Assessment Report

on

"Market Basket Analysis"

submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY DEGREE

SESSION 2024-25

in

CSE(AIML)

By

Name : Upwan Singh

Roll Number :

202401100400200

Section: C

Under the supervision of

"ABHISHEK SHUKLA"

KIET Group of Institutions, Ghaziabad

Introduction

- **Market Basket Analysis (MBA)** is a powerful data mining technique used to discover patterns and associations between items frequently purchased together in transactional data.
- **Primary Objective:** To understand customer purchasing behavior and leverage insights for:
    - Product placement optimization
    - Inventory management
    - Cross-selling and up-selling strategies
    - Personalized marketing and recommendations
- **Traditional Approach:**
    - Utilizes **Association Rule Mining algorithms** like **Apriori** and **FP-Growth**.
    - Generates rules such as:
      "If a customer buys bread and butter, they are likely to buy milk."
- **Limitation of Rule-Based MBA:**
    - Focuses only on item-level associations.
    - Does not consider broader customer behavior or segmentation.
- **Advanced Approach – Clustering Techniques:**
    - **K-Means Clustering** is used to identify groups of customers with similar purchasing habits.
    - Helps in profiling customer segments for better-targeted marketing and strategic planning.
- **K-Means Overview:**
    - An unsupervised learning algorithm.
    - Partitions data into **K clusters** based on similarity.
    - Each customer is represented as a vector in a multi-dimensional space (e.g., product categories or spending amounts).
- **Benefits of Using Clustering with MBA:**
    - Moves from individual item relationships to **customer behavior analysis**.
    - Enables discovery of patterns like:
        - High-value customer groups
        - Seasonal shoppers
        - Category-specific buyers
- **Purpose of This Report:**
    - To demonstrate how integrating **K-Means clustering** with MBA can extract deeper insights from transactional data.
    - To provide actionable recommendations for enhancing retail strategies using customer segmentation.

