



CS 634

Apriori Algorithm Implementation

Midterm project implementation 1

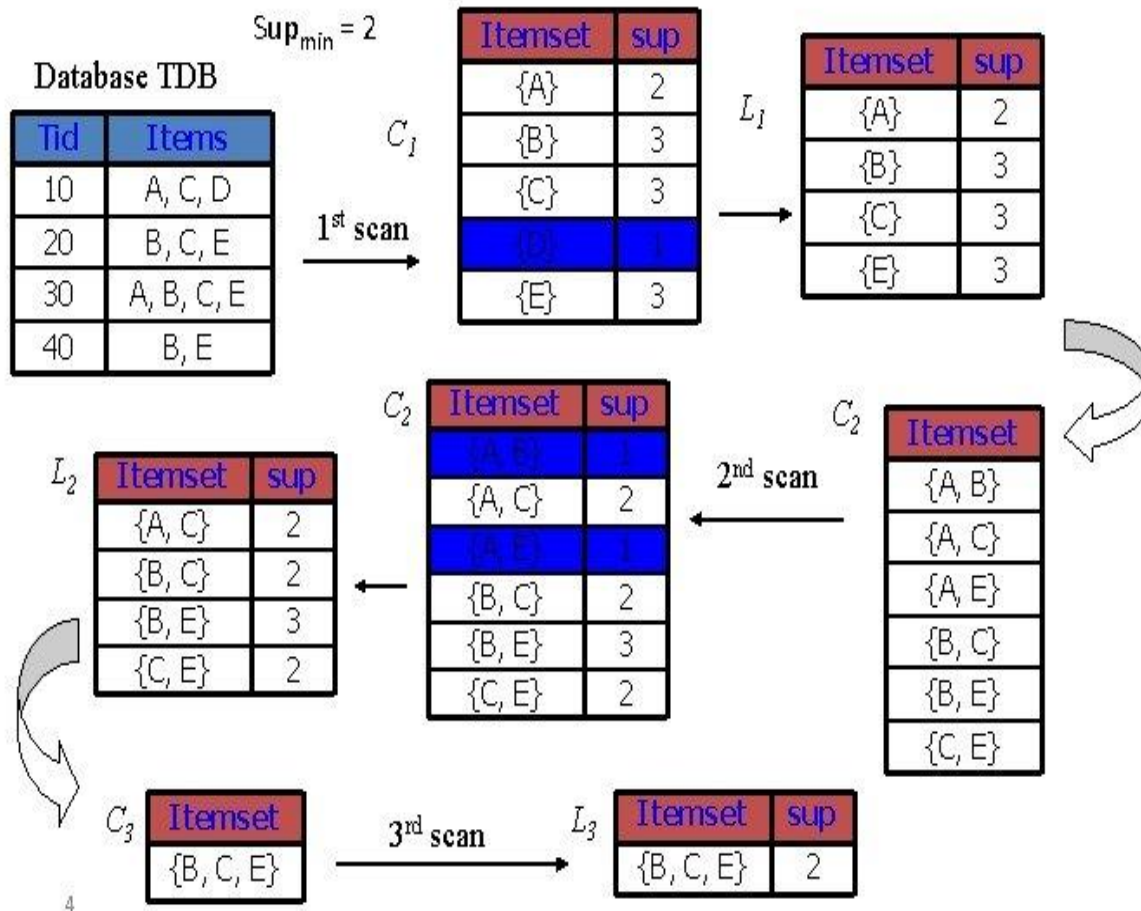
Himani Patel

Hpp5@njit.edu

Contents

Introduction.....	1
Apriori Algorithm Overview.....	1
Requirement.....	1
Software.....	1
Hardware.....	1
Name of Source Code Files.....	1
List of Data set Files.....	1
Implementation Overview.....	2
Implementation of algorithm in Python.....	2
Testing Implementation and Datasets.....	4
Console Mode.....	4
Amazon.....	4
BestBuy.....	5
KMart.....	5
Nike.....	6
Custom Dataset.....	7
Implementation Logs and Execution Output.....	7
GitHub Link.....	14

The Apriori Algorithm—An Example



Introduction

Apriori Algorithm Overview:

Apriori is designed to operate on databases containing transactions (for example, collections of items bought by customers, or details of a website frequentation). Other algorithms are designed for finding association rules in data having no transactions (Winepi and Minepi), or having no timestamps (DNA sequencing). Each transaction is seen as a set of items (an itemset). Given a threshold C , the Apriori algorithm identifies the item sets which are subsets of at least C transactions in the database.

