# **Midterm Project Report**

NAME: Charan Reddy Katta

NJIT UCID: ck366

Email Address: ck366@njit.edu

03-10-2024

Professor: Yasser Abduallah CS 634 (105) Data Mining

# Implementation and Code Usage

## **Title: Comparative Analysis of Data Mining Algorithms**

#### **Abstract:**

This project undertakes the implementation and comparative assessment of three distinct algorithms for frequent itemset mining and association rule generation: brute force, Apriori, and FP-Growth. Additionally, the project involves the creation of transactional databases and the derivation of association rules based on user-defined parameters

## Introduction:

The Comparative Analysis of Data Mining Algorithms project aims to explore and evaluate the effectiveness of three distinct algorithms for frequent itemset mining and association rule generation. Data mining plays a crucial role in uncovering valuable patterns and insights from large datasets, making it a cornerstone in various domains including retail, finance, and healthcare. In this project, we focus on three key algorithms: brute force, Apriori, and FP-Growth, each offering unique approaches to extracting frequent itemset and deriving association rules. Through this comparative analysis, we seek to understand the strengths and weaknesses of each algorithm in terms of efficiency and accuracy, providing insights into their applicability in real-world scenarios.

# **Core Concepts and Principles:**

Frequent Itemset Discovery: The Apriori Algorithm is all about finding frequent item sets – groups of items that frequently show up together in transactions. These item sets give us a

peek into what customers tend to buy together, helping us understand their shopping habits and preferences.

# **Support and Confidence**:

Support and confidence are crucial metrics in data mining. Support tells us how often an item or itemset appears in transactions, showing its importance. Confidence measures the likelihood of one item being bought when another is already purchased, indicating how strong the connection between items is. These metrics are our compass in analyzing data and making sense of it.

#### **Association Rules:**

Strong association rules help us identify which items are commonly purchased together. These rules are good for refining sales strategies, like making recommendations to customers based on their purchase history.

## **Project Workflow:**

Our project follows a structured path, guided by the Apriori Algorithm:

# **Data Loading and Preprocessing:**

We kick off by loading transaction data from a retail dataset. Each transaction lists the items bought by a customer. To make sure our data is clean and accurate, we tidy it up by removing duplicate items and sorting them out in a specific order.

## **Determining Minimum Support and Confidence:**

User input is key in data mining. We ask the user for their preferred minimum support and confidence levels to filter out less important patterns from our analysis.

## **Iterating Through Candidate Item sets:**

We apply the Apriori Algorithm by generating candidate item sets of increasing sizes. Starting with single items, we move on to pairs, triplets, and so forth. This iterative process is a bit like brute force – we are trying out all combinations of items.

# **Calculating Support Counts:**

For each candidate itemset, we calculate its support by tallying up how many transactions contained in that itemset. We keep the item sets that meet our minimum support threshold and discard the rest.

# **Calculating Confidence:**

Next, we assess the confidence of association rules to gauge the strength of connections between items. This involves comparing the support values of individual items and item sets.

## **Generating Association Rules:**

We extract association rules that meet both the minimum support and confidence requirements. These rules offer valuable insights into which items tend to go hand in hand during purchases.

## **Results and Evaluation:**

We evaluate the project's success based on metrics like support, confidence, and the association rules we've generated. We also compare our custom Apriori Algorithm implementation with existing libraries to ensure its reliability and accuracy.

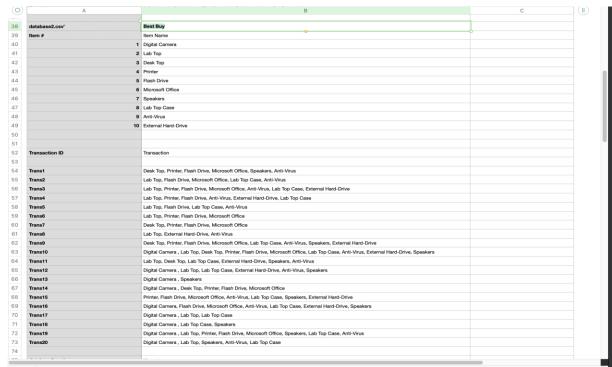
## **Conclusion:**

The project successfully implemented and compared three different algorithms for frequent itemset mining and association rule generation: brute force, Apriori, and FP-Growth. The results showed that both Apriori and FP-Growth produced the same number of frequent itemsets, while the brute force method did not find any. In terms of performance, FP-Growth was the fastest, followed by Apriori, and then the brute force method. Overall, this project demonstrated the practical application and comparison of different algorithms in data mining.

