DTSA_5510_UNSUPERVISED_LEARNING_FINAL_PROJECT

PRESENTATION HEADLINES

- Problem Definition
- KMeans Clustering
- Conclusions & Results

Problem Definition

The problem at hand revolves around understanding the customer base of a supermarket mall and identifying the target customers for effective marketing strategies. The dataset provides information such as Customer ID, age, gender, annual income, and spending score. The spending score is a metric assigned to each customer based on their purchasing behavior.

Problem Definition

The objective is to utilize machine learning techniques, specifically the KMeans Clustering algorithm, to segment the customers and determine the easily converging target customers. This segmentation will enable the marketing team to tailor their strategies and effectively engage with the identified customer groups. By addressing this problem, we aim to gain insights into customer segmentation, execute a simplified implementation in Python, and explore the practical applications of marketing strategies in the real world.

About the Data

The data has 200 observations and 5 columns such that 4 features are numerical and we have 1 feature that is categorical data.

```
In [3]: df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 5 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   CustomerID              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
dtypes: int64(4), object(1)
memory usage: 7.9+ KB

In [4]: df.shape

Out[4]: (200, 5)
```

EDA and Data Cleaning

In [8]:
```
# let's drop the CustomerID columns
df = df.drop(['CustomerID'], axis=1)
```

In [9]:
```
# let's check missing values
df.isna().sum().sum()
```

Out[9]: 0

In [10]:
```
# let's check duplicated rows
df.duplicated().sum()
```

Out[10]: 0

EDA and Data Cleaning

In [13]:

```
# Perform one-hot encoding on the gender feature
df_encoded = pd.get_dummies(df, columns=['Gender'], drop_first=True)
```

KMeans Clustering - PCA

Dimensionality Reduction

In [18]:

```
from sklearn.decomposition import PCA

# Perform PCA
pca = PCA(n_components=2)
pca_features = pca.fit_transform(df[['Age', 'Annual Income (k$)', 'Spending Score (1-100)']])
df_pca = pd.DataFrame(data=pca_features, columns=['PC1', 'PC2'])
```

Utilizing PCA for dimensionality reduction in this KMeans project offers several advantages, including improved computational efficiency, enhanced visualization, addressing multicollinearity, and facilitating feature selection and interpretation. By leveraging the power of PCA, data scientists and analysts can streamline the clustering process, gain valuable insights, and make informed decisions based on the reduced-dimensional representation of the data.

KMeans Clustering – Elbow Method

Elbow Method

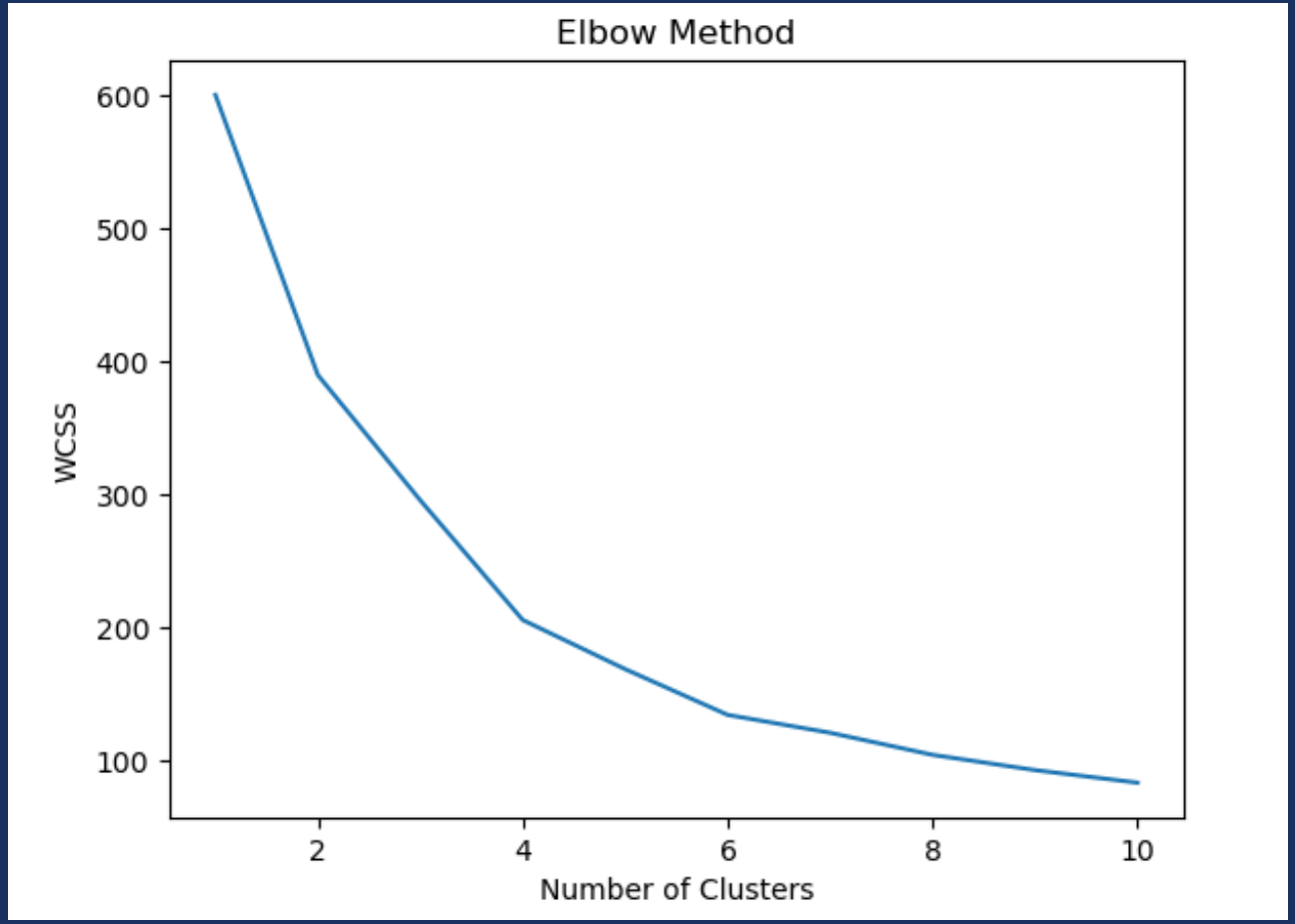
In [19]:

```
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt

# Calculate within-cluster sum of squares (WCSS) for different number of clusters
wcss = []
for i in range(1, 11):
    kmeans = KMeans(n_clusters=i, random_state=42)
    kmeans.fit(df[['Age', 'Annual Income (k$)', 'Spending Score (1-100)']])
    wcss.append(kmeans.inertia_)

# Plot WCSS against number of clusters
plt.plot(range(1, 11), wcss)
plt.xlabel('Number of Clusters')
plt.ylabel('WCSS')
plt.title('Elbow Method')
plt.show()
```

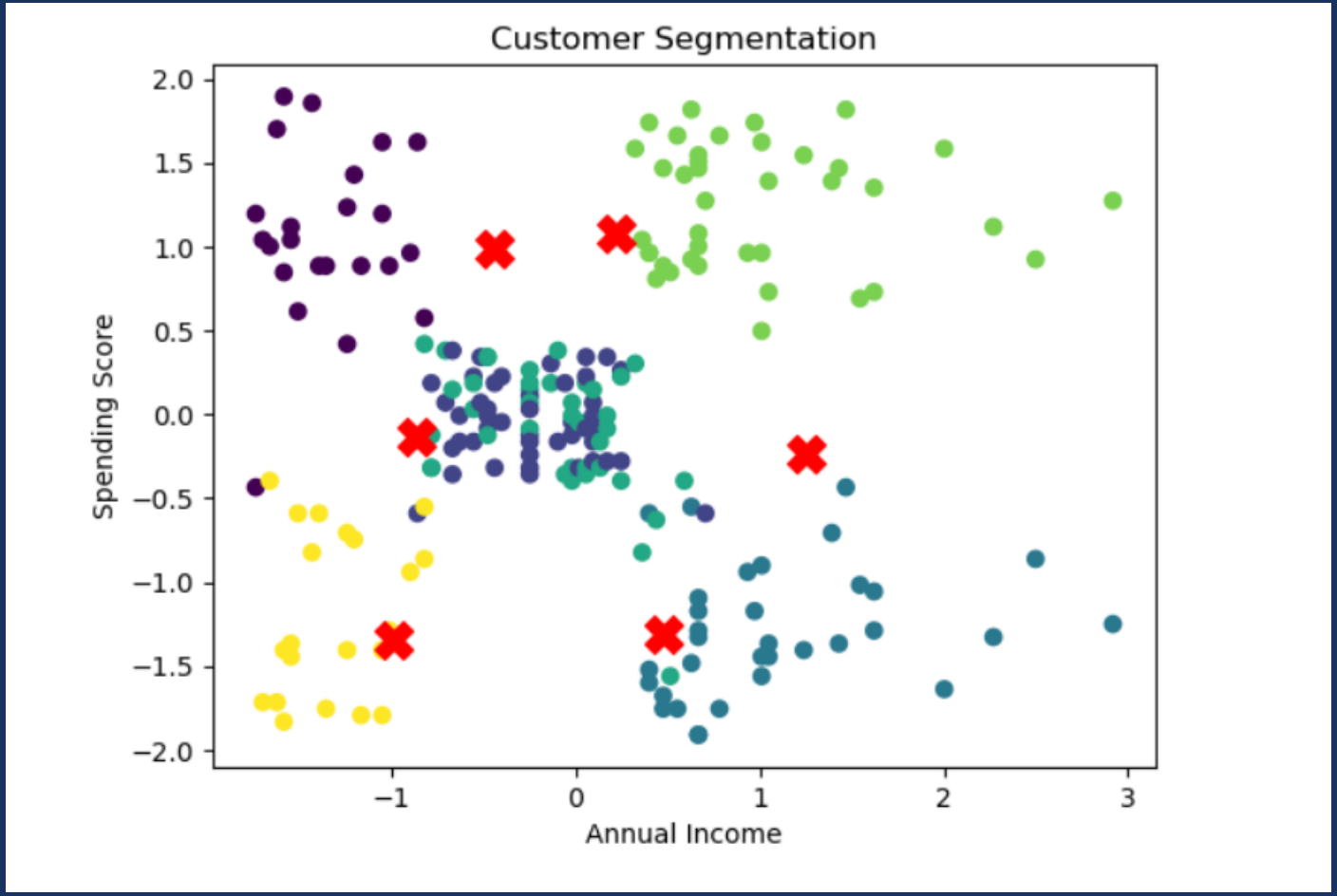
KMeans Clustering – Elbow Method

The Elbow Method is a valuable technique for determining the optimal number of clusters in a KMeans project. By analyzing the WCSS values and visually inspecting the resulting plot, data analysts can make informed decisions about the appropriate number of clusters to use. This method enhances the quality of clustering results, facilitates algorithm parameter tuning, and provides stakeholders with a clear and concise understanding of the clustering structure within the dataset.

KMeans Clustering – Elbow Method



KMeans Clustering – Customer Segmentation



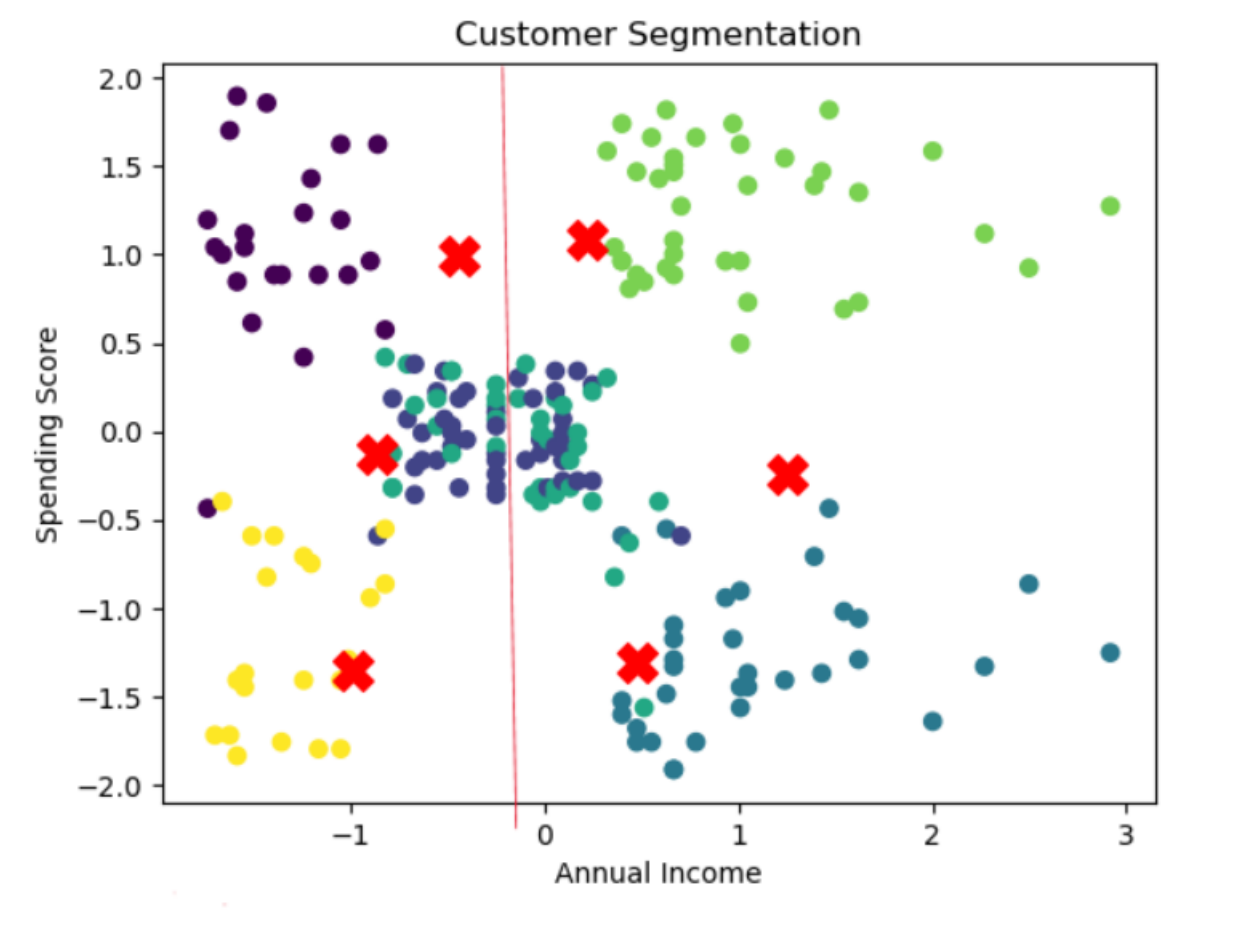
Conclusion and Results

Level 1: Two clusters with negative spending score values.

Level 2: Two clusters with neutral spending score values (near zero).

Level 3: Two clusters with positive spending score values.

Conclusion and Results



The left level has 3 clusters, all with low annual income but different spending scores. The right level also has 3 clusters, all with similar high annual income but different spending scores.

THANK YOU