NAME: Suraj Bhardwaj

NJIT UCID: srb34

Email Address: [srb34@njit.edu](mailto:srb34@njit.edu)

10/7/2024

Professor: Yasser Abduallah

CS 634 101 Data Mining

**Midterm Project Report**

*Implementation and Code Usage*

**Apriori Algorithm Implementation in Retail Data Mining Abstract:**

In this project, I explore the Apriori Algorithm, a fundamental technique in data mining, to uncover associations within retail transactions. By implementing the algorithm and employing various data mining concepts, principles, and methods, I assess its effectiveness and efficiency. Through the design and development of custom data mining tools, I create a custom model for mining valuable insights from transaction data.

**Introduction:**

Data mining is a powerful approach for uncovering hidden patterns and associations within large datasets. Our project focuses on the Apriori Algorithm, a classic method for association rule mining, and its application in a retail context. We'll outline the core data mining concepts and principles applied in our work.

Writing and displaying association rules.The main concept of Apriori Algorithm is to create associations. In order to create associations, I had to figure out what items were most frequent when given the list of transactions. Once the items that are most frequent were found, depending on the user’s support parameter, the support would have to be calculated for each item. After calculating the support value for each item, we can eliminate the items that do not meet the user-defined support parameter. The Apriori algorithm is a classic data mining algorithm that utilizes a brute force approach to find frequent itemsets and generate association rules. It works by iteratively increasing the size of itemsets and filtering out those that do not meet a minimum support threshold.

In this implementation, I applied the Apriori algorithm to a custom dataset associated with a retail store, allowing us to find frequent itemsets and association rules. Key steps in this process included:

* Initializing dictionaries for candidate and frequent itemsets.
* Loading the dataset and itemsets from CSV files.
* Preprocessing the dataset to ensure item order and uniqueness.
* Collecting user input for minimum support and confidence thresholds.
* Iteratively generating candidate itemsets and updating frequent itemsets using the Apriori algorithm, which employs a brute force approach by considering all possible combinations of items.

**Core Concepts and Principles:**

**Frequent Itemset Discovery:**

The Apriori Algorithm revolves around discovering frequent itemsets, i.e., sets of items that frequently co-occur in transactions. These itemsets provide insights into customer purchase behavior and preferences.

**Support and Confidence:**

Two key metrics in data mining are support and confidence. Support measures how frequently an item or itemset occurs, while confidence assesses the likelihood of items being purchased together. These metrics guide our analysis.

**Association Rules:**

By determining strong association rules, I identify which items are commonly purchased together. These rules are instrumental for optimizing sales strategies, such as recommendations.

**Project Workflow:**

Our project follows a structured workflow involving various stages and the application of the Apriori Algorithm:

**Data Loading and Preprocessing:**

We begin by loading transaction data from a retail store dataset. Each transaction consists of a list of items bought by a customer. To ensure data accuracy, we preprocess the dataset, filtering unique items and sorting them based on a predefined order.

**Determination of Minimum Support and Confidence:**

User input is crucial in data mining. We collect the user's preferences for minimum support and confidence levels to filter out less significant patterns.

**Iteration Through Candidate Itemsets:**

The iterative application of the Apriori Algorithm involves generating candidate itemsets of increasing sizes. We start with single items (itemset size K = 1) and proceed to K = 2, K = 3, and so on. This iterative process involves a "brute force" method of generating all possible itemset combinations.

**Support Count Calculation:**

For each candidate itemset, we calculate its support by counting how many transactions contain the itemset. Itemsets that meet the minimum support threshold are retained, while others are discarded.

**Confidence Calculation:**

We evaluate the confidence of association rules, indicating the strength of associations between items. This step requires careful comparison of support values for individual items and itemsets.

**Association Rule Generation:**

Association rules that satisfy both the minimum support and minimum confidence requirements are extracted. These rules reveal valuable insights into which items are often purchased together.

**Results and Evaluation:**

The project's effectiveness and efficiency are evaluated based on performance measures such as support, confidence, and the resulting association rules. We also compare our custom Apriori Algorithm implementation with the Apriori library to assess its reliability.

**Conclusion:**

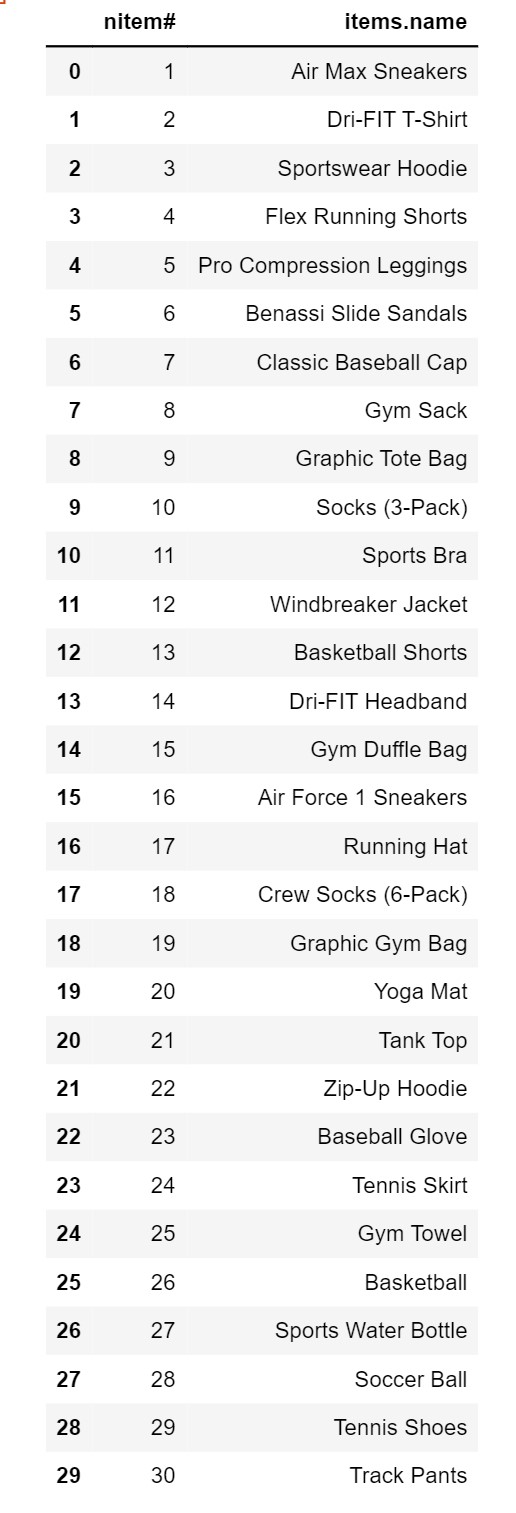
In conclusion, our project demonstrates the application of data mining concepts, principles, and methods. We successfully implemented the Apriori Algorithm to extract meaningful association rules from retail transaction data. The iterative, "brute force" approach, custom algorithm design, and adherence to user-defined parameters exemplify the power of data mining in revealing valuable patterns for decision-making in the retail industry.

*Screenshots*

Here are what the csv files (This program takes in two separate csv files: Item Names & Transactions).

Figure 1 : Nike Item Names CSV file

Figure 2 : Nike Transactions CSV file



Below are screenshots of the code from python file:

Prompts to choose which store you want. We first have to read in the csv files and make sure that the inputs received from the user are valid