Requirements:

Software Version:

Python 3.8.8

Visual Studio Code

Hardware:

Mac Os Version 13.0.1

Name of source code files:

Algo.py

List of dataset Files:

- 1) Amazon
- 2) BestBuy
- 3) K-mart
- 4) Nike
- 5) Custom Dataset

Implementation Overview

The implementation is built from scratch. It uses python as the main programming language. The implementation uses Database to store the items sets and transactions.

When you run the program, it will prompt you with a number of choices to choose from based on the dataset type, support and confidence.

```
(base) himanis-air:midterm-project himanipatel$ python algo.py
Hello, Welcome to association rule mining
Please choose the dataset for which you would like to get the Association rules output
[Please type
1 for Amazon
2 for BestBuy
3 for KMART
4 for Nike
5 for Custom
3
You selected (3)KMART Dataset
Enter the minimum support: 0.4
Enter the minimum Confidence: 0.2
```

Implementation of algorithm in Python:

Step 1: Importing the required libraries

```
EXPLORER  ...  algo.py U X
MIDTERM-PROJECT
  algo.py U
  amazon.csv U
  bestbuy.csv U
  grocery.csv U
  kmart.csv U
  nike.csv U

1 from csv import reader
2
3
4 def append_data():
5     #The purpose is to load the data set and return the dataset that contains a list of transactions and further, each transaction contains s...
```

Step 2: Loading and exploring the data

```
EXPLORER  ...  algo.py U X
MIDTERM-PROJECT
  algo.py U
  amazon.csv U
  bestbuy.csv U
  grocery.csv U
  kmart.csv U
  nike.csv U

48 def gen_Candidate1(dataset):
49     # Generate and return a set(dataset) which contains all frequent candidate 1-itemsets.
50     Can1 = set()
51     for t in dataset:
52         for item in t:
53             item_set = frozenset([item])
54             Can1.add(item_set)
55     return Can1
56 def apriori(Cank_item, LisK1):
57     #To check whether a frequent candidate k-itemset satisfy Apriori property and return true/false accordingly.
58     for item in Cank_item:
59         sub_Cank = Cank_item - frozenset([item])
60         if sub_Cank not in LisK1:
61             return False
62         return True
63
64
65
66
67
68
69
70
```

Step 3: Generating itemset k-1

```

88
89
90 amazon.csv U
91 bestbuy.csv U
92 grocery.csv U
93 kmart.csv U
94 nike.csv U
95
96
97 def gen_Lisk(dataset, Cank, minsup, sup_data):
98     #Generate and return a list/set which contains all frequent k-itemsets.
99
100     Lisk = set()
101     item_count = {}
102     for t in dataset:
103         for item in Cank:
104             if item.issubset(t):
105                 if item not in item_count:
106                     item_count[item] = 1
107                 else:
108                     item_count[item] += 1
109     t_num = float(len(dataset))
110     for item in item_count:
111         if (item_count[item] / t_num) >= minsup:
112             Lisk.add(item)
113             sup_data[item] = item_count[item] / t_num
114     return Lisk
115
116
117
118
119
120
121
122
123
124
125
126

```

Step 5: Generating Association rules

```
MIDTERM-PROJECT  algo.py
algo.py  U  126
amazon.csv  U  127
bestbuy.csv  U  128
grocery.csv  U  129
kmart.csv  U  130
nike.csv  U  131
132
133     def association_rules(List, sup_data, minconf):
134         #Generate and return association rules.
135
136         asso_rules = []
137         sublist = []
138         for i in range(0, len(List)):
139             for freq_set in List[i]:
140                 for sub_set in sublist:
141                     if sub_set.issubset(freq_set):
142                         conf = sup_data[freq_set] / sup_data[freq_set - sub_set]
143                         rule = (freq_set - sub_set, sub_set, conf)
144                         if conf >= minconf and rule not in asso_rules:
145                             # print freq_set-sub_set, " => ", sub_set, "conf: ", conf
146                             asso_rules.append(rule)
147                 sublist.append(freq_set)
148         return asso_rules
149
150
```


