CUSTOME SEGMENTATION WITH DATA SCIENCE

PHASE 1-DOCUMENT SUBMISSION

PROBLEM DEFINITION:

In today's competitive market, it is essential for businesses to understand and target customers with similar characteristics and behaviors. Aims to improve business strategy, improve customer service and increase profitability. Customer segmentation is a data-based method that divides a company's customers into different groups based on their characteristics or behavior, allowing for targeted advertising, business and products or services.

The goal of this data exploration is to create powerful customer solutions that enable companies to make informed decisions and align business processes, copy, sales and customer experience around customer experience.

DESIGN THINKING:

STEP 1- DATA COLLECTION AND GATHERING:

Collecting relevant data from various sources such as kaggle. Ensuring data quality and accuracy by addressing missing values, outliers and inconsistencies.

I. DATA SOURCE:

Dataset link:
(https://www.kaggle.com/datasets/vedavyasv/usa)

1	CustomerI	Genre	Age	Annual Inc	Spending Score (1-10
2	1	Male	19	15	39
3	2	Male	21	15	81
4	3	Female	20	16	6
5	4	Female	23	16	77
6	5	Female	31	17	40
7	6	Female	22	17	76
8	7	Female	35	18	6
9	8	Female	23	18	94
10	9	Male	64	19	3
11	10	Female	30	19	72
12	11	Male	67	19	14
13	12	Female	35	19	99
14	13	Female	58	20	15
15	14	Female	24	20	77
16	15	Male	37	20	13
17	16	Male	22	20	79
18	17	Female	35	21	35
19	18	Male	20	21	66
20	19	Male	52	23	29
21	20	Female	35	23	98
22	21	Male	35	24	35
23	22	Male	25	24	73
24	23	Female	46	25	5
25	24	Male	31	25	73
26	25	Female	54	28	14
27	26	Male	29	28	82

STEP 2- DATA PREPROCESSING:

Cleaning and preprocessing the data to make it suitable for analysis. Encoding categorical variables using techniques like one-hot encoding or label encoding.

CODE:

```
#import necessary libraries

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

#dataset = pd.read_csv('/kaggle/input/mall-customers/Mall_Customers.csv')

#dataset.head()

#dataset.shape

#dataset.info()
```

#dataset.isnull().sum()

#x=dataset.iloc[:,[3,4]].values

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

#(200, 5)

#<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 5 columns):

s.no	Column	Non-Null Count	Dtype
0	CustomerID	200 non-null	int64
1	Genre	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

dtypes: int64(4), object(1)
memory usage: 7.9+ KB

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

STEP 3- FEATURE ENGINEERING:

Creating meaningful features that can help in customer segmentation such as customer lifetime value, purchase frequency, etc.

STEP 4- EXPLORATORY DATA ANALYSIS (EDA):

Conducting exploratory data analysis to gain insights into the data. Visualizing and summarizing key statistics and trends.

STEP 5- MODEL SELECTION:

Choosing an appropriate segmentation technique or algorithm.

Using **K-MEANS CLUSTERING** algorithm for this project.

STEP 6- MODEL TRAINING:

Training the selected segmentation model on the preprocessed data.

CODE:

```
#from sklearn.cluster import KMeans

# wcss=[]
for i in range(1,11):
   kmeans=KMeans(n_clusters = i , init="k-
means++",random_state=0)
   kmeans.fit(x)
   wcss.append(kmeans.inertia_)
```

OUTPUT:

```
/opt/conda/lib/python3.10/site-
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
  warnings.warn(
/opt/conda/lib/python3.10/site-
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
  warnings.warn(
```

```
/opt/conda/lib/python3.10/site-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
  warnings.warn(
/opt/conda/lib/python3.10/site-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
  warnings.warn(
/opt/conda/lib/python3.10/site-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
  warnings.warn(
/opt/conda/lib/python3.10/site-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
  warnings.warn(
/opt/conda/lib/python3.10/site-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
  warnings.warn(
```

```
/opt/conda/lib/python3.10/site-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
warnings.warn(
/opt/conda/lib/python3.10/site-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
  warnings.warn(
/opt/conda/lib/python3.10/site-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
  warnings.warn(
```

STEP 7- MODEL EVALUATION:

Analyzing the segments created and interpreting the characteristics and behaviors of each segments.

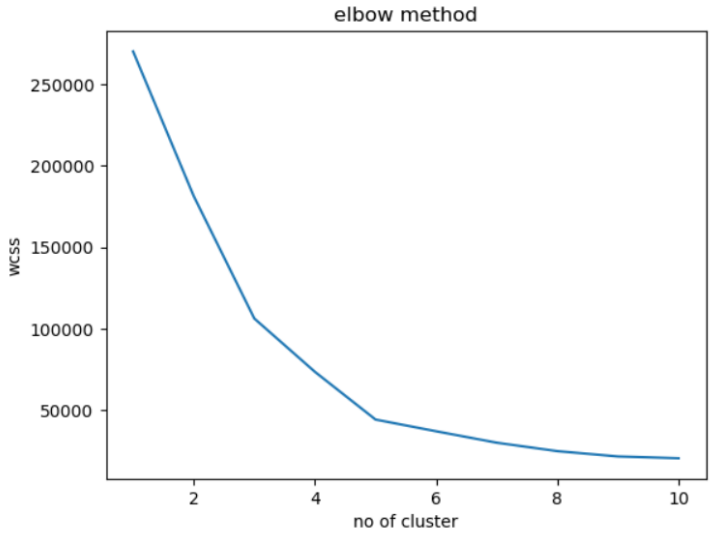
STEP 8- DOCUMENTATION:

Documenting the entire data science process, including data sources, preprocessing steps, model selection and validation methods.

VISUALIZATION:

CODE:

```
plt.plot(range(1,11),wcss)
plt.title("elbow method")
plt.xlabel("no of cluster")
plt.ylabel("wcss")
plt.show()
```



CODE:

```
plt.scatter(x[y_kmeans==0,0],x[y_kmeans==0,1],s=100,c="red",l
abel = "cluster 1")
plt.scatter(x[y_kmeans==1,0],x[y_kmeans==1,1],s=100,c="blue"
,label = "cluster 2")
plt.scatter(x[y_kmeans==2,0],x[y_kmeans==2,1],s=100,c="gree
n",label = "cluster 3")
plt.scatter(x[y_kmeans==3,0],x[y_kmeans==3,1],s=100,c="cyan
",label = "cluster 4")
```

```
plt.scatter(x[y_kmeans==4,0],x[y_kmeans==4,1],s=100,c="magenta",label = "cluster 5")
plt.scatter(kmeans.cluster_centers_[:,0],kmeans.cluster_centers_[:,1],s=300,c="yellow",label="centroids")
plt.title("clusters of customers")
plt.xlabel("Yıllık gelir")
plt.ylabel=("harcama skoru")
plt.legend()
plt.show()
```

OUTPUT: