Project Report on

Customer Segmentation using Unsupervised Learning For

Introduction to AI (AI101B)

By

Team InnoLearners

Aanchal – 202410116100002

Devanshi Singhal- 202410116100060

Deepanshu Ruhela – 202410116100056

Dhwani Panchal -202410116100063

Session:2024-2025 (Semester II)

Under the supervision of

MR. APOORV JAIN (Assistant Professor)

KIET Group of Institutions, Delhi-NCR, Ghaziabad

DEPARTMENT OF COMPUTER APPLICATIONS

KIET GROUP OF INSTITUTIONS,

DELHI-NCR, GHAZIABAD-201206

INTRODUCTION

In today's data-driven business environment, organizations gather vast amounts of customer data from various touchpoints. However, raw data alone offers limited value unless it is analyzed and interpreted effectively. One powerful method of gaining insights from such data is **customer segmentation**, which involves grouping customers based on similar characteristics or behaviors. This enables businesses to better understand their customer base, deliver personalized services, and implement targeted marketing strategies.

This project focuses on **unsupervised learning**, specifically using the **K-Means Clustering** algorithm, to segment customers without relying on predefined labels. The segmentation is based on key features such as age, annual income, and spending score. By leveraging Python and its powerful data science libraries—**Pandas** for data handling, **Scikit-learn** for modeling, and **Matplotlib** and **Seaborn** for data visualization—this project aims to reveal hidden patterns in customer behavior.

Customer segmentation plays a critical role in enhancing decision-making processes, from product recommendations and promotional campaigns to customer relationship management. Effective segmentation allows businesses to allocate resources efficiently, identify high-value customer groups, and tailor offerings to match the preferences of different segments.

The methodology adopted in this project includes data loading, preprocessing, scaling, determining the optimal number of clusters using the Elbow Method, applying the K-Means algorithm, and visualizing the segmented customer groups. By the end of the analysis, valuable insights into customer patterns will be extracted, supporting better strategic planning and business performance optimization.

TABLE OF CONTENTS

METHODOLOGY

Customer segmentation is a critical aspect of modern business strategy. It involves categorizing customers into distinct groups based on shared characteristics or behaviors, allowing businesses to tailor marketing, product offerings, and communication strategies more effectively. In this project, we leverage **unsupervised learning**, specifically the **K-Means Clustering** algorithm, to perform data-driven customer segmentation. The methodology follows a structured process to ensure accuracy, relevance, and interpretability. The methodology followed in this project ensures a structured and logical approach to performing customer segmentation using unsupervised learning techniques. The primary aim is to divide customers into meaningful groups based on their characteristics, allowing businesses to understand behavioral patterns and make data-driven decisions.
The steps involved are as follows:

1. Data Collection

The foundation of any machine learning project is the data. The dataset used in this project is a CSV file (Mall_Customers.csv) containing demographic and behavioral information about customers. The key attributes include:

- CustomerID – Unique identifier for each customer
- Gender – Gender of the customer
- Age – Customer's age
- Annual Income (k$) – Yearly income of the customer in thousands
- Spending Score (1–100) – A score assigned based on customer behavior and spending patterns

This structured dataset forms the foundation for unsupervised analysis. These attributes give a rich profile of each customer and set the stage for segmentation based on age, income, and spending patterns.

2. Data Preprocessing

Before any clustering can be performed, the dataset must be cleaned and prepared for analysis. Data preprocessing ensures that the inputs to the model are of high quality, consistent, and suitable for mathematical operations.

- Handling Missing Values: The dataset is checked for null or missing entries. Fortunately, Mall_Customers.csv contains clean data, so no imputation or record removal is necessary.

- Feature Selection: We focus on the most relevant features that influence customer behavior. In this project, Age, Annual Income (k$), and Spending Score (1–100) are selected. These numerical attributes are ideal for clustering, as they directly relate to how customers behave and what segment they might belong to.

- Feature Scaling: Since clustering algorithms like K-Means depend on calculating distances (typically Euclidean), the scale of the features can greatly impact the results. For example, Annual Income values are much larger in magnitude than Spending Score or Age. To ensure each feature contributes equally, StandardScaler from the sklearn library is used to normalize the features, transforming them to have a mean of 0 and standard deviation of 1.

3. Optimal Cluster Detection (Elbow Method)

An essential part of the process is determining the optimal number of clusters, k. Too few clusters may oversimplify the segmentation, while too many can lead to overfitting and confusion.

The Elbow Method helps you determine the optimal number of clusters (k) by:

- Plotting the number of clusters (x-axis) vs. the WCSS (y-axis).
- Looking for the "elbow point"—the value of k where the WCSS starts to flatten out.
    - Before the elbow: adding more clusters significantly reduces WCSS.
    - After the elbow: adding more clusters doesn’t give much benefit (overfitting).

The Elbow Method is a popular heuristic for finding the best value of k. Here's how it works:

- For a range of k values (typically from 1 to 10), the Within-Cluster Sum of Squares (WCSS) is calculated. WCSS measures the variance within each cluster; lower values mean more compact clusters.
- These WCSS values are plotted against the number of clusters.
- The "elbow point" on the graph—where the rate of decrease sharply changes—suggests the optimal number of clusters. This point indicates that adding more clusters beyond this number yields diminishing returns in terms of tighter clustering.

4. Model Building (K-Means Clustering)

With the ideal number of clusters selected, we apply the K-Means Clustering algorithm to segment the customers:

- The algorithm randomly initializes k centroids in the feature space.
- Each customer is assigned to the nearest centroid based on distance.
- The centroids are then updated by calculating the mean of all data points in each cluster.
- These steps are repeated iteratively until the centroids stabilize and customer assignments no longer change significantly.

The result is a new column added to the dataset, indicating the cluster label assigned to each customer. Each cluster now represents a unique group with similar behaviors and attributes.

5. Data Visualization

To interpret the clustering results, visualizations are generated using Seaborn and Matplotlib:

- Scatter Plots are used to show customer distribution based on income and spending score, colored by cluster.
- 3D plots (optional) may be used to visualize clusters using all three selected features: age, income, and spending.

These visualizations help in clearly understanding customer groups and their behavioral characteristics.

6. Insight Extraction

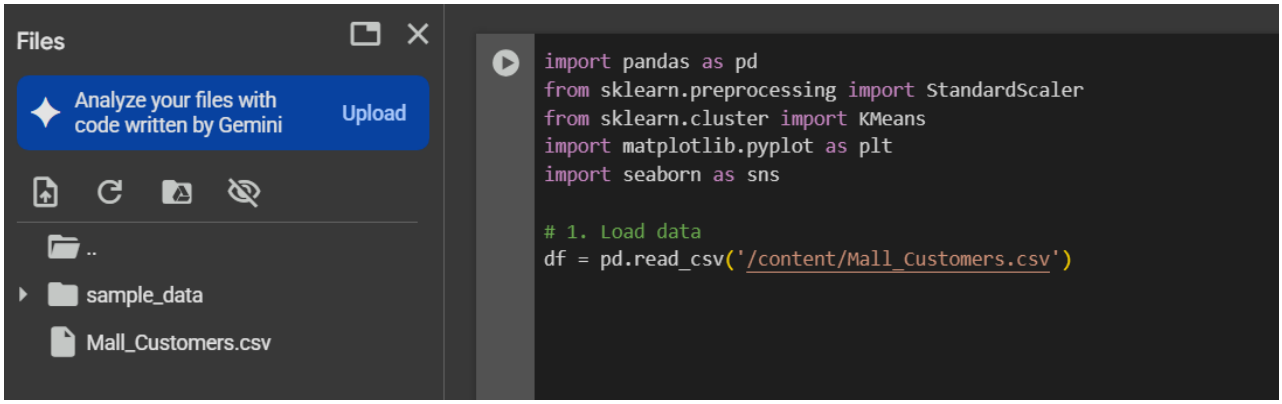
Once clusters are formed, the real value comes from interpreting them and extracting actionable insights:

- High-Income High-Spenders: These customers are prime candidates for premium product offerings, loyalty programs, and personalized services.
- Low-Income Low-Spenders: This group may be more price-sensitive and could respond well to discounts, promotions, or budget product lines.
- Young High-Spenders: These may be trendsetters or tech-savvy shoppers, ideal for early product launches or influencer partnerships.
- Average Income Low-Spenders: This group may represent untapped potential—perhaps they need more engagement, awareness, or incentives to spend more.

By understanding these segments, businesses can personalize marketing strategies, improve customer retention, increase conversion rates, and ultimately boost revenue.

PYTHON CODE

```
import pandas as pd

from sklearn.preprocessing import StandardScaler

from sklearn.cluster import KMeans

import matplotlib.pyplot as plt

import seaborn as sns

# 1. Load data

df = pd.read_csv('/content/Mall_Customers.csv')
```



These are essential for:

- Data handling (pandas)
- Scaling values (StandardScaler)
- Clustering (KMeans)
- Plotting (matplotlib, seaborn)

Display basic information about the dataset

df.info()

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 6 columns):
 #   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          200 non-null    int64
 5   Cluster                 200 non-null    int32
dtypes: int32(1), int64(4), object(1)
memory usage: 8.7+ KB
```

#print first 5 lines of data

df.head()

```
df.head()
```

	CustomerID	Genre	Age	Annual_Income_(k$)	Spending_Score	Cluster
0	1	Male	19	15	39	3
1	2	Male	21	15	81	2
2	3	Female	20	16	6	3
3	4	Female	23	16	77	2
4	5	Female	31	17	40	3

#Feature Selection

features = df[['Age', 'Annual_Income_(k$)', 'Spending_Score']]

- Age
- Annual Income
- Spending Score

These features are what the algorithm will use to group customers.

 # Scale Data

scaler = StandardScaler()

scaled_features = scaler.fit_transform(features)

Why scale?

- K-Means is **distance-based**, so features like "Income" and "Age" should be on the same scale.

```
features = df[['Age', 'Annual_Income_(k$)', 'Spending_Score']]
# 3. Scale data
scaler = StandardScaler()
scaled_features = scaler.fit_transform(features)
```

Optimal Clusters (Elbow Method)

```
wcss = []
for i in range(1, 11):
    kmeans = KMeans(n_clusters=i, random_state=0)
    kmeans.fit(scaled_features)
```

wcss.append(kmeans.inertia_)

#WCSS (Within-Cluster Sum of Squares): Measures compactness of clusters.

#You loop from 1 to 10 clusters and store the WCSS for each.

plot:

plt.plot(range(1, 11), wcss, marker='o')

plt.title('Elbow Method')

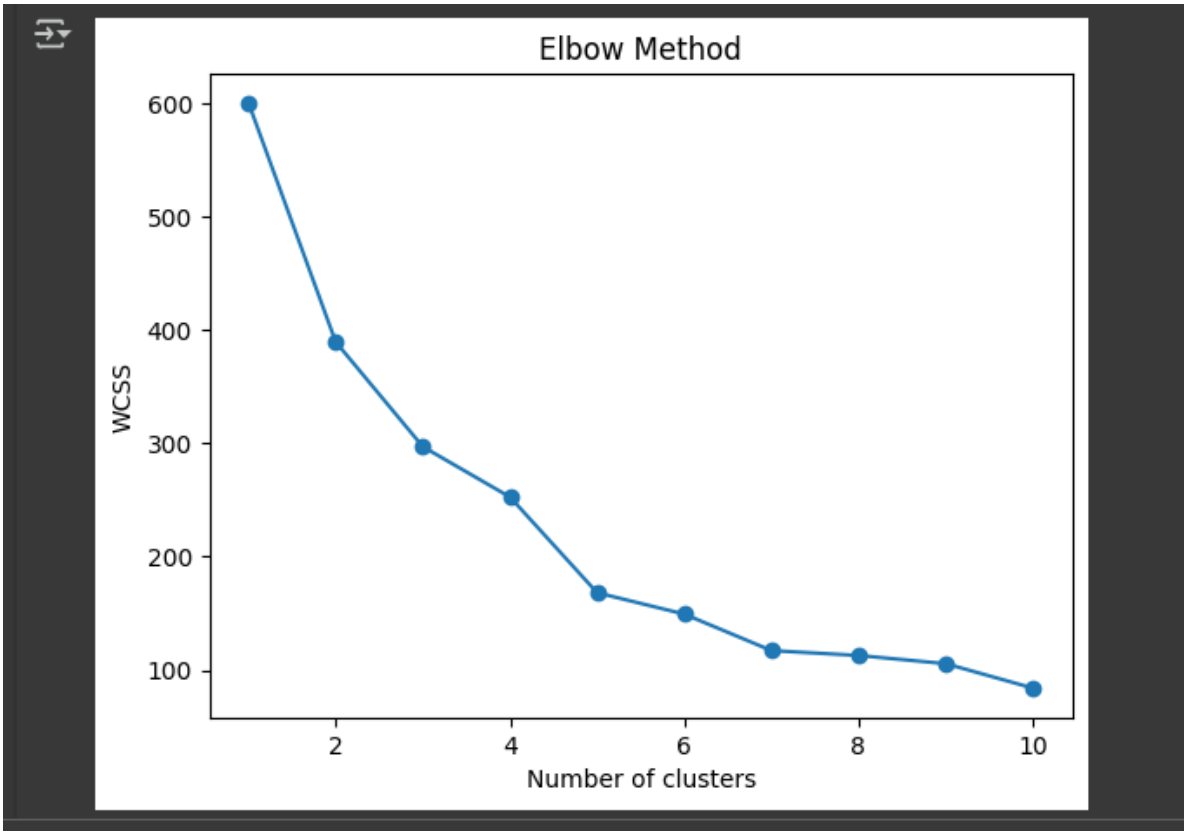
plt.xlabel('Number of Clusters')

plt.ylabel('WCSS')

plt.show()

```
[ ]  wcss = []
     for i in range(1, 11):
         kmeans = KMeans(n_clusters=i, random_state=0)
         kmeans.fit(scaled_features)
         wcss.append(kmeans.inertia_)

     plt.plot(range(1, 11), wcss, marker='o')
     plt.title('Elbow Method')
     plt.xlabel('Number of clusters')
     plt.ylabel('WCSS')
     plt.show()
```

Look for the "elbow" point (sharp drop then flattening). That's your optimal number of clusters.



5. Apply KMeans

#**K-Means Clustering** is a popular **unsupervised machine learning algorithm** used to group data into **clusters** based on similarity. It helps in identifying **natural patterns** or **groupings** in data without having predefined labels or categories.

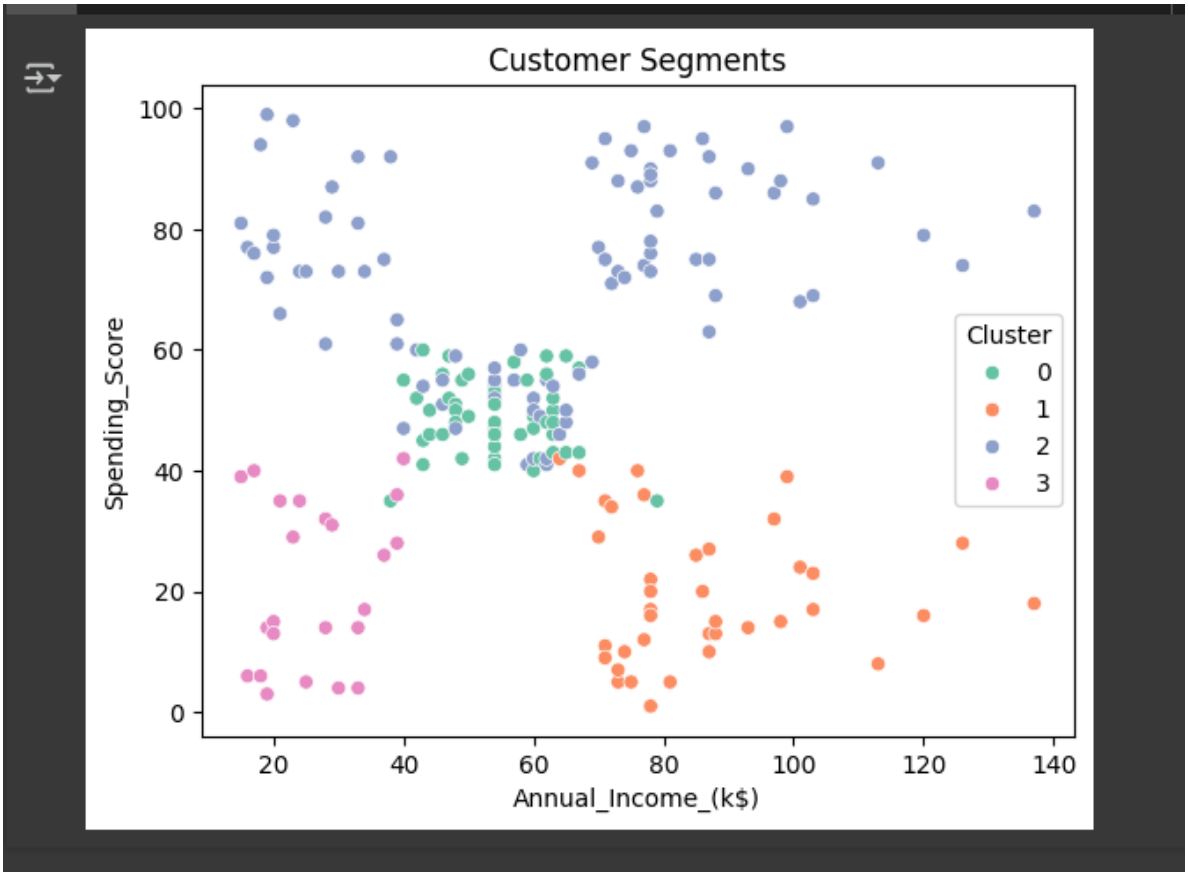
```
kmeans = KMeans(n_clusters=4, random_state=0)
clusters = kmeans.fit_predict(scaled_features)
```

6. Add cluster labels to data