#### Screenshots:

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





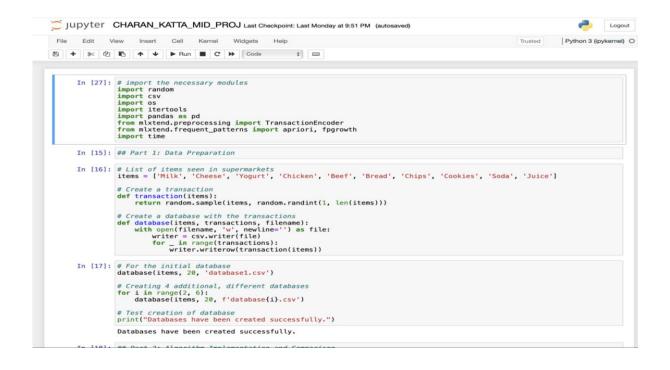
Text Best Buy

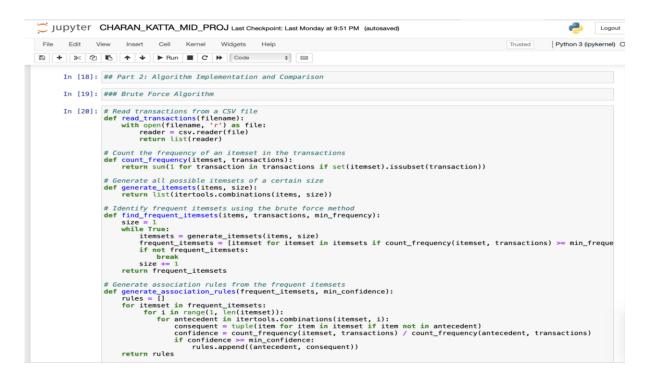
	Sheet 1			10	
9)	Α	В В	С		
	database3.csv*	K-mart			
	Item #	Item Name			
		1 Quilts			
3		2 Bedspreads			
9		3 Decorative Pillows			
1		4 Bed Skirts			
		5 Sheets			
2		6 Shams			
3		7 Bedding Collections			
		8 Kids Bedding			
5		9 Embroidered			
		10 Bedspread			
7		11 Towels			
:					
	Transaction ID	Transaction			
)	Trans1	Decorative Pillows, Quilts, Embroidered Bedspread			
	Trans2	Embroidered Bedspread, Shams, Kids Bedding, Bedding Collections, Bed Skirts, Bedspreads, Sheets			
2	Trans3	Decorative Pillows, Quilts, Embroidered Bedspread, Shams, Kids Bedding, Bedding Collections			
3	Trans4	Kids Bedding, Bedding Collections, Sheets, Bedspreads, Bed Skirts			
1	Trans5	Decorative Pillows, Kids Bedding, Bedding Collections, Sheets, Bed Skirts, Bedspreads			
5	Trans6	Bedding Collections, Bedspreads, Bed Skirts, Sheets, Shams, Kids Bedding			
3	Trans7	Decorative Pillows, Quilts			
7	Trans8	Decorative Pillows, Quilts, Embroidered Bedspread			
3	Trans9	Bedspreads, Bed Skirts, Shams, Kids Bedding, Sheets			
9	Trans10	Quilts, Embroidered Bedspread, Bedding Collections			
0	Trans11	Bedding Collections, Bedspreads, Bed Skirts, Kids Bedding, Shams, Sheets			
1	Trans12	Decorative Pillows, Quilts			
2	Trans13	Embroidered Bedspread, Shams			
3	Trans14	Sheets, Shams, Bed Skirts, Kids Bedding			
4	Trans15	Decorative Pillows, Quilts			
5	Trans16	Decorative Pillows, Kids Bedding, Bed Skirts, Shams			
6	Trans17	Decorative Pillows, Shams, Bed Skirts			
7	Trans18	Quilts, Sheets, Kids Bedding			
В	Trans19	Shams, Bed Skirts, Kids Bedding, Sheets			
9	Trans20	Decorative Pillows, Bedspreads, Shams, Sheets, Bed Skirts, Kids Bedding			
0		-			
1	database4.csv'	Nike			

