusters                200 non-null    int32
dtypes: int32(1), int64(4), object(1)
memory usage: 8.7+ KB
```

➢ Describing the dataset

```
# Discribing the dataset
customersdata.describe()
```

	Mall Customer ID	Age	Annual Income (k$)	Spending Score (1-100)	clusters
count	200.000000	200.000000	200.000000	200.000000	200.000000
mean	100.500000	38.850000	60.560000	50.200000	2.190000
std	57.879185	13.969007	26.264721	25.823522	1.208804
min	1.000000	18.000000	15.000000	1.000000	0.000000
25%	50.750000	28.750000	41.500000	34.750000	1.000000
50%	100.500000	36.000000	61.500000	50.000000	3.000000
75%	150.250000	49.000000	78.000000	73.000000	3.000000
max	200.000000	70.000000	137.000000	99.000000	4.000000

➢ **graphical representations about the dataset**



5. Applicable constraints:

- You can save time and money by using machine learning algorithms to perform customer segmentation and eliminate manual work. Your algorithms need to solve a real business problem to be effective.
- This process, if done in-house, can cost anywhere between $60,000 to $95,000 over five years. This includes model infrastructure, data support, and engineering/deployment. If you add additional models, this price grows quickly.
- Using a ready-made machine learning algorithm can get you potentially better results at a fraction of the cost compared to developing a custom machine learning system.
- If the customer segmentation is done manually it requires more space for the storage of the model.

6. Business Model (Monetization Idea):

In order to run a successful promotion the E-commerce companies first need to understand the customer. Once the customers in different segments are identified the company can run targeted promotions on specific customer segment. In order to create customer segments we used this clustering approach which determines the clusters based on our analysis.

7. Implementing the Model:

- For Predicting Customer Segmentation we use **K-means clustering Algorithm.**
- K-means clustering is an algorithm where the points of each cluster group are as similar as possible, and points in different clusters are as dissimilar as possible.
- **K-Means clustering** is an efficient machine learning algorithm to solve data clustering problems. It's an unsupervised algorithm that's quite suitable for solving customer segmentation problems.
- This algorithm is used when we have unlabelled data. Unlabelled data means input data without categories or groups provided.

➢ Defining the model

```
# Define K-means model
kmeans_model = KMeans(init='k-means++',  max_iter=400, random_state=42)
```

• Training the model

```
# Train the model
kmeans_model.fit(customersdata[['Age','Annual Income (k$)','Spending Score (1-100)']])
```

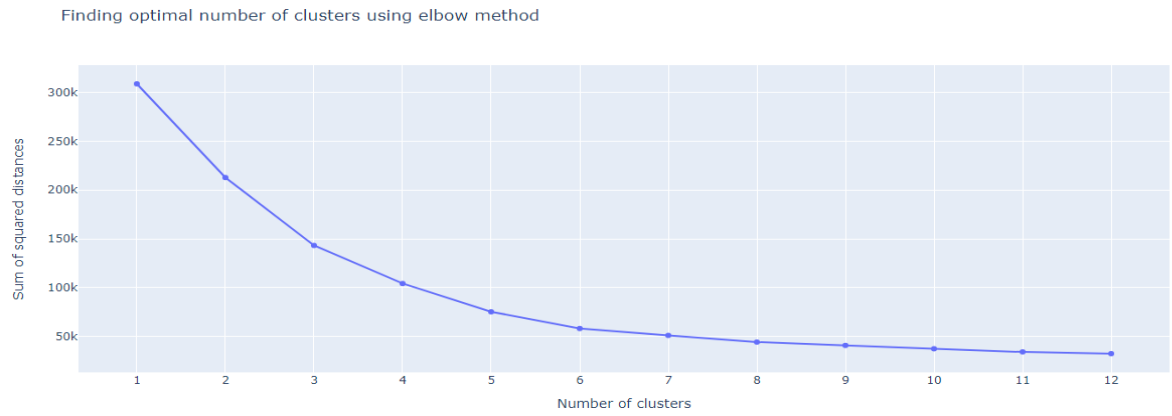
• Creating the K-means model

```python
# Create the K means model for different values of K
def try_different_clusters(K, data):

    cluster_values = list(range(1, K+1))
    inertias=[]

    for c in cluster_values:
        model = KMeans(n_clusters = c,init='k-means++',max_iter=400,random_state=42)
        model.fit(data)
        inertias.append(model.inertia_)

    return inertias
```

- # Finding optimal number of clusters

we use **elbow method** for finding the optimal number of clusters

```python
# Finding optimal number of clusters k
figure = go.Figure()
figure.add_trace(go.Scatter(x=distances["clusters"], y=distances["sum of squared distances"]))

figure.update_layout(xaxis = dict(tick0 = 1,dtick = 1,tickmode = 'linear'),
                     xaxis_title="Number of clusters",
                     yaxis_title="Sum of squared distances",
                     title_text="Finding optimal number of clusters using elbow method")
figure.show()
```



- **Re-Training the model with 5 clusters**

