Market Basket Analysis Using the Apriori Algorithm

Mohammad Umar
Yugal Verma
Adam Bielecki
Verna Perumal
Denis Kormaz
Samar Labidi

December 7, 2025

1 Introduction

This report presents the implementation of **Market Basket Analysis (MBA)** using the Apriori algorithm. The objective was to identify associations among products frequently purchased together in real transaction data obtained from Kaggle. The Apriori algorithm was applied to discover frequent itemsets and generate association rules characterized by support, confidence, and lift metrics.

2 Dataset Description

The dataset used in this experiment is the **Groceries Dataset** from Kaggle, containing 9,835 transactions. Each row represents a customer's shopping basket, and each column lists items purchased in that transaction. Products include categories such as dairy, bakery, produce, beverages, and household goods.

Characteristic	**Value**	**Description**
Number of Transactions	9,835	Individual purchase records
Unique Items	169	Distinct products sold
Average Items per Basket	4.4	Mean number of items per transaction
Largest Basket Size	32	Maximum items in a single purchase

Table 1: Dataset Overview

3 Methodology

The analysis followed these main steps:

1. **Data Preprocessing:** Transactions were imported and converted into a binary format using the `TransactionEncoder` from `mlxtend`.

2. **Frequent Itemset Mining:** The Apriori algorithm was applied with a minimum support threshold of 2%.

3. **Rule Generation:** Association rules were extracted using the `association_rules()` function with support ≥ 0.02 and lift > 1.0.

4. **Visualization:** The relationships between support, confidence, and lift were visualized using Matplotlib and Seaborn.

3.1 Mathematical Definitions

The following metrics are used in Association Rule Mining:

$$\text{Support}(A \Rightarrow B) = \frac{\text{count}(A \cup B)}{N}$$

$$\text{Confidence}(A \Rightarrow B) = \frac{\text{count}(A \cup B)}{\text{count}(A)} = P(B|A)$$

$$\text{Lift}(A \Rightarrow B) = \frac{\text{Support}(A \cup B)}{\text{Support}(A) \times \text{Support}(B)}$$

4 Results

4.1 Top Frequent Items

Item	**Support (%)**
Whole Milk	25.5
Other Vegetables	19.1
Rolls/Buns	18.0
Soda	17.0
Yogurt	13.7

Table 2: Most Frequent Individual Items

4.2 Top Association Rules

Rule	**Support**	**Confidence**	**Lift**
Yogurt ⇒ Whole Milk	0.056	0.40	1.57
Rolls/Buns ⇒ Whole Milk	0.048	0.36	1.41
Other Vegetables ⇒ Whole Milk	0.075	0.39	1.55
Whole Milk ⇒ Other Vegetables	0.075	0.30	1.55
Root Vegetables ⇒ Other Vegetables	0.034	0.45	2.36

Table 3: Top Association Rules Derived from Apriori Algorithm

figures/top_rules.png

Figure 1: Scatter plot of Support vs Confidence, with Lift represented by color.

4.3 Observations

- **Whole Milk** appeared in over 25% of all baskets, making it the most frequently purchased product.
- Items such as **Other Vegetables**, **Rolls/Buns**, and **Yogurt** were often purchased alongside Whole Milk.
- The highest lift value (2.36) was found for the rule *Root Vegetables ⇒ Other Vegetables*, suggesting a strong positive association between these two categories.
- Most rules exhibited moderate lift values (1.3–1.6), indicating typical grocery co-purchase patterns.

5 Scalability Discussion

Although this experiment was conducted on a medium-sized dataset, it is known that the Apriori algorithm does not scale efficiently to very large datasets. Its computational cost increases exponentially with the number of unique items because it generates and evaluates all possible candidate item combinations.

If applied to a dataset containing hundreds of thousands of transactions, the algorithm would become significantly slower or even infeasible to execute on standard hardware. This limitation arises from Apriori's exhaustive candidate generation process.

For large-scale data mining, more efficient algorithms such as **FP-Growth** or **Eclat** are commonly used. These algorithms avoid explicit candidate generation by using compressed tree structures, making them suitable for "Big Data" contexts.

6 Conclusion

The Apriori algorithm successfully identified frequent product combinations and interpretable association rules within the Groceries dataset. The analysis confirmed meaningful purchasing relationships, such as the co-occurrence of dairy and vegetable products. While Apriori performs well for educational and small-scale analytical purposes, it is not suitable for large datasets due to its exponential complexity. Future work could involve implementing FP-Growth or parallelized versions of Apriori to enhance scalability.