2) BestBuy

Running BestBuy data set with minimum support 0.5 and minimum confidence 0.5

```
(base) Himanis-MacBook-Air:midterm-project himanipatel$ python algo.py
Hello, welcome to association rule mining
Please choose the dataset for which you would like to get the Association rules
Please type
1 for Amazon
2 for BestBuy
3 for KMART
4 for Nike
5 for Custom
2
You selected BestBuy Dataset
Enter the minimum support: 0.5
Enter the minimum Confidence: 0.5
-----
frequent 7-itemsets      support
-----
frozenset({'Flash Drive'}) 0.65
frozenset({'Speakers'}) 0.55
frozenset({'Printer'}) 0.5
frozenset({'Anti-Virus'}) 0.7
frozenset({'Microsoft Office'}) 0.55
frozenset({'Lab Top'}) 0.6
frozenset({'Lab Top Case'}) 0.7
-----
frequent 7-itemsets      support
-----
frozenset({'Printer', 'Flash Drive'}) 0.5
frozenset({'Lab Top Case', 'Lab Top'}) 0.5
frozenset({'Lab Top Case', 'Anti-Virus'}) 0.6
frozenset({'Anti-Virus', 'Flash Drive'}) 0.5
frozenset({'Lab Top', 'Anti-Virus'}) 0.5
frozenset({'Flash Drive', 'Microsoft Office'}) 0.55
-----
frequent 7-itemsets      support
-----

-----
Association Rules with Confidence
-----
frozenset({'Printer'}) => frozenset({'Flash Drive'}) confidence: 1.0
frozenset({'Flash Drive'}) => frozenset({'Printer'}) confidence: 0.7692307692307692
frozenset({'Lab Top Case'}) => frozenset({'Lab Top'}) confidence: 0.7142857142857143
frozenset({'Lab Top'}) => frozenset({'Lab Top Case'}) confidence: 0.8333333333333334
frozenset({'Lab Top Case'}) => frozenset({'Anti-Virus'}) confidence: 0.8571428571428572
frozenset({'Anti-Virus'}) => frozenset({'Lab Top Case'}) confidence: 0.8571428571428572
frozenset({'Anti-Virus'}) => frozenset({'Flash Drive'}) confidence: 0.7142857142857143
frozenset({'Flash Drive'}) => frozenset({'Anti-Virus'}) confidence: 0.7692307692307692
frozenset({'Lab Top'}) => frozenset({'Anti-Virus'}) confidence: 0.8333333333333334
frozenset({'Anti-Virus'}) => frozenset({'Lab Top'}) confidence: 0.7142857142857143
frozenset({'Microsoft Office'}) => frozenset({'Flash Drive'}) confidence: 1.0
frozenset({'Flash Drive'}) => frozenset({'Microsoft Office'}) confidence: 0.8461538461538461
(base) Himanis-MacBook-Air:midterm-project himanipatel$
```