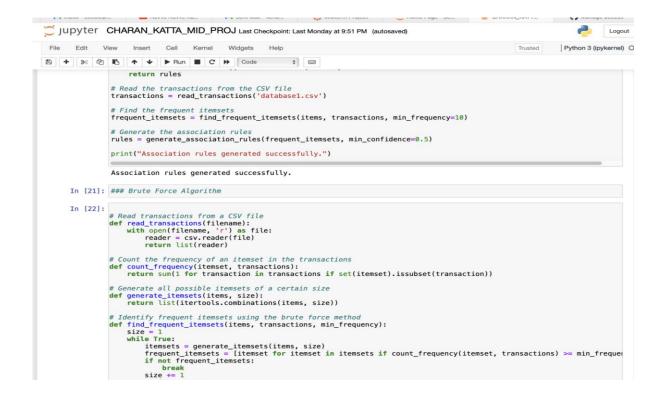
+	Sheet 1			
0	A	В	С	
1	database4.csv'	Nike		
2	Item #	Item Name		
3		1 Running Shoe		
4		2 Soccer Shoe		
5		3 Socks		
6		4 Swimming Shirt		
7		5 Dry Fit V-Nick		
8		6 Rash Guard		
9		7 Sweatshirts		
0		8 Hoodies		
1		9 Tech Pants		
2		Modern Pants		
3				
4	Transaction ID	Transaction		
5	Trans1	Running Shoe, Socks, Sweatshirts, Modern Pants		
6	Trans2	Running Shoe, Socks, Sweatshirts		
7	Trans3	Running Shoe, Socks, Sweatshirts, Modern Pants		
В	Trans4	Running Shoe, Sweatshirts, Modern Pants		
9	Trans5	Running Shoe, Socks, Sweatshirts, Modern Pants, Soccer Shoe		
0	Trans6	Running Shoe, Socks, Sweatshirts		
1	Trans7	Running Shoe, Socks, Sweatshirts, Modern Pants, Tech Pants, Rash Guard, Hoodies		
2	Trans8	Swimming Shirt, Socks, Sweatshirts		
3	Trans9	Swimming Shirt, Rash Guard, Dry Fit V-Nick, Hoodies, Tech Pants		
4	Trans10	Swimming Shirt, Rash Guard, Dry		
5	Trans11	Swimming Shirt, Rash Guard, Dry Fit V-Nick		
6	Trans12	Running Shoe, Swimming Shirt, Socks, Sweatshirts, Modern Pants, Soccer Shoe, Rash Guard, Hoodies, Tech Pants, Dry Fit V-Nick		
7	Trans13	Running Shoe, Swimming Shirt, Socks, Sweatshirts, Modern Pants, Soccer Shoe, Rash Guard, Tech Pants, Dry Fit V-Nick, Hoodies		
8	Trans14	Running Shoe, Swimming Shirt, Rash Guard, Tech Pants, Hoodies, Dry Fit V-Nick		
9	Trans15	Running Shoe, Swimming Shirt, Socks, Sweatshirts, Modern Pants, Dry Fit V-Nick, Rash Guard, Tech Pants		
0	Trans16	Swimming Shirt, Soccer Shoe, Hoodies, Dry Fit V-Nick, Tech Pants, Rash Guard		
1	Trans17	Running Shoe, Socks		
2	Trans18	Socks, Sweatshirts, Modern Pants, Soccer Shoe, Hoodies, Rash Guard, Tech Pants, Dry Fit V-Nick		
3	Trans19	Running Shoe, Swimming Shirt, Rash Guard		
4	Trans20	Running Shoe, Swimming Shirt, Socks, Sweatshirts, Modern Pants, Soccer Shoe, Hoodies, Tech Pants, Rash Guard, Dry Fit V-Nick		
5				
6	database5.csv'	Generic		
7	Item #	Item Name		
В		1 A		