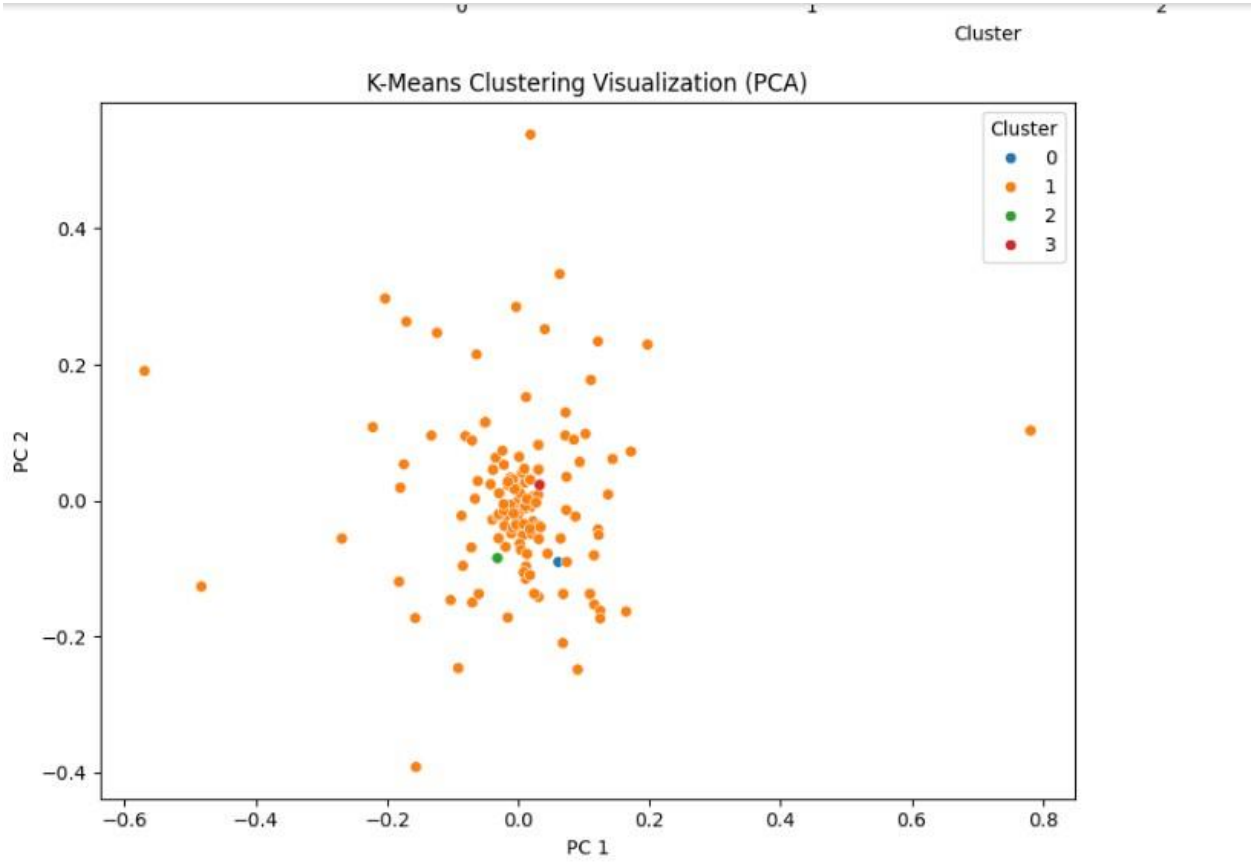
Methodology

- **1. Data Preprocessing:**
  - The raw transactional dataset was cleaned and structured for analysis.
  - Each transaction was converted into a list of items purchased together.
- **2. One-Hot Encoding:**
  - A **one-hot encoding technique** was applied to transform the dataset into a **binary matrix**.
  - In this matrix:
    - Rows represent individual transactions.
    - Columns represent unique items.
    - Each cell contains `1` if the item is present in the transaction, otherwise `0`.
- **3. Clustering with K-Means:**
  - The binary matrix served as input for the **K-Means clustering algorithm**.
  - Transactions were grouped into clusters based on **similarity in item presence**.
  - The optimal number of clusters (**K**) was selected using the **elbow method** or **silhouette score** (if applicable).
- **4. Cluster Analysis:**
  - Each cluster was analyzed to interpret dominant item combinations and purchase patterns.
  - Insights were drawn regarding distinct transaction types and customer behaviors.
- **5. Visualization:**
  - A **heatmap** was generated to visualize item frequency and cluster characteristics.
  - **Principal Component Analysis (PCA)** was applied to reduce data dimensionality.
  - The PCA output was used to plot clusters in a **2D space** for better interpretability and visual analysis.

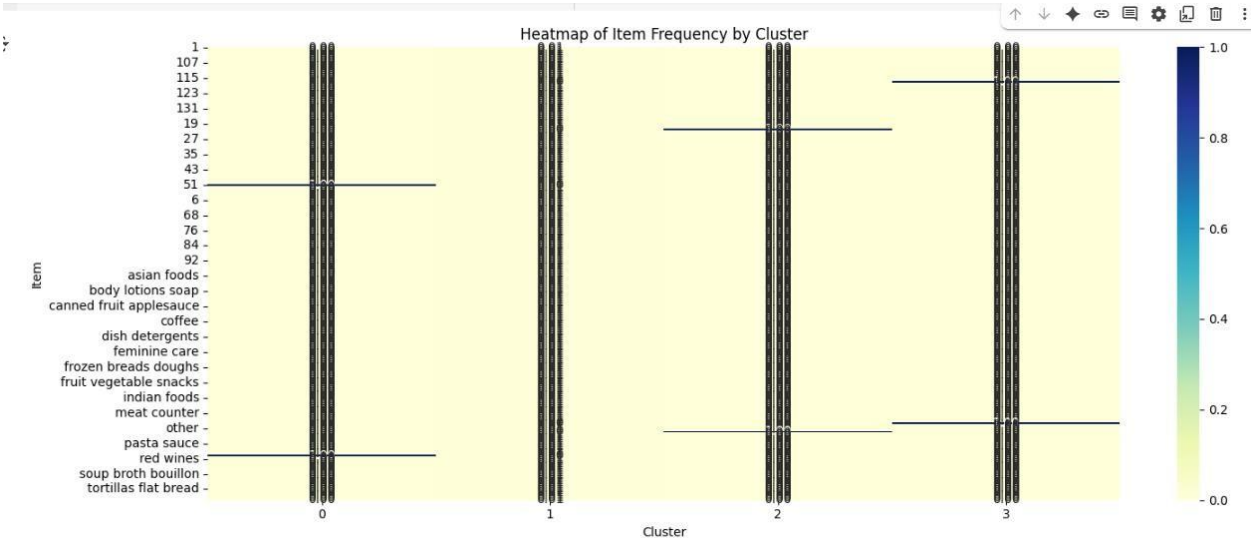
Code

```
import pandas as pd
from mlxtend.preprocessing import TransactionEncoder
from sklearn.cluster import KMeans
from sklearn.decomposition import PCA
import seaborn as sns
import matplotlib.pyplot as plt

# Load the data
df = pd.read_csv("10. Market Basket Analysis.csv")

# Preprocess into transactions
transactions = []
for _, row in df.iterrows():
    items = [str(i).strip() for i in row if pd.notna(i)]
    transactions.append(items)

# One-hot encode
te = TransactionEncoder()
te_array = te.fit(transactions).transform(transactions)
df_encoded = pd.DataFrame(te_array, columns=te.columns_)

# Apply K-Means clustering (you can change n_clusters)
kmeans = KMeans(n_clusters=4, random_state=42)
df_encoded['Cluster'] = kmeans.fit_predict(df_encoded)

# Heatmap of average item presence per cluster
cluster_profile = df_encoded.groupby('Cluster').mean()

plt.figure(figsize=(15, 6))
sns.heatmap(cluster_profile.T, cmap="YlGnBu", annot=True, fmt=".2f")
```

```
plt.title("Heatmap of Item Frequency by Cluster")
plt.xlabel("Cluster")
plt.ylabel("Item")
plt.tight_layout()
plt.show()

# Optional: Visualize clusters using PCA (2D)
pca = PCA(n_components=2)
components = pca.fit_transform(df_encoded.drop(columns='Cluster'))

plt.figure(figsize=(8, 6))
sns.scatterplot(x=components[:, 0], y=components[:, 1], hue=df_encoded['Cluster'],
palette='tab10')
plt.title("K-Means Clustering Visualization (PCA)")
plt.xlabel("PC 1")
plt.ylabel("PC 2")
plt.tight_layout()
plt.show()
```



OUTPUT / RESULT



References / Credits

- mlxtend: http://rasbt.github.io/mlxtend/

- Scikit-learn: https://scikit-learn.org/

- Seaborn: https://seaborn.pydata.org/

- Matplotlib: https://matplotlib.org/

- Dataset Source: '10. Market Basket Analysis.csv'