```
[10] # Re-Train K means model with k=5
     kmeans_model_new = KMeans(n_clusters = 5,init='k-means++',max_iter=400,random_state=42)

     kmeans_model_new.fit_predict(customersdata[['Age','Annual Income (k$)','Spending Score (1-100)']])
```

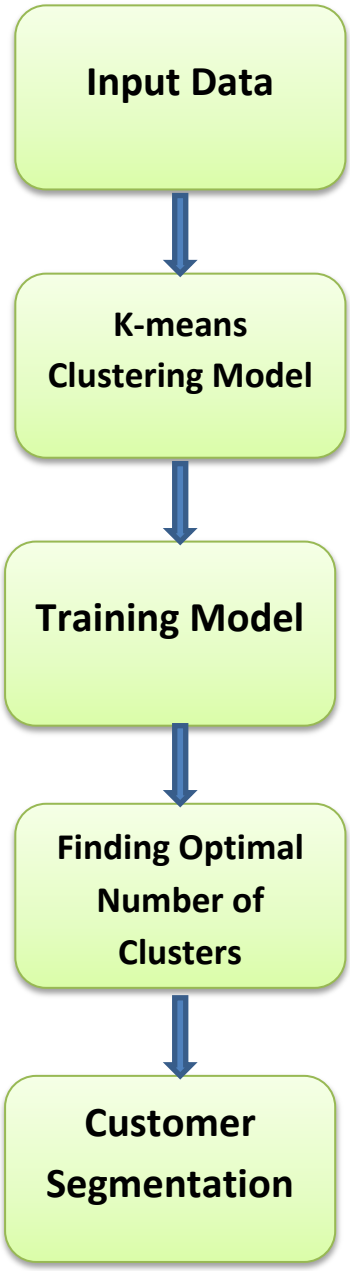
```
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning:

The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress the warning

array([0, 4, 0, 4, 0, 4, 0, 4, 0, 4, 0, 4, 0, 4, 0, 4, 0, 4, 0, 4, 0, 4,
       0, 4, 0, 4, 0, 4, 0, 4, 0, 4, 0, 4, 0, 4, 0, 4, 0, 4, 0, 4, 0, 3,
       0, 4, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3,
       3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3,
       3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3,
       3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 1, 2, 1, 3, 1, 2, 1, 2, 1,
       2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1,
       2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1,
       2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1,
       2, 1], dtype=int32)
```

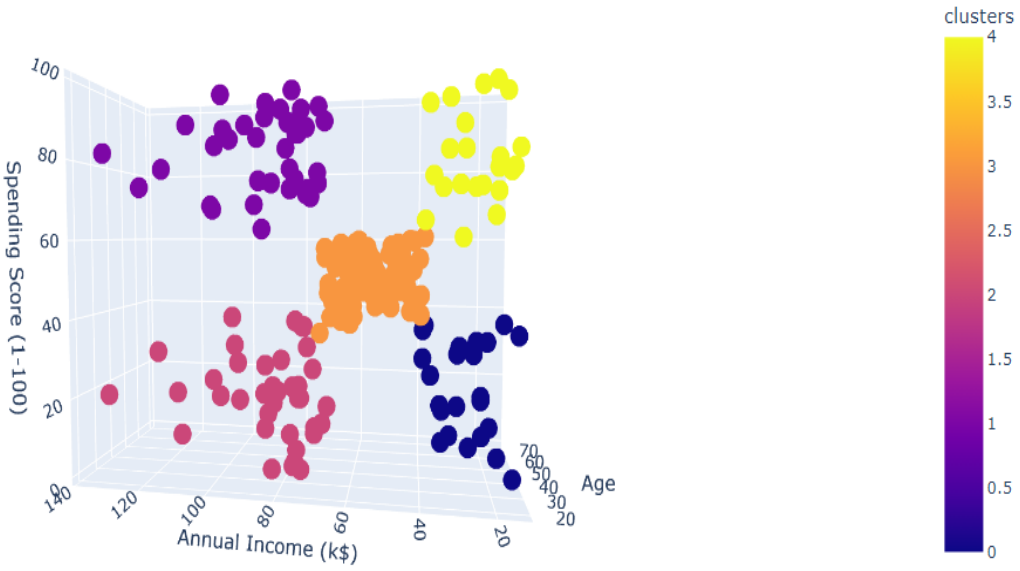
- **Visualizing the clusters**

