www.kiet.edu
Delhi-NCR, Ghaziabad

KIET
GROUP OF INSTITUTIONS
Connecting Life with Learning

Assessment Report

on

“Customer Segmentation in E-commerce:”

submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY DEGREE

SESSION 2024-25

in

CSE-AIML

By

Prashant Tiwari (202401100400142)

Under the supervision of

“Abhishek Shukla Sir”

KIET Group of Institutions, Ghaziabad

Affiliated to
Dr. A.P.J. Abdul Kalam Technical University, Lucknow
(Formerly UPTU)
“April 18, 2025”

Introduction

In this problem, we are given a customer transactional dataset. The objective is to preprocess the data, engineer meaningful features, and apply the K-Means clustering algorithm to segment customers based on their shopping behavior.

Clustering helps businesses to:

- Identify customer groups.
- Target marketing campaigns.
- Improve personalized recommendations.

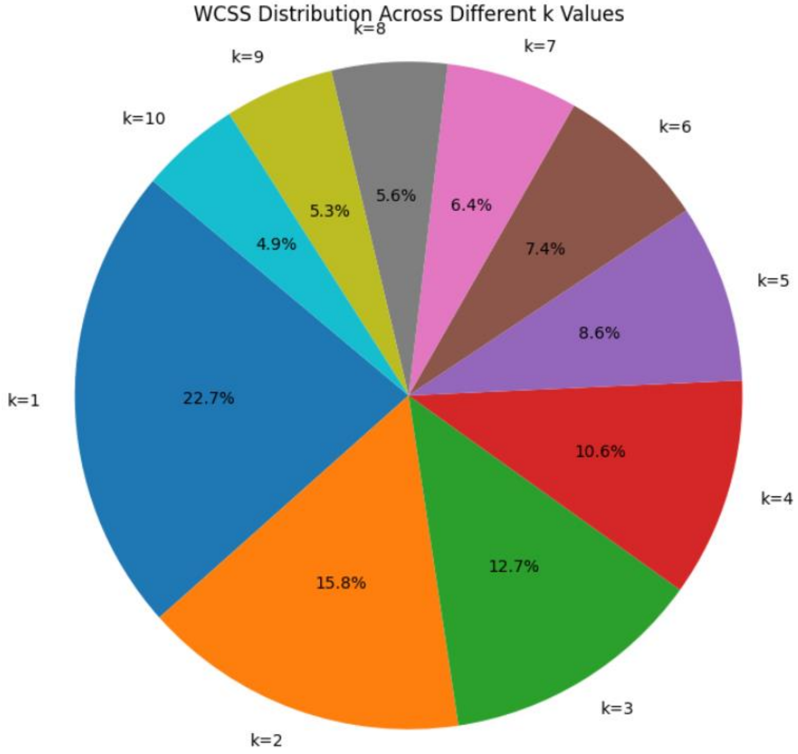
Methodology

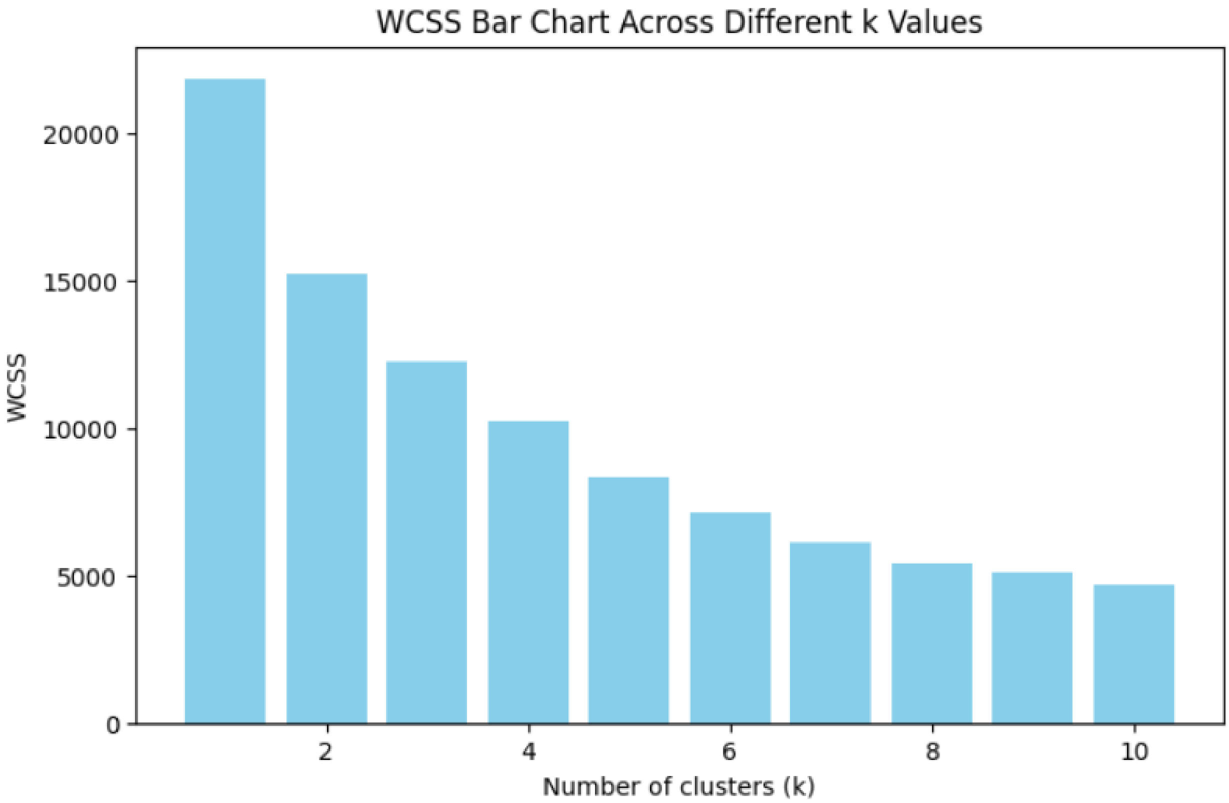
1. **Data Preprocessing:**
   - **Dropped any records with missing Customer ID.**
   - **Created a new feature Total Price (calculated as Quantity × Unit Price).**
2. **Feature Engineering:**
   - **Aggregated data on a customer level to calculate total items purchased, total spend, and average order size.**
3. **Normalization:**
   - **Standardized the features using Standard Scaler to ensure fair distance calculation during clustering.**
4. **Clustering:**
   - **Used the Elbow Method to determine the optimal number of clusters.**
   - **Applied K-Means clustering for customer segmentation.**
5. **Visualization:**
   - **Created three types of graphs to support interpretation:**
     - **Line Plot (Elbow curve).**
     - **Bar Chart.**
     - **Pie Chart.**

CODE

```
# Import libraries
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler

# Load dataset
df = pd.read_csv('your_dataset.csv')
df.dropna(subset=["CustomerID"], inplace=True)

# Feature creation
df["TotalPrice"] = df["Quantity"] * df["UnitPrice"]

# Grouping customer data
customer_df = df.groupby("CustomerID").agg({
    "InvoiceNo": "nunique",
    "Quantity": "sum",
    "TotalPrice": "sum"
}).rename(columns={
    "InvoiceNo": "NumPurchases",
    "Quantity": "TotalItems",
    "TotalPrice": "TotalSpent"
})

# Normalization
scaler = StandardScaler()
X_scaled = scaler.fit_transform(customer_df)

# KMeans clustering and WCSS calculation
wcss = []
for k in range(1, 11):
    kmeans = KMeans(n_clusters=k, random_state=42)
    kmeans.fit(X_scaled)
    wcss.append(kmeans.inertia_)

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

Output/Result





References/Credits

Dataset: [Kaggle]

Libraries:

- pandas
- matplotlib
- sklearn