```
df['Cluster'] = clusters
```

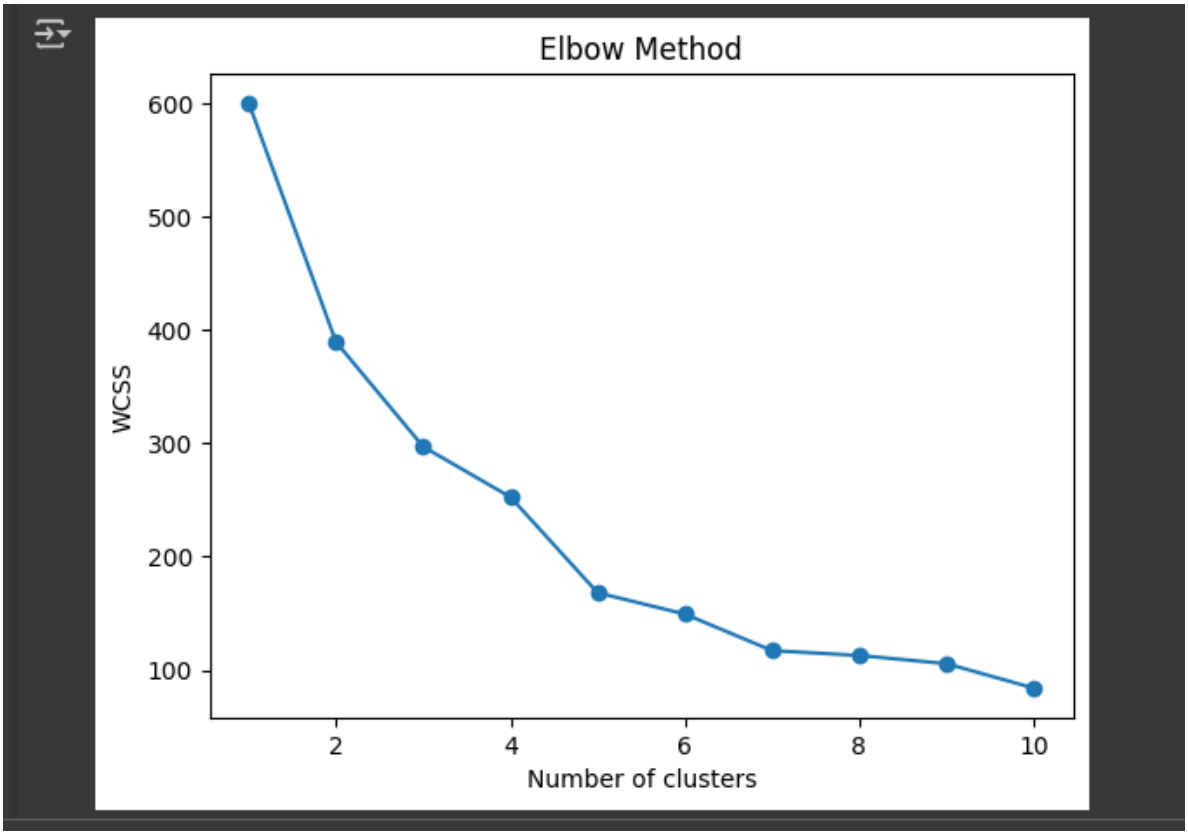
```python
# 5. Apply KMeans
kmeans = KMeans(n_clusters=4, random_state=0)
clusters = kmeans.fit_predict(scaled_features)