3) K-Mart

Running BestBuy data set with minimum support 0.4 and minimum confidence 0.3

```
(base) Himanis-MacBook-Air:midterm-project himanipatel$ python algo.py
Hello, welcome to association rule mining
Please choose the dataset for which you would like to get the Association rules
Please type
1 for Amazon
2 for BestBuy
3 for KMART
4 for Nike
5 for Custom
3
You selected KMART Dataset
Enter the minimum support: 0.4
Enter the minimum Confidence: 0.3
-----
frequent 4-itemsets      support
-----
frozenset({'Decorative Pillows'}) 0.45
frozenset({'Shamsâ€¢'}) 0.5
frozenset({'Bed Skirtsâ€¢'}) 0.55
frozenset({'Kids Bedding \n'}) 0.55
-----
frequent 4-itemsets      support
-----
frozenset({'Kids Bedding \n', 'Shamsâ€¢'}) 0.4
frozenset({'Bed Skirtsâ€¢', 'Shamsâ€¢'}) 0.4
frozenset({'Kids Bedding \n', 'Bed Skirtsâ€¢'}) 0.45
-----
frequent 4-itemsets      support
-----

-----
Association Rules with Confidence
-----
frozenset({'Kids Bedding \n'}) => frozenset({'Shamsâ€¢'}) confidence: 0.7272727272727273
frozenset({'Shamsâ€¢'}) => frozenset({'Kids Bedding \n'}) confidence: 0.8
frozenset({'Bed Skirtsâ€¢'}) => frozenset({'Shamsâ€¢'}) confidence: 0.7272727272727273
frozenset({'Shamsâ€¢'}) => frozenset({'Bed Skirtsâ€¢'}) confidence: 0.8
frozenset({'Kids Bedding \n'}) => frozenset({'Bed Skirtsâ€¢'}) confidence: 0.8181818181818181
frozenset({'Bed Skirtsâ€¢'}) => frozenset({'Kids Bedding \n'}) confidence: 0.8181818181818181
(base) Himanis-MacBook-Air:midterm-project himanipatel$
```


4) Nike

Running Nike data set with minimum support 0.5 and minimum confidence 0.6

```
((base) Himanis-MacBook-Air:midterm-project himanipatel$ python algo.py
Hello, welcome to association rule mining
Please choose the dataset for which you would like to get the Association rules
Please type
1 for Amazon
2 for BestBuy
3 for KMART
4 for Nike
5 for Custom
4
You selected Nike Dataset
Enter the minimum support: 0.5
Enter the minimum Confidence: 0.6
-----
frequent 7-itemsets      support
-----
frozenset({'Socks'}) 0.6
frozenset({'Dry Fit V-Nick'}) 0.5
frozenset({'Sweatshirts'}) 0.65
frozenset({'Rash Guard'}) 0.6
frozenset({'Running Shoe'}) 0.7
frozenset({'Swimming Shirt'}) 0.55
frozenset({'Modern Pants'}) 0.5
-----
frequent 7-itemsets      support
-----
frozenset({'Rash Guard', 'Dry Fit V-Nick'}) 0.5
frozenset({'Sweatshirts', 'Running Shoe'}) 0.55
frozenset({'Modern Pants', 'Sweatshirts'}) 0.5
frozenset({'Socks', 'Sweatshirts'}) 0.55
frozenset({'Socks', 'Running Shoe'}) 0.55
frozenset({'Rash Guard', 'Swimming Shirt'}) 0.5
-----
frequent 7-itemsets      support
-----
frozenset({'Socks', 'Sweatshirts', 'Running Shoe'}) 0.5

#-----#
Association Rules with Confidence
#-----#
frozenset({'Rash Guard'}) => frozenset({'Dry Fit V-Nick'}) confidence: 0.8333333333333334
frozenset({'Dry Fit V-Nick'}) => frozenset({'Rash Guard'}) confidence: 1.0
frozenset({'Running Shoe'}) => frozenset({'Sweatshirts'}) confidence: 0.7857142857142858
frozenset({'Sweatshirts'}) => frozenset({'Running Shoe'}) confidence: 0.8461538461538461
frozenset({'Modern Pants'}) => frozenset({'Sweatshirts'}) confidence: 1.0
frozenset({'Sweatshirts'}) => frozenset({'Modern Pants'}) confidence: 0.7692307692307692
frozenset({'Sweatshirts'}) => frozenset({'Socks'}) confidence: 0.8461538461538461
frozenset({'Socks'}) => frozenset({'Sweatshirts'}) confidence: 0.9166666666666667
frozenset({'Running Shoe'}) => frozenset({'Socks'}) confidence: 0.7857142857142858
frozenset({'Socks'}) => frozenset({'Running Shoe'}) confidence: 0.9166666666666667
frozenset({'Swimming Shirt'}) => frozenset({'Rash Guard'}) confidence: 0.9090909090909091
frozenset({'Rash Guard'}) => frozenset({'Swimming Shirt'}) confidence: 0.8333333333333334
frozenset({'Sweatshirts', 'Running Shoe'}) => frozenset({'Socks'}) confidence: 0.9090909090909091
frozenset({'Socks', 'Running Shoe'}) => frozenset({'Sweatshirts'}) confidence: 0.9090909090909091
frozenset({'Socks', 'Sweatshirts'}) => frozenset({'Running Shoe'}) confidence: 0.9090909090909091
frozenset({'Socks'}) => frozenset({'Sweatshirts', 'Running Shoe'}) confidence: 0.8333333333333334
frozenset({'Running Shoe'}) => frozenset({'Socks', 'Sweatshirts'}) confidence: 0.7142857142857143
frozenset({'Sweatshirts'}) => frozenset({'Socks', 'Running Shoe'}) confidence: 0.7692307692307692
(base) Himanis-MacBook-Air:midterm-project himanipatel$
```

