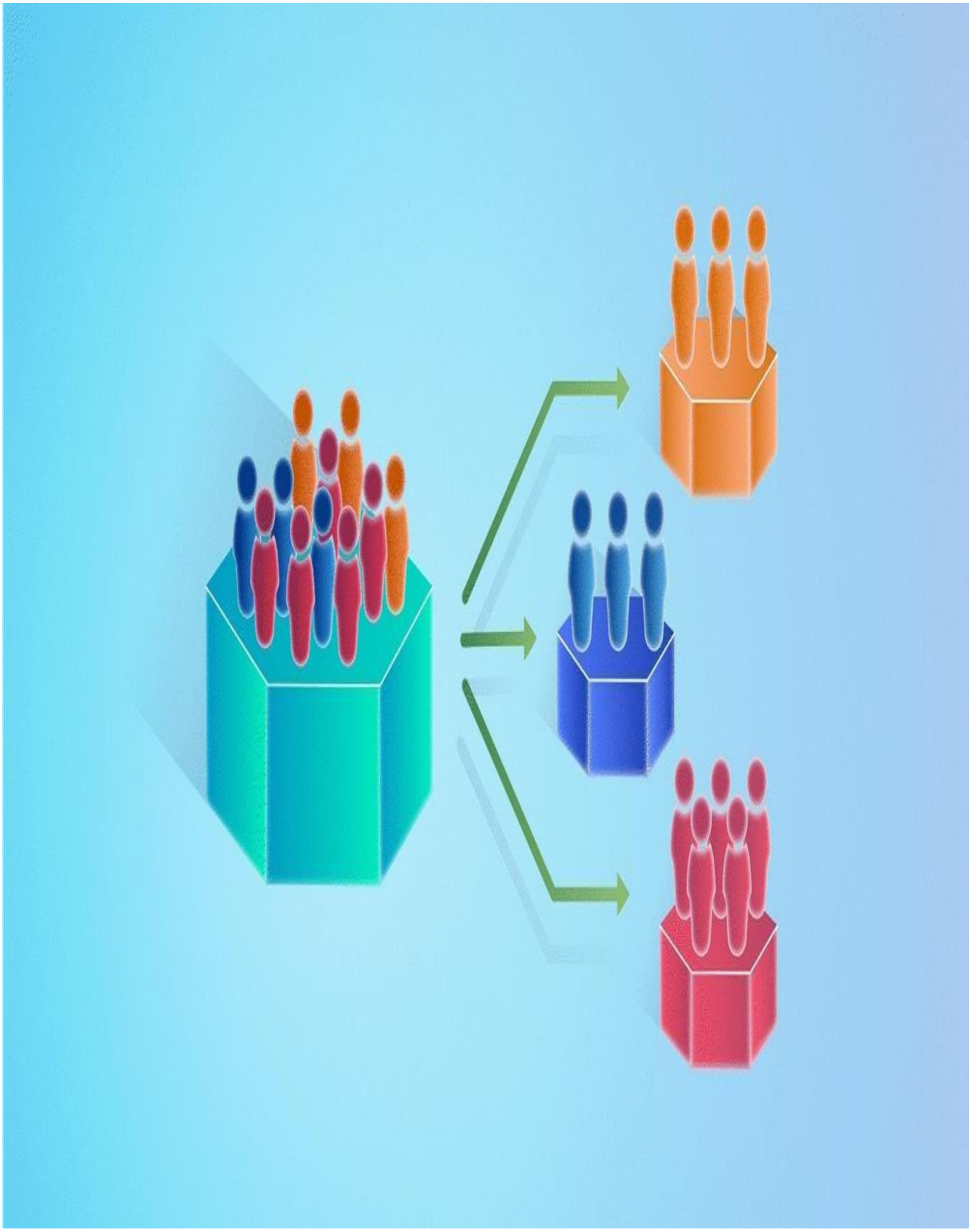
CUSTOMER SEGMENTATION USING DATA SCIENCE

Phase 5 submission Documents

Name:N.Moulika

Reg.no:712221104007

College: park college of Engineering And Technology



PHASE1

Problem Definition :

*The problem is to implement data science techniques to segment customers based on their behavior, preferences, and demographic attributes. The goal is to enable businesses to personalize marketing strategies and enhance customer satisfaction. This project involves data collection, data preprocessing, feature engineering, clustering algorithms, visualization, and interpretation of results.

