**IBM Naan Mudhalavan Project**

**Group-1 Aritificial Intelligence**

**Team Members**  - 4

**Register Number** - 311521106082

**Name**  - Sai Kishore A

**Project name** - market basket analysis

**Phase -** Phase-4

**Algorithm:**

**Apriori Algorithm**:

* The Apriori algorithm is a classic algorithm for association rule mining. It works by iteratively generating item sets with increasing sizes and pruning those that do not meet the minimum support threshold.
* The key steps in the Apriori algorithm are as follows:
  + Generate frequent item sets by finding all item sets that meet the minimum support threshold.
  + Generate association rules from the frequent item sets and calculate their confidence.
  + Filter the rules based on a minimum confidence threshold.

To perform market basket analysis using these algorithms, you typically follow these steps:

1. **Data Preparation**:
   * Collect and preprocess your transaction data, ensuring it's in a suitable format with unique transactions and items.
2. **Algorithm Selection**:
   * Choose the association rule mining algorithm you want to use (e.g., Apriori or FP-Growth).
3. **Parameter Tuning**:
   * Set parameters, such as minimum support and minimum confidence thresholds, based on your analysis objectives and dataset characteristics.
4. **Algorithm Execution**:
   * Run the selected algorithm on your transaction data to find frequent itemsets and generate association rules.
5. **Rule Filtering**:
   * Filter the generated rules based on your confidence and other criteria, if necessary.
6. **Interpretation and Application**:
   * Review the generated rules to gain insights into item associations and patterns in your data.
   * Apply the discovered rules to improve business strategies, such as product recommendations, store layout, or marketing campaigns.
7. **Visualization and Reporting**:
   * Create visualizations and reports to present the results to stakeholders in a clear and understandable way.
8. **Iterate and Refine**:
   * Market basket analysis is an ongoing process. Continuously monitor and refine your strategies based on the insights gained from the analysis.

**Phython Coding:**

**# Import necessary libraries**

import pandas as pd

from mlxtend.frequent\_patterns import apriori

from mlxtend.frequent\_patterns import association\_rules

**# Load your transaction data into a DataFrame**

**# This example assumes a CSV file with a 'Transaction' column containing transaction IDs and an 'Item' column with item names.**

data = pd.read\_csv('transaction\_data.csv')

**# Perform one-hot encoding to convert the data into a binary format**

basket = (data.groupby(['Transaction', 'Item'])['Item']

.count().unstack().reset\_index().fillna(0)

.set\_index('Transaction'))

**# Convert item counts to 0s and 1s**

basket\_sets = basket.applymap(lambda x: 1 if x > 0 else 0)

**# Apply the Apriori algorithm to find frequent itemsets**

frequent\_itemsets = apriori(basket\_sets, min\_support=0.05, use\_colnames=True)

**# Display frequent itemsets**

print("Frequent Itemsets:")

print(frequent\_itemsets)

**# Generate association rules**

rules = association\_rules(frequent\_itemsets, metric='lift', min\_threshold=1.0)

**# Display association rules**

print("\nAssociation Rules:")

print(rules)

**# Filter rules based on different metrics (e.g., confidence)**

filtered\_rules = rules[rules['confidence'] >= 0.7]

**# Display filtered rules**

print("\nFiltered Rules (Confidence >= 0.7):")

print(filtered\_rules)

**# Sort rules by lift**

sorted\_rules = rules.sort\_values(by='lift', ascending=False)

**# Display rules sorted by lift**

print("\nRules Sorted by Lift (Highest Lift First):")

print(sorted\_rules)

**Analysis variation:**

1. **Frequent Itemsets**: It finds and displays the frequent itemsets based on a specified minimum support threshold.
2. **Association Rules**: It generates and displays association rules with the lift as the metric, including antecedent, consequent, and other relevant information.
3. **Filtered Rules**: It filters the association rules based on a confidence threshold and displays the filtered rules.
4. **Rules Sorted by Lift**: It sorts the association rules by lift in descending order and displays the results.