5) Custom Dataset

Running Nike data set with minimum support 0.5 and minimum confidence 0.6

```
(base) Himanis-MacBook-Air:midterm-project himanipatel$ python algo.py
Hello, welcome to association rule mining
Please choose the dataset for which you would like to get the Association rules
Please type
1 for Amazon
2 for BestBuy
3 for KMART
4 for Nike
5 for Custom
5
You selected Custom Dataset
Enter the minimum support: 0.2
Enter the minimum Confidence: 0.1
-----
frequent 9-itemsets      support
-----
frozenset({'MAGGI'}) 0.25
frozenset({'CORNFLAKES'}) 0.3
frozenset({'MILK'}) 0.25
frozenset({'BOURNVITA'}) 0.2
frozenset({'BREAD'}) 0.65
frozenset({'SUGER'}) 0.3
frozenset({'COFFEE'}) 0.4
frozenset({'BISCUIT'}) 0.35
frozenset({'TEA'}) 0.35
-----
frequent 9-itemsets      support
-----
frozenset({'TEA', 'MAGGI'}) 0.2
frozenset({'CORNFLAKES', 'COFFEE'}) 0.2
frozenset({'SUGER', 'COFFEE'}) 0.2
frozenset({'MILK', 'BREAD'}) 0.2
frozenset({'BISCUIT', 'BREAD'}) 0.2
frozenset({'SUGER', 'BREAD'}) 0.2
frozenset({'TEA', 'BREAD'}) 0.2
-----
frequent 9-itemsets      support
-----

-----
Association Rules with Confidence
-----
frozenset({'TEA'}) => frozenset({'MAGGI'}) confidence: 0.5714285714285715
frozenset({'MAGGI'}) => frozenset({'TEA'}) confidence: 0.8
frozenset({'CORNFLAKES'}) => frozenset({'COFFEE'}) confidence: 0.5
frozenset({'COFFEE'}) => frozenset({'SUGER'}) confidence: 0.5
frozenset({'SUGER'}) => frozenset({'COFFEE'}) confidence: 0.6666666666666667
frozenset({'BREAD'}) => frozenset({'MILK'}) confidence: 0.3076923076923077
frozenset({'MILK'}) => frozenset({'BREAD'}) confidence: 0.8
frozenset({'BISCUIT'}) => frozenset({'BREAD'}) confidence: 0.5714285714285715
frozenset({'BREAD'}) => frozenset({'BISCUIT'}) confidence: 0.3076923076923077
frozenset({'SUGER'}) => frozenset({'BREAD'}) confidence: 0.6666666666666667
frozenset({'BREAD'}) => frozenset({'SUGER'}) confidence: 0.3076923076923077
frozenset({'TEA'}) => frozenset({'BREAD'}) confidence: 0.5714285714285715
frozenset({'BREAD'}) => frozenset({'TEA'}) confidence: 0.3076923076923077
(base) Himanis-MacBook-Air:midterm-project himanipatel$
```

Implementation Logs and Execution Output:

For Amazon:

```
(base) Himanis-MacBook-Air:midterm-project himanipatel$ python algo.py
Hello, Welcome to association rule mining
Please choose the dataset for which you would like to get the Association rules output
Please type
1 for Amazon
2 for BestBuy
3 for KMART
4 for Nike
5 for Custom
1
You selected (1)Amazon Dataset
Enter the minimum support: 0.4
Enter the minimum Confidence: 0.4
-----
frequent 5-itemsets      support
-----
frozenset({'Java: The Complete Reference'}) 0.45
```


For BestBuy:

(base) Himanis-MacBook-Air:midterm-project himanipatel\$ python algo.py

Hello, Welcome to association rule mining

Please choose the dataset for which you would like to get the Association rules output

Please type

1 for Amazon

2 for BestBuy

3 for KMART

4 for Nike

5 for Custom

2

You selected (2)BestBuy Dataset

Enter the minimum support: 0.5

Enter the minimum Confidence: 0.5

frequent 7-itemsets support

frozenset({'Lab Top'}) 0.6

frozenset({'Printer'}) 0.5

frozenset({'Anti-Virus'}) 0.7

frozenset({'Flash Drive'}) 0.65

frozenset({'Lab Top Case'}) 0.7

frozenset({'Speakers'}) 0.55

frozenset({'Microsoft Office'}) 0.55

frequent 7-itemsets support

frozenset({'Printer', 'Flash Drive'}) 0.5

frozenset({'Anti-Virus', 'Lab Top Case'}) 0.6

frozenset({'Anti-Virus', 'Flash Drive'}) 0.5

frozenset({'Lab Top', 'Lab Top Case'}) 0.5

frozenset({'Microsoft Office', 'Flash Drive'}) 0.55

frozenset({'Lab Top', 'Anti-Virus'}) 0.5

frequent 7-itemsets support

#####

Association Rules with Confidence

#####

frozenset({'Flash Drive'}) => frozenset({'Printer'}) confidence: 0.7692307692307692

frozenset({'Printer'}) => frozenset({'Flash Drive'}) confidence: 1.0

frozenset({'Lab Top Case'}) => frozenset({'Anti-Virus'}) confidence: 0.8571428571428572

frozenset({'Anti-Virus'}) => frozenset({'Lab Top Case'}) confidence: 0.8571428571428572

For KMart:

Hello, Welcome to association rule mining

Please type

2 for BestBuy

4 for Nike

3

Enter the minimum support: 0.4

```
frozenset({'Kids Bedding \n'}) 0.55
```

```
frozenset({'Bed Skirts'}) 0.55
```

```
frozenset({'Shamsâ€™', 'Bed Skirtsâ€™'}) 0.4
```

#####

#####

For Nike:

Hello, Welcome to association rule mining

Please type

2 for BestBuy

4 for Nike

4

Enter the minimum support: 0.5

* * * * *

```
frozenset({'Rash Guard'}) 0.6
```

```
frozenset({'Swimming Shirt'}) 0.55
```

```
frozenset({'Dry Fit V-Nick'}) 0.5
```

```
frozenset({'Socks'}) 0.6
```

```
frozenset({'Modern Pants'}) 0.5
```

```
frozenset({'Running Shoe'}) 0.7
```

```
frozenset({'Rash Guard', 'Dry Fit V-Nick'}) 0.5
```

```
frozenset({'Rash Guard', 'Swimming Shirt'}) 0.5
```

```
frozenset({'Modern Pants', 'Sweatshirts'}) 0.5
```

```
frozenset({'Socks', 'Sweatshirts'}) 0.55
```

```
frozenset({'Running Shoe', 'Socks'}) 0.55
```

```

-----*
frequent 7-itemsets      support
-----*
frozenset({'Running Shoe', 'Socks', 'Sweatshirts'}) 0.5

```

```

#####-
Association Rules with Confidence
#####-
frozenset({'Running Shoe'}) => frozenset({'Sweatshirts'}) confidence: 0.7857142857142858
frozenset({'Sweatshirts'}) => frozenset({'Running Shoe'}) confidence: 0.8461538461538461
frozenset({'Dry Fit V-Nick'}) => frozenset({'Rash Guard'}) confidence: 1.0
frozenset({'Rash Guard'}) => frozenset({'Dry Fit V-Nