Assessment Report

on

"MARKET BASKET ANALYSIS"

submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY DEGREE

SESSION 2024-25

in

AI

By

Disha (202401100400083, CSE(AI&ML)-B)

Under the supervision of

"Mr. Abhishek Shukla"

KIET Group of Institutions, Ghaziabad

18 April, 2025

INTRODUCTION

The problem statement is “Market Basket Analysis: Use association rule mining to classify customer purchasing patterns for targeted marketing strategies.” We use the Apriori algorithm to perform this classification.

METHODOLOGY

Data Preparation:

The dataset containing `aisle_id` and `aisle` is preprocessed to simulate customer transactions, grouping similar aisle interactions to form individual baskets.

Transaction Encoding:

Each transaction is transformed into a one-hot encoded format using `TransactionEncoder` to prepare it for pattern mining.

Frequent Itemset Mining:

The **Apriori algorithm** is applied to identify frequent combinations of aisles (itemsets) with a minimum support threshold.

Association Rule Generation:

Using the frequent itemsets, association rules are generated based on **lift** and **confidence** to uncover strong relationships between items.

CODE

```
# Step 1: Import necessary libraries

import pandas as pd

from mlxtend.preprocessing import TransactionEncoder

from mlxtend.frequent_patterns import apriori, association_rules


# Step 2: Load the dataset

file_path = "10. Market Basket Analysis.csv"  # Replace with your path if different

data = pd.read_csv(file_path)


# Step 3: Prepare the data - simulate transactions
```

In real scenario, we'd have user_id or order_id mapping. Let's simulate transactions by treating each 'aisle_id' as a customer's basket

transactions = data.groupby('aisle_id')['aisle'].apply(list).tolist()

Step 4: One-hot encode the transactions

te = TransactionEncoder()

te_data = te.fit_transform(transactions)

df = pd.DataFrame(te_data, columns=te.columns_)

Step 5: Apply the Apriori algorithm

frequent_itemsets = apriori(df, min_support=0.05, use_colnames=True)

Step 6: Generate Association Rules

```
rules = association_rules(frequent_itemsets,
metric="lift", min_threshold=1)


# Step 7: Display the results

print("Frequent Itemsets:\n", frequent_itemsets)

print("\nAssociation Rules:\n",
rules[['antecedents', 'consequents', 'support',
'confidence', 'lift']])
```

OUTPUT

```
frequent_itemsets is empty: True
No frequent itemsets found. Try lowering min_support further or reviewing your data.
```

REFERENCE

- Google
- Kaggle
- Instacart Dataset