Initializing Candidate and Frequent Itemset Dictionaries

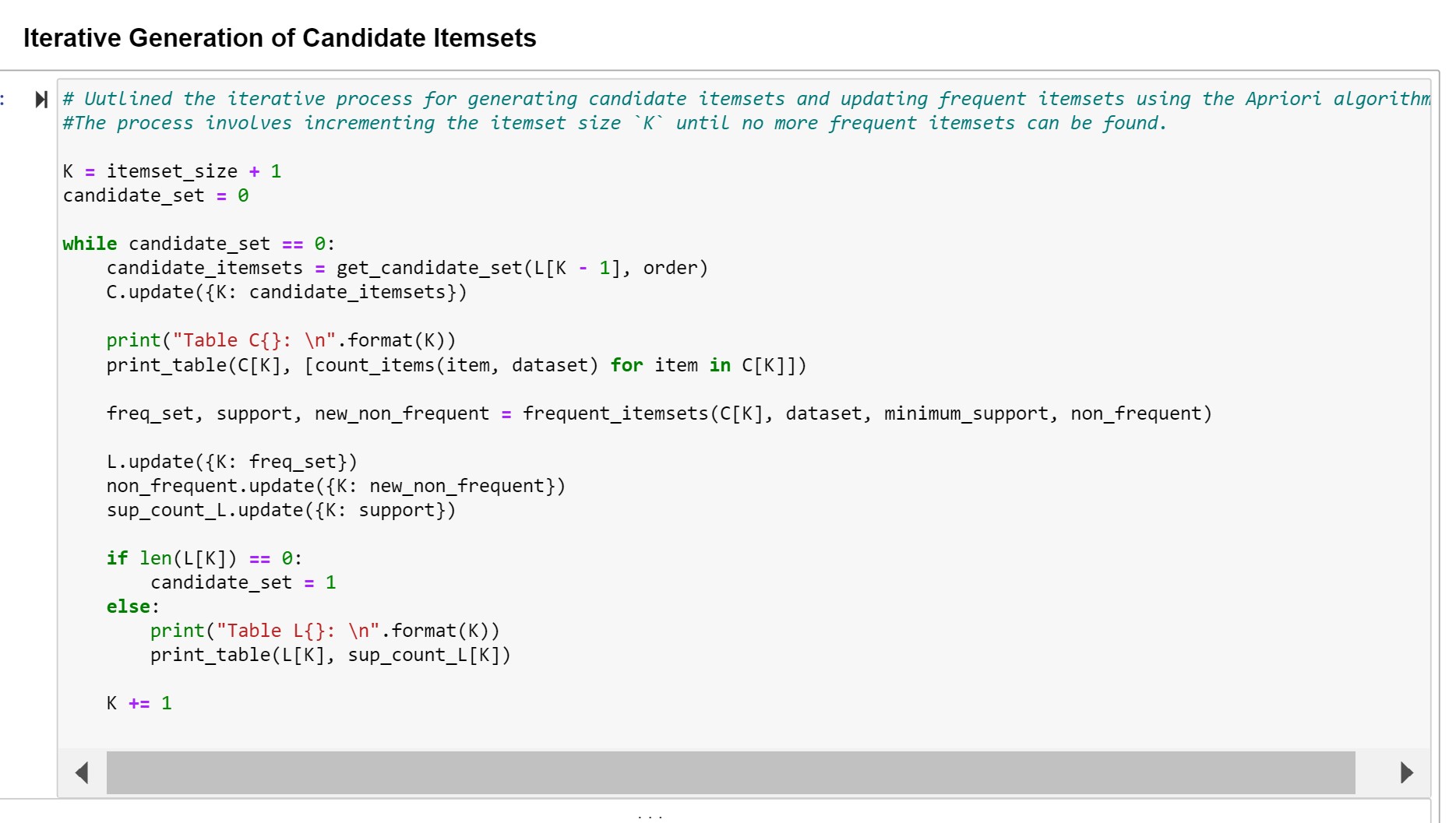


Counting Items and Finding Frequent Itemsets



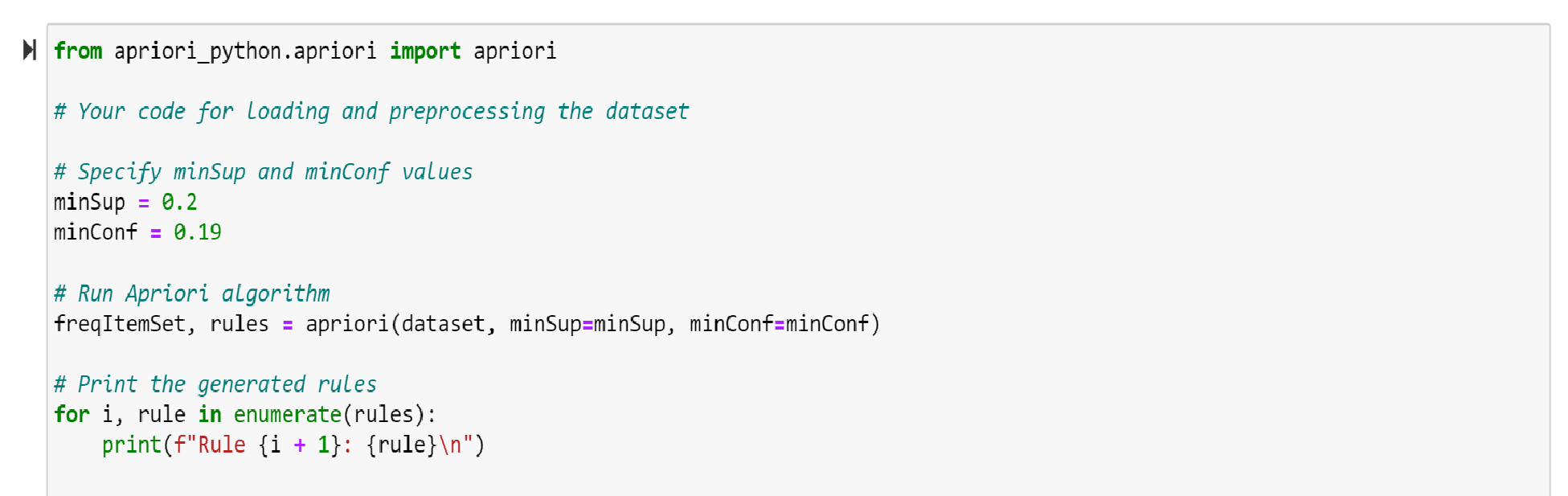




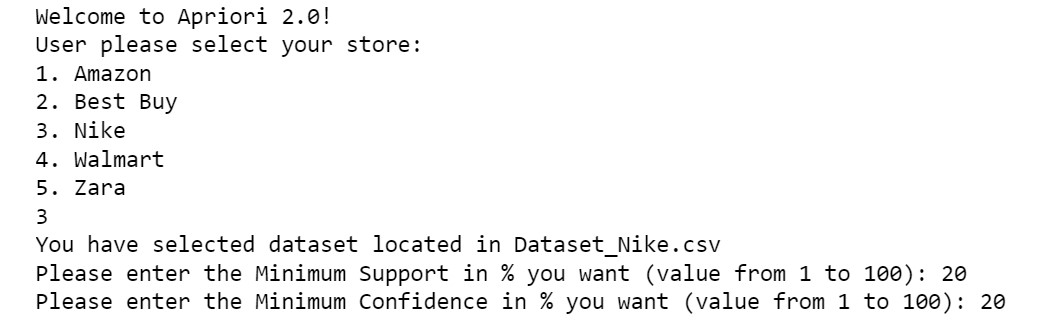


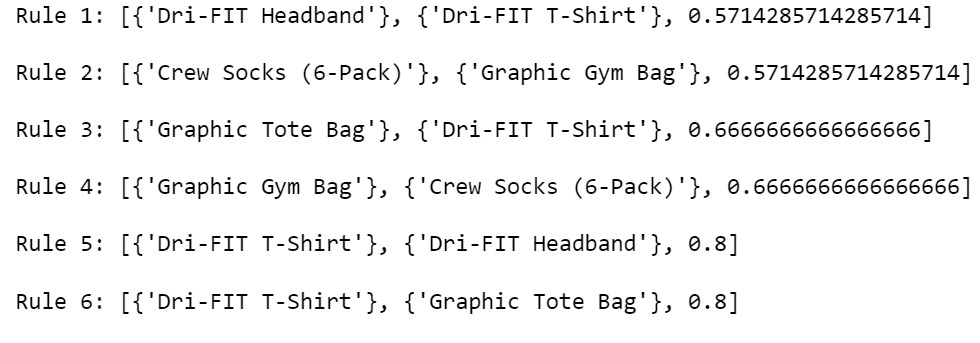


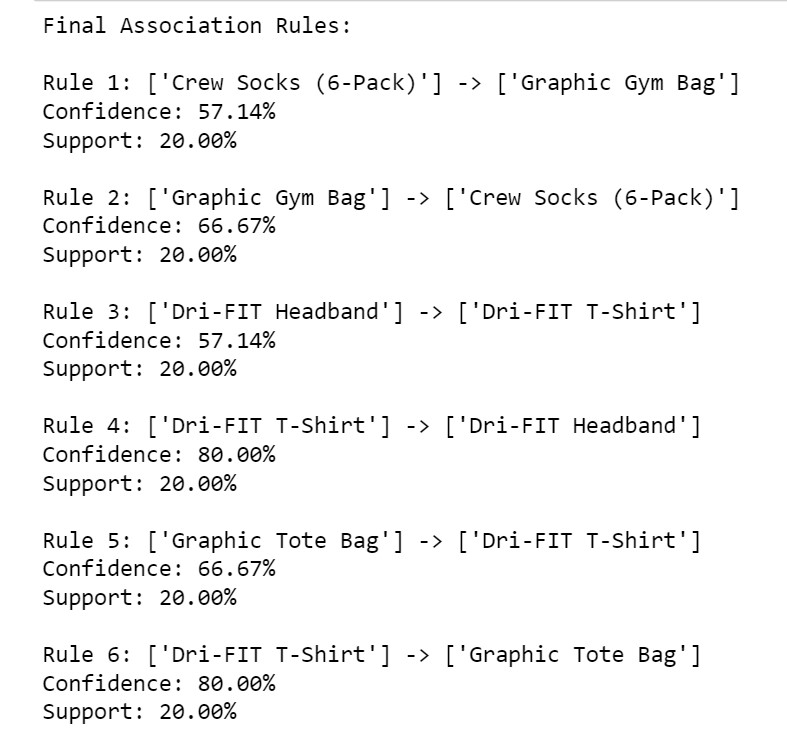
Verified Results with the built-in python package.



Below are screenshots to show that the program runs in the Terminal.



**The final output should be the following:**



Verified With Built in Package:

***Other***

The source code (.py file) and data sets (.csv files) will be attached to the zip file. *Link to Git Repository* https://github.com/mahumabid/Apriori\_Algorithm