```
# visualize clusters
figure = px.scatter_3d(customersdata,
                       color='clusters',
                       x="Age",
                       y="Annual Income (k$)",
                       z="Spending Score (1-100)",
                       # category_orders = {"clusters": ["0", "1", "2", "3", "4"]}
                       )
figure.update_layout()
figure.show()
```

8. Final Product Prototype :

Schematic Diagram For Customer Segmentation





- **From the above visualization we can see that Mall Customers is broadly divided into 5 Groups.**
  1) **Clusters 1- Purple**
  2) **Clusters 2- Pink**
  3) **Clusters 3- Orange**
  4) **Clusters 4- Yellow**
  5) **Clusters 5- Blue**

➢ Benefits of customer segmentation:

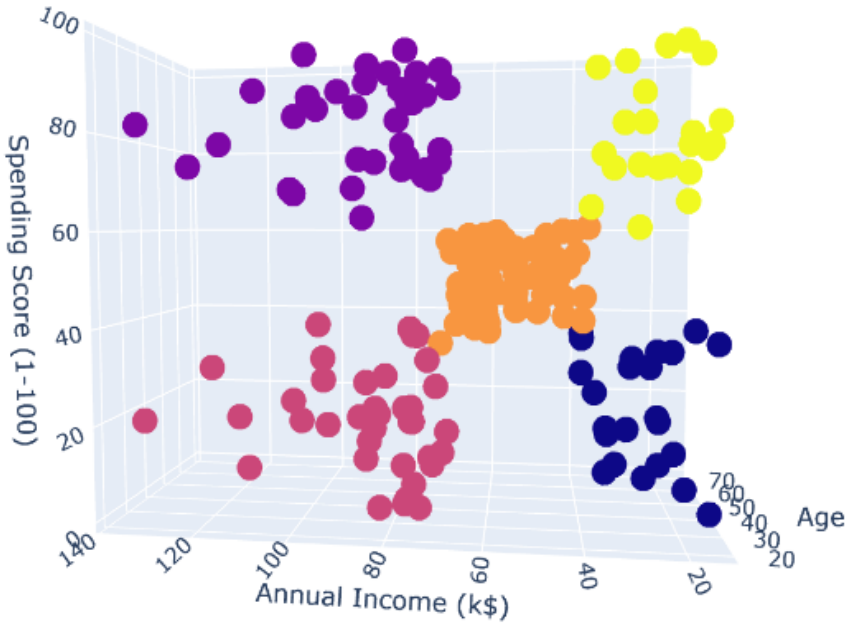
- By enabling companies to target specific groups of customers, a customer segmentation model allows for the effective allocation of marketing resources and the maximization of cross- and up-selling opportunities.
- Other benefits of customer segmentation include staying a step ahead of competitors in specific sections of the market and identifying new products that existing or potential customers could be interested in or improving products to meet customer expectations.

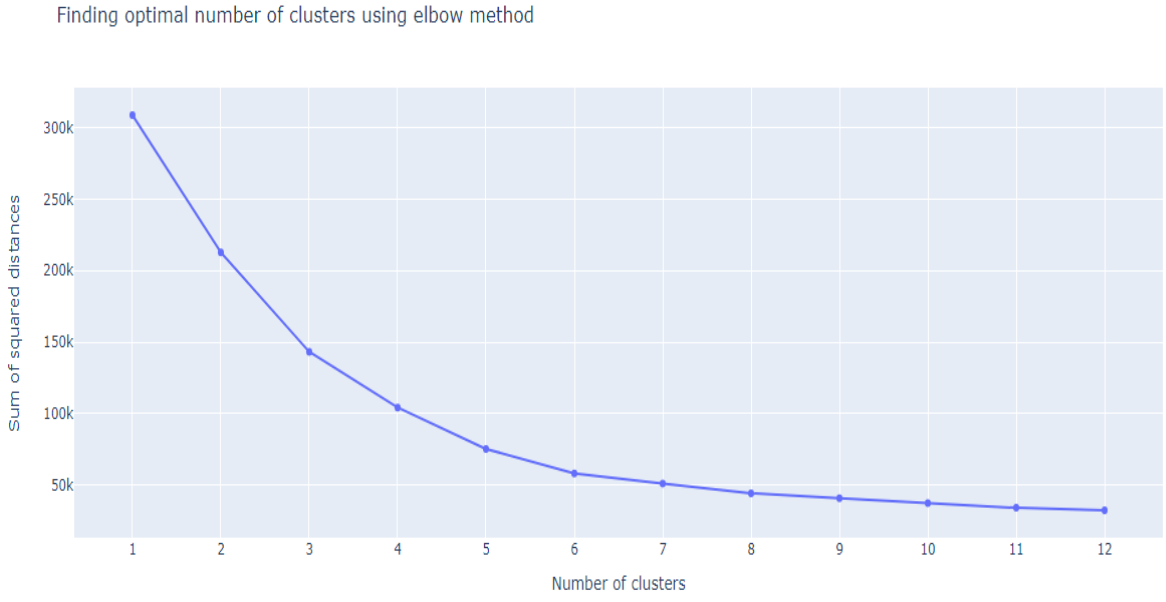
9. Code Implementation/Validation on Small Scale:

- **Github Link To Code Implementation-**

https://github.com/VANAPALLI-DIMPLE-SATYA-DEEPAK/customer-segmentation/blob/main/Prediting%20Customer%20Segmentation%20Final.ipynb

- **Some Basic Visualizations**





- In the code implementation used models are **K-means clustering** and for finding optimal number of clusters **Elbow method** is used.

10. Conclusion:

In This Project, I Performed Unsupervised Clustering. I Used **K-Means Clustering** And I Came Up With **5 Groups Of Clusters**. This Can Be Used In Planning Better Marketing Strategies**.**