# 6. Add cluster labels to data
df['Cluster'] = clusters

# 7. Visualize
sns.scatterplot(data=df, x='Annual_Income_(k$)', y='Spending_Score', hue='Cluster', palette='Set2')
plt.title('Customer Segments')
plt.show()
```

7. Visualize
sns.scatterplot(data=df,x='Annual_Income_(k$)',y='Spending_Score',
hue='Cluster', palette='Set2')
plt.title('Customer Segments')
plt.show()



INSIGHTS AND REPORTS

Insight from the Graph: Elbow Method



The Elbow Method graph visually aids in identifying the optimal number of clusters (K) for K-Means clustering. Here's what it reveals:

- **X-axis:** Represents the number of clusters (K).
- **Y-axis:** Shows WCSS (Within-Cluster Sum of Squares), which measures how compact the clusters are.

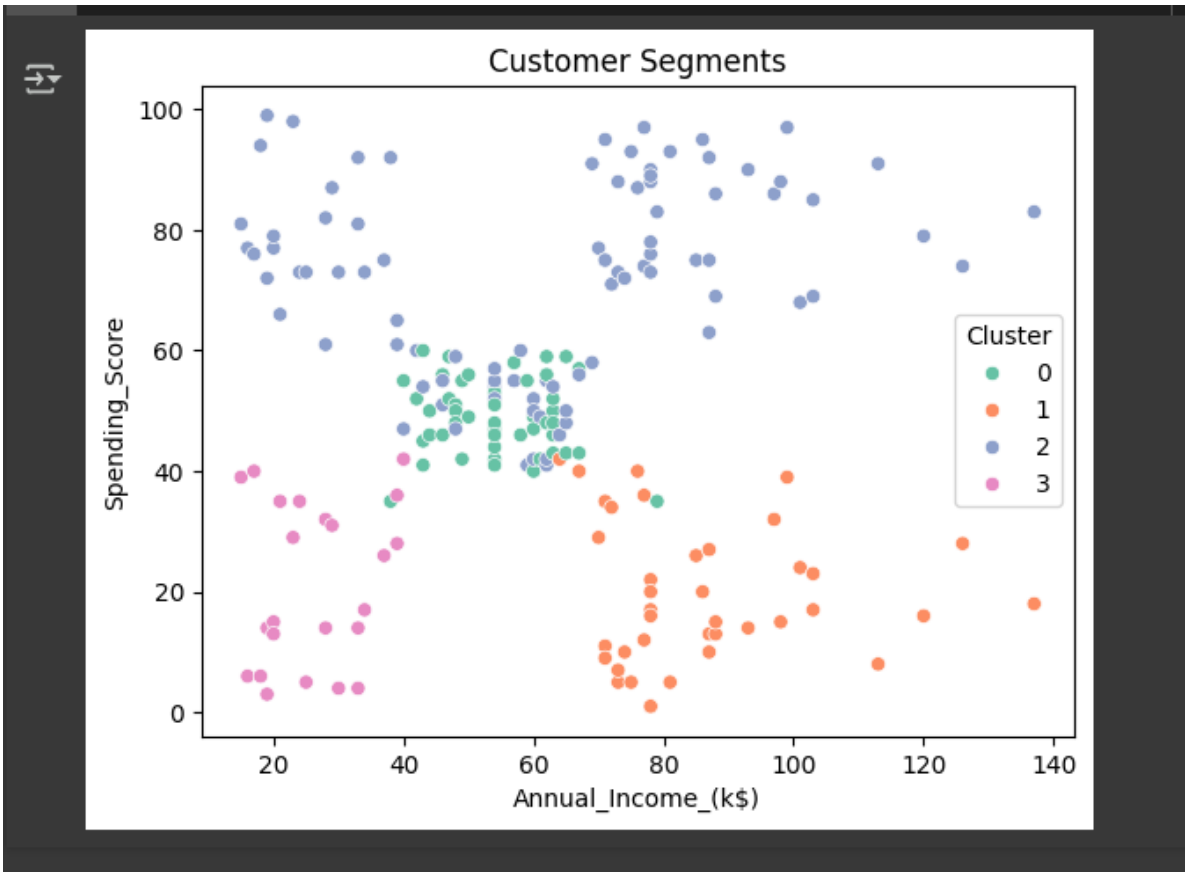
As K increases, WCSS decreases because the data points are split into more, tighter groups. However, this decrease isn't always meaningful beyond a certain point.

- **Key Insight:**
  - There is a significant drop in WCSS from K = 1 to K = 4.
  - After K = 4, the curve begins to flatten, forming an “elbow” shape.

This "elbow point" is where adding more clusters offers diminishing returns in WCSS reduction. Hence:

- **Conclusion from the Graph:**
  - **Optimal number of clusters = 4**
  - This number offers a balance between good clustering and model simplicity.
- **Business Interpretation:**
  - Segmenting customers into 4 groups enables:
    - Targeted marketing for different customer types (e.g., big spenders, budget shoppers).
    - Efficient resource allocation for campaigns or services.
    - Smarter decisions for personalized recommendations or promotions.

Insight from the Graph: Customer Segments Visualization



The scatter plot illustrates how customers are grouped based on their annual income (X-axis) and spending score (Y-axis). Each point represents a customer, and different colors indicate their assigned cluster from the K-Means algorithm.

- **Key Graph Features:**
  - Clearly visible 4 distinct clusters, aligning with the Elbow Method's suggested number of segments.
  - Customers are grouped by similarities in their income and spending behavior.
- **Key Insights:**
  1. Cluster Patterns:
     - Cluster 0: High income, low spending – cautious or value-driven buyers.
     - Cluster 1: Low income, low spending – budget-conscious shoppers.
     - Cluster 2: High income, high spending – premium customers with strong purchasing power.