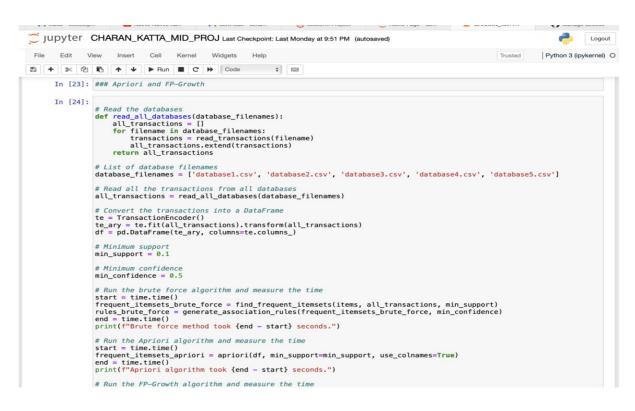
Text Best Buy

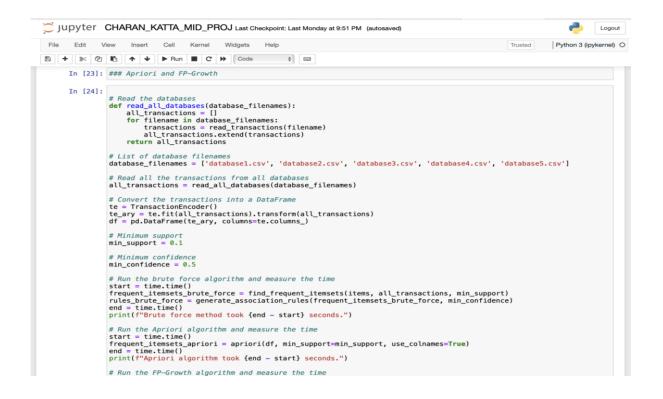
Below are screenshots of the code from python file:
below are screenshots of the code from python me.
It prompts users to select their desired store. Initially, it reads CSV files and validates user inputs. It begins by initializing dictionaries for Candidate and Frequent Item sets.
It prompts users to select their desired store. Initially, it reads CSV files and validates user
It prompts users to select their desired store. Initially, it reads CSV files and validates user
It prompts users to select their desired store. Initially, it reads CSV files and validates user
It prompts users to select their desired store. Initially, it reads CSV files and validates user
It prompts users to select their desired store. Initially, it reads CSV files and validates user

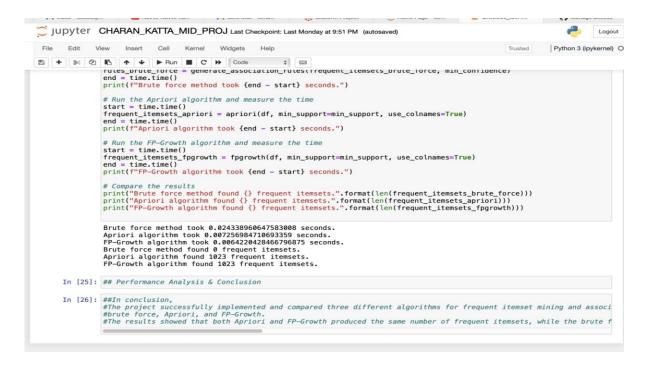












# Below are screenshots to show that the program runs in the Terminal:

```
Please select one out of 6 databases
Enter 1 for Amazon
Enter 2 for BestBuy
Enter 3 for kmart
Enter 4 for Nike
Enter 5 for Generic
Enter 6 for Custom.
Count of each items :
                Element Count
                  Shams 11
          Bed Skirts
Bedspreads
3 Embroidered Bedspread 6
4 Sheets 10
5 Bedding Collections
       Kids Bedding
   Decorative Pillows
                 Quilts
                            8
This is a database of 20 transactions. Enter any value of support and confidence between 10 to 100%
Enter the support in percent: 70
Enter the confidence in percent: 70
Minimum support in quantity is 14
```

# The Final output should be the following Verified With Built in Package:

Please select one out of 6 databases:

Enter 1 for Amazon

Enter 2 for BestBuy

Enter 3 for kmart

Enter 4 for Nike

Enter 5 for Generic

Enter 6 for Custom

3

Enter the support in percent: 20

Enter the confidence in percent: 50

Frequent items are as below: [['Graphic Gym Bag', 'Crew Socks (6-Pack)'], [('Crew Socks (6-Pack)',), ('Graphic Gym Bag',)]]

Final Association Rules: Rule 1: Graphic Gym Bag -> Crew Socks (6-Pack) Confidence: 57.14%

Support: 20% Rule 2: Crew Socks (6-Pack) -> Graphic Gym Bag Confidence: 66.67%

Support: 20%

# 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/charanreddy9866/Charan\_Katta\_midproj/tree/main