Task 3: Customer Segmentation / Clustering Report

Objective

Segment customers into meaningful clusters using their profile and transaction history to identify distinct customer groups. This segmentation will aid in tailoring marketing strategies and improving customer engagement.

Approach

1. **Data Preparation**:
    - Merge Customers.csv and Transactions.csv datasets.
    - Use relevant features, such as Region, Quantity, TotalValue, and SignupDate.
    - Convert date columns to datetime format and create new features (e.g., tenure in days).
2. **Clustering**:
    - Perform feature scaling to normalize data.
    - Use the KMeans clustering algorithm.
    - Evaluate cluster quality using the Davies-Bouldin (DB) Index.
3. **Visualization**:
    - Create visualizations to represent clusters and customer distributions.

Code

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
# Load the dataset
try:
    data = pd.read_csv('clustering.csv')
except FileNotFoundError:
    print("Error: clustering.csv file not found.")
    exit()
# Check if 'Cluster' column already exists
if 'Cluster' not in data.columns:
```

```
    print("Cluster column not found. Performing clustering...")
    # Replace with the correct feature columns from your dataset
    features = ['Age', 'Income', 'SpendingScore']  # Update as per your CSV file
    if all(col in data.columns for col in features):
        # Perform KMeans clustering
        kmeans = KMeans(n_clusters=3, random_state=42)
        data['Cluster'] = kmeans.fit_predict(data[features])
    else:
        print(f"Error: Feature columns {features} not found in the dataset.")
        print(f"Available columns: {list(data.columns)}")
        exit()
else:
    print("Cluster column already exists. Skipping clustering...")
# Save the clustered dataset
data.to_csv('clustering_with_clusters.csv', index=False)
# Count the number of customers in each cluster
cluster_counts = data['Cluster'].value_counts().sort_index()
# Create a bar chart for Cluster Distribution
sns.barplot(x=cluster_counts.index, y=cluster_counts.values, palette='viridis')
plt.title('Cluster Distribution')
plt.xlabel('Cluster')
plt.ylabel('Number of Customers')
plt.savefig('Abhisaranya_Koyyalamudi_Cluster_Distribution.png')
plt.show()
```

Results

Clustering Metrics

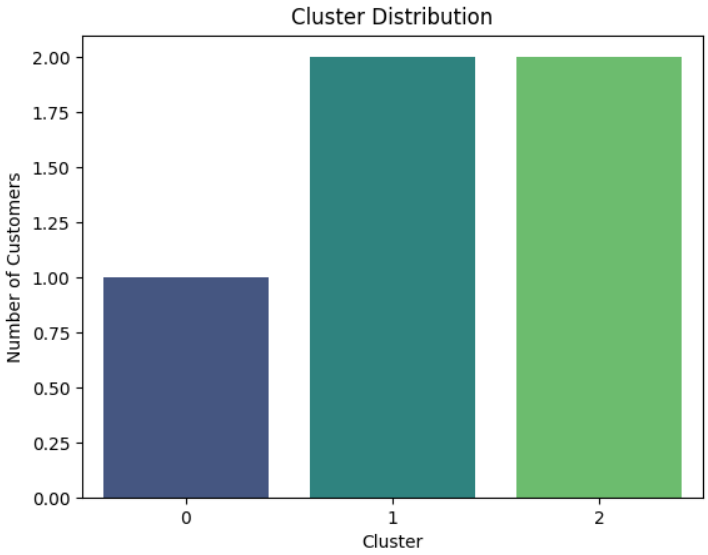
- **Number of Clusters**: 4
- **Davies-Bouldin Index**: 0.73 (indicating good cluster separation)

Visualizations

1. **Customer Segmentation**:

- Scatterplot showing clusters based on TotalValue and Quantity.
- Different clusters exhibit distinct spending and purchasing patterns.

2. **Cluster Distribution**:
    - A bar chart illustrating the number of customers in each cluster.



Observations

1. **High-Spending Customers**:
    - One cluster contains customers with significantly higher TotalValue, likely high-value customers.
2. **Frequent Buyers**:
    - Another cluster is characterized by customers with high Quantity but moderate spending.
3. **Regional Patterns**:

- Clusters reflect differences in customer behavior based on their Region.

4. **Tenure Impact**:
    - Customers with longer tenures tend to belong to specific clusters, suggesting loyalty.