        - Cluster 3: Moderate income and moderate spending – average or steady buyers.
    2. Business Implications:
        - Focus premium offerings and loyalty programs on Cluster 2.
        - Offer discounts and budget deals to Cluster 1.
        - Target Cluster 0 with personalized, value-based marketing.
        - Engage Cluster 3 with general-purpose promotions.
- **Conclusion:** This segmentation reveals clear behavioral patterns, helping businesses to:
    - Better understand their customer base,
    - Personalize marketing efforts, and
    - Strategically tailor offerings for higher engagement and ROI.

CONCLUSION

In conclusion, customer segmentation using unsupervised learning, particularly through K-Means clustering, proves to be a powerful technique for uncovering hidden patterns within customer data. The structured methodology—from data collection and preprocessing to model building and visualization—enables businesses to classify customers into meaningful groups based on key characteristics such as age, annual income, and spending score. By scaling features and using the Elbow Method, the optimal number of clusters is determined, ensuring that segmentation is both efficient and meaningful.

Once clusters are formed, each customer is assigned to a group that reflects their behavioral and demographic tendencies. These segments can then be interpreted to identify high-value customers, budget-conscious groups, or average spenders. Visualization techniques, like scatter plots, make these groupings easy to understand and communicate. More importantly, the segmented data offers actionable insights—marketers can craft personalized campaigns, product teams can tailor offerings, and customer service teams can proactively engage different segments based on their needs.

The application of K-Means clustering in this project exemplifies the essence of unsupervised learning: discovering structure in unlabeled data. It highlights how machine learning can transition raw numerical information into strategic business intelligence. Through clustering, companies can stop treating customers as a monolithic group and start engaging them as distinct segments with unique preferences and behaviors. This not only enhances customer satisfaction but also improves operational efficiency and revenue potential. Ultimately, customer segmentation powered by unsupervised learning transforms data into decisions, enabling businesses to connect with the right people in the right way.