Design Thinking

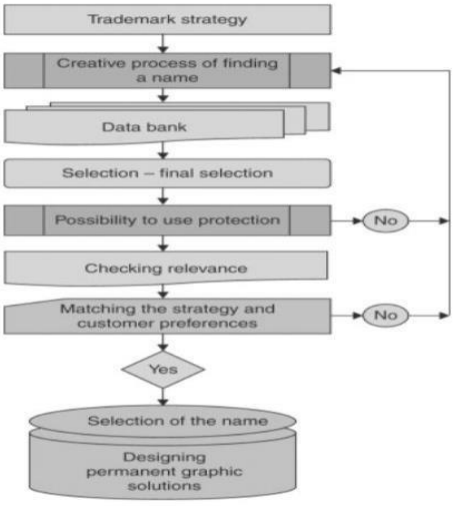
- Data Collection: Collect customer data, including attributes like purchase history, demographic information, and interaction behavior.
- Data Preprocessing: Clean and preprocess the data, handle missing values, and convert categorical features into numerical representations.
- DesignClustering Algorithms: Apply clustering algorithms like K -Means, DBSCAN, or hierarchical clustering to segment customers.

Data set:

https://www.kaggle.com/datasets/akram24/mall-customersClustering

phase2

FLOWCHART:



Overview:

Problem Definition

Data set

Flow chart

Text Act

Phase3

Introduction:

- Customer segmentation using data science is a powerful technique that involves dividing a company's customer base into distinct groups based on specific characteristics, behaviors, or demographics.

- By doing so, businesses can gain valuable insights into their customers, allowing them to tailor their marketing strategies, products, and services to meet the unique needs of each segment.

- Data science techniques, such as clustering algorithms, machine learning models, and data mining, are employed to analyze large sets of customer data.

- These methods identify patterns, trends, and relationships within the data, enabling businesses to create meaningful segments.

GIVEN DATASET:

	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

Necessary step to follow:

1.Import libraries:

Start by importing the necessary libraries:

Program:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

2. Load the Dataset:

Load your dataset into a pandas dataframe. You Can typically find customer segmentation using datascience dataset in CSV format,but you can adapt code to other formats as needed.

Program:

```
df=pd.read_csv('/kaggle/input/mall-customers/Mall_Customers.csv')

df.rename(columns={'Genre':'Gender'},inplace=True)
df.head()
```

Df.describe()

```
df.isnull().sum()
```

```
CustomerID                0
Gender                    0
Age                       0
Annual Income (k$)        0
Spending Score (1-100)    0
dtype: int64
```

challenge involved in loading and preprocessing a customer segmentation using datascience dataset:

Data Quality:

- Incomplete or missing data: You may encounter missing values in your dataset, and deciding how to handle them (imputation or removal) is critical.
- Outliers: Identifying and dealing with outliers that could skew your segmentation analysis is important.

Data Cleaning:

- Data may contain inconsistencies, errors, and duplicates that need to be addressed.
- Standardizing and normalizing data, especially for categorical variables, is necessary.

How to overcome the challenge involved in loading and preprocessing customer segmentation using datascience dataset:

Data Integration:

- Create a comprehensive data integration plan to merge data from different sources. Ensure that all data is consistent in terms of format and units.

Data Scaling and Transformation:

- Scale numerical features to ensure that they have equal weight in the segmentation process.
- Apply necessary transformations, such as logarithmic transformations, to make data more suitable for clustering.

Loading the Dataset:

- **Data Exploration:** After loading the dataset, it's a good practice to explore the data to understand its structure and the information it contains. You can use functions like **`head()`**, **`info()`**, and **`describe()`** to get an initial overview of the data.
- **Load the Dataset:** You can load your dataset from various sources like CSV files, Excel files, or databases.

Program:

```
df=pd.read_csv('/kaggle/input/mall-customers/Mall_Customers.csv')

df.rename(columns={'Genre':'Gender'},inplace=True)
df.head()
```

```
df.describe()
```

Loading the Dataset:

Output:

	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

Preprocessing the Dataset:

- **Handling Missing Values:** Check for missing data in your dataset and decide on an appropriate strategy for handling them. You can either fill in missing values with a specific value (e.g., mean, median, or mode) or remove rows or columns with too many missing values.

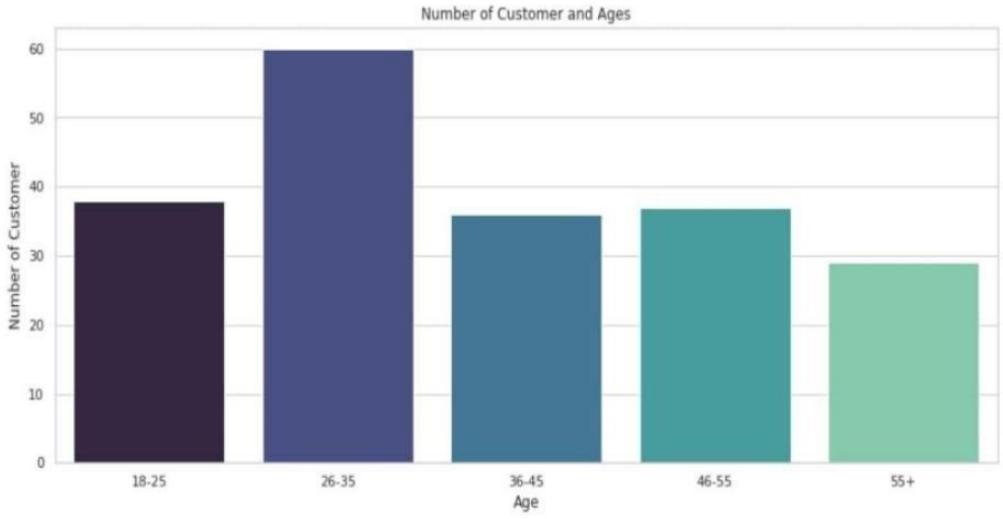
- **Feature Scaling:** Depending on the algorithms you plan to use for segmentation, it might be necessary to scale or normalize your numerical features to have a consistent scale.

Virtualization and preprocessing of data:

In[1]:

```
age_18_25 = df.Age[(df.Age >=18) & (df.Age
<= 25)]
age_26_35 = df.Age[(df.Age >=26) & (df.Age
<= 35)]
age_36_45 = df.Age[(df.Age >=36) & (df.Age
<= 45)]
age_46_55 = df.Age[(df.Age >=46) & (df.Age
<= 55)]
age_55_above = df.Age[(df.Age >= 56)]

age_x =["18-25","26-35","36-45","46-55","5
5+"]
age_y = [len(age_18_25.values),len(age_26_
35.values),len(age_36_45),len(age_46_55),l
en(age_55_above)]

plt.figure(figsize = (15,6))
sns.barplot(x=age_x, y=age_y,palette = "ma
ko")
plt.title("Number of Customer and Ages")
plt.xlabel("Age")
plt.ylabel("Number of Customer")
plt.show()
```

Out[1]:

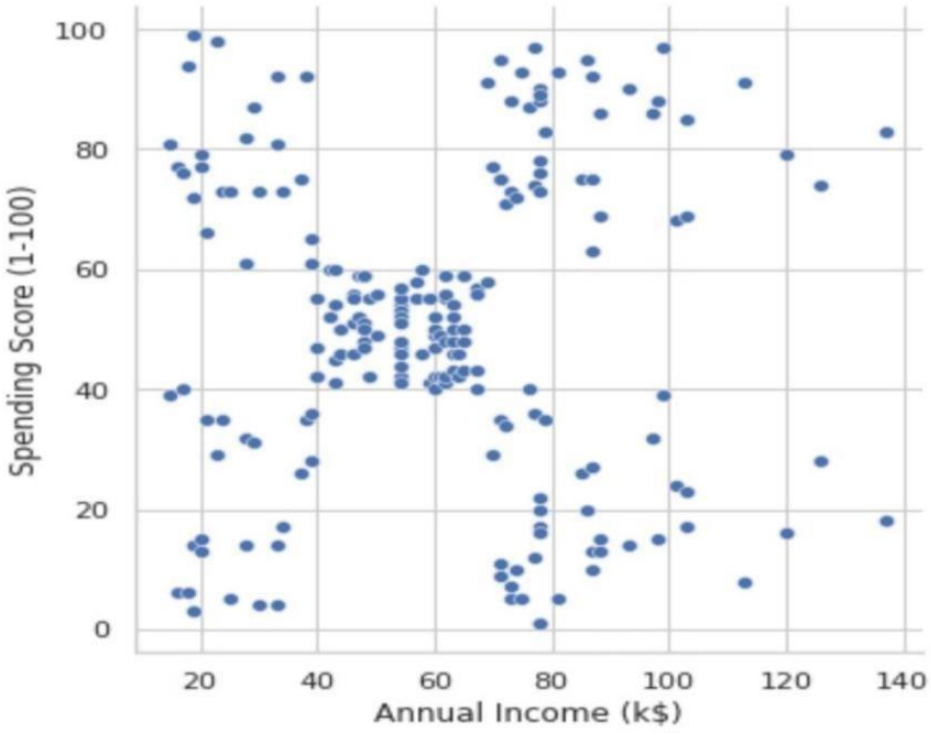


In[2]:

```
sns.relplot(x="Annual Income (k$)",y = "Sp
ending Score (1-100)",data=df)
```

Out[2]:

```
<seaborn.axisgrid.FacetGrid at 0x7fcef053
6fd0>
```



Phase4

FEATURE ENGINEERING:

- Feature engineering plays a crucial role in customer segmentation using data science. It involves selecting, transforming, or creating relevant features (variables) from raw data to improve the performance of machine learning models and enhance the accuracy of customer segmentation. Here are some key techniques and considerations for feature engineering in customer segmentation:

  - Domain Knowledge: Understand the business domain and customer behavior to identify meaningful features. Domain experts can provide valuable insights into which variables might be relevant for segmentation.

  - Demographic Features:Utilize demographic information such as age, gender, income, education, and marital status. These features can provide a foundation for understanding customer characteristics.

- Geographic Features:Include geographic data like location, city, region, or climate, especially if your business targets specific geographical areas.

- Behavioral Features:Incorporate customer behavior data such as purchase history, frequency of purchases, average transaction value, browsing patterns, and interactions with products or services.

- Recency, Frequency, Monetary (RFM) Analysis: RFM analysis quantifies customer behavior by evaluating how recently a customer made a purchase, how frequently they make purchases, and how much money they spend. These metrics can be powerful features for segmentation.

APPLYING CLUSTERING ALGORITHM:

- ❖ Applying clustering algorithms is a common technique in customer segmentation using data science. Clustering algorithms group similar data points together based on specific features, allowing businesses to identify distinct customer segments. Here are the steps to apply clustering algorithms for customer segmentation:
- ❖ Gather and preprocess relevant customer data, ensuring it is clean, consistent, and properly formatted.

- Normalize or scale the data to bring features to a similar scale, especially if you're using distance-based clustering algorithms like k-means.

- ❖ Selecting a Clustering Algorithm:

- Choose an appropriate clustering algorithm based on the nature of your data and business requirements. Common clustering algorithms include k-means, hierarchical clustering, DBSCAN, and Gaussian mixture models.

- ❖ Determining the Number of Clusters (k):

- For algorithms like k-means, you need to specify the number of clusters (k). Use techniques like the elbow method or silhouette score to find the optimal number of clusters that best represent the data's structure.

VISUALIZATION:

Visualizing customer segmentation results is essential for understanding the identified clusters and communicating the insights to stakeholders

effectively. Here are several visualization techniques commonly used in data science for visualizing customer segmentation:

- ❖ Scatter Plots:

- Create scatter plots for two selected features to visualize the clusters. Each point represents a customer, and the clusters can be distinguished by different colors or markers.

- ❖ Parallel Coordinates:

- Use parallel coordinates to visualize multivariate data in a comprehensible way. Each axis represents a feature, and lines connecting points across different axes can help identify patterns and differences between clusters.

❖ 3D Scatter Plots:

- If your data has more than two features, consider creating 3D scatter plots to visualize clusters in a three-dimensional space. This approach provides a more detailed view of the data distribution.

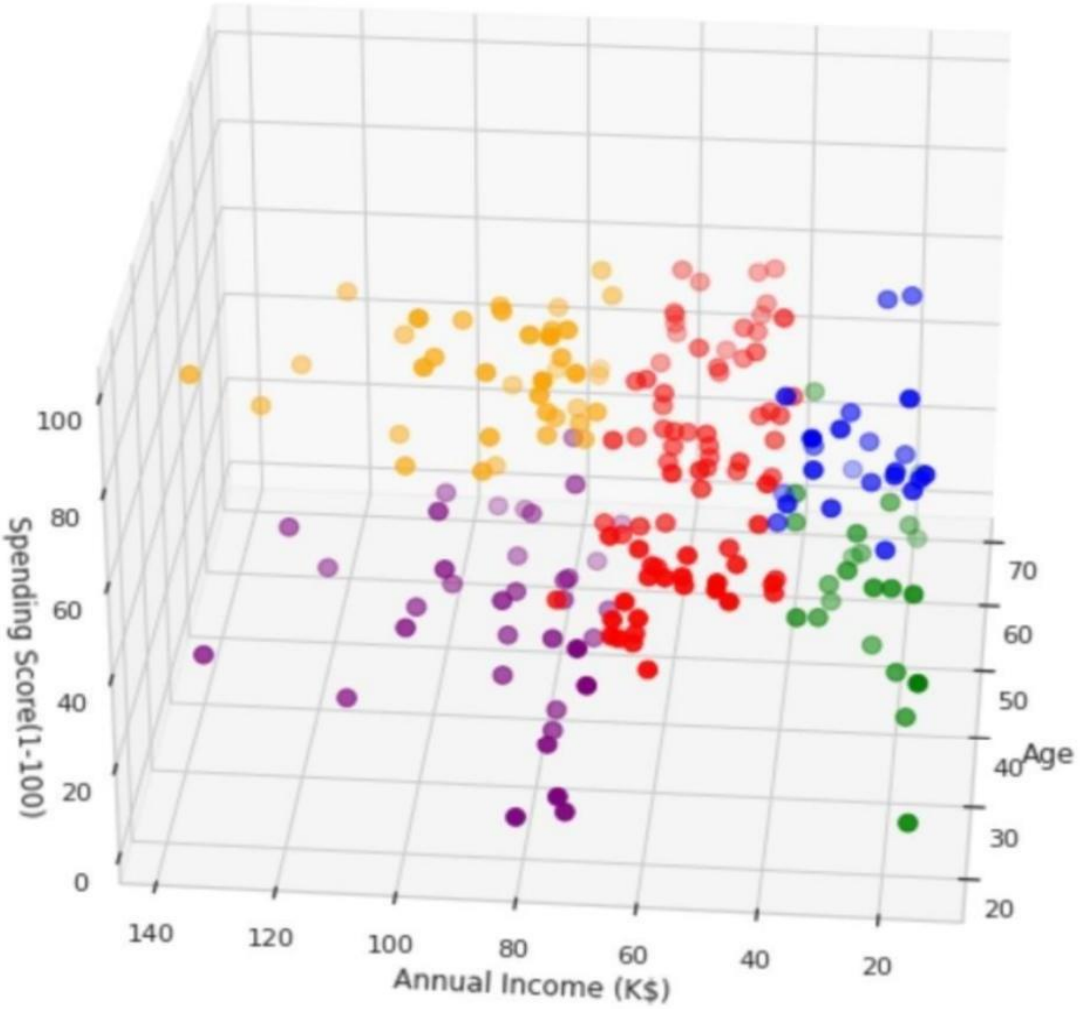
❖ Geospatial Visualizations:

- If your data includes geographic information, use maps to visualize clusters based on geographical locations. Geospatial visualizations provide insights into regional customer behavior patterns.

PROGRAM:

```
cluster = kmeans.fit_predict(X3)
df["label"] = cluster


from mpl_toolkits.mplot3d import Axes3D


fig = plt.figure(figsize=(20,10))
ax = fig.add_subplot(111,projection = '3d')

ax.scatter(df.Age[df.label == 0 ],df["Annual Income (k$)"][df.label == 0],df["Spending
Score (1-100)"][df.label == 0], c = 'blue',s=60)
ax.scatter(df.Age[df.label == 1 ],df["Annual Income (k$)"][df.label == 1],df["Spending
Score (1-100)"][df.label == 1], c = 'red',s=60)
ax.scatter(df.Age[df.label == 2 ],df["Annual Income (k$)"][df.label == 2],df["Spending
Score (1-100)"][df.label == 2], c = 'green',s=60)
ax.scatter(df.Age[df.label == 3 ],df["Annual Income (k$)"][df.label == 3],df["Spending
Score (1-100)"][df.label == 3], c = 'orange',s=60)
ax.scatter(df.Age[df.label == 4 ],df["Annual Income (k$)"][df.label == 4],df["Spending
Score (1-100)"][df.label == 4], c = 'purple',s=60)


ax.view_init(30,185)


plt.xlabel("Age")
plt.ylabel("Annual Income (K$)")
ax.set_zlabel('Spending Score(1-100)')



plt.show()
```

OUTPUT:


Spending Score(1-100)
100
80
60
40
20
0
140
120
100
80
60
40
20
Annual Income (K$)
70
60
50
40
30
20
Age

INTERPRETATION:

Customer segmentation in data science refers to the process of dividing a company's customer base into distinct groups based on specific characteristics such as demographics, behavior, or purchasing patterns. This segmentation allows businesses to better understand their customers, target their marketing efforts, and enhance overall customer satisfaction. Data science techniques play a crucial role in this process by analyzing large datasets and extracting meaningful insights.

Here's how data science is typically applied to interpret customer segmentation:

- Data Collection:Gather relevant data from various sources, such as customer profiles, transaction history, website interactions, and social media activity.

- Data Preprocessing: Cleanse and preprocess the data to handle missing values, outliers, and inconsistencies. This step ensures that the data used for segmentation is accurate and reliable.

- Segmentation Techniques:Apply various data science techniques such as clustering algorithms (like k-means clustering), decision trees, or neural networks to segment customers based on the selected features.

- These algorithms group customers with similar characteristics together.

- Interpretation: Interpret the results obtained from the segmentation models. Understand the characteristics of each segment, such as their preferences, behaviors, and needs. This interpretation helps in creating targeted marketing strategies for each segment.

- Validation:Validate the segmentation results to ensure their accuracy and reliability. This step might involve using different validation metrics depending on the technique used.

- Implementation: Implement the insights gained from customer segmentation into business strategies. This could involve personalized marketing campaigns, product

- recommendations, or tailored customer experiences for each segment.

- Monitoring and Iteration:Continuously monitor customer behavior and iteratively refine segmentation models as new data becomes available. Customer preferences and behaviors can change over time, so it’s essential to adapt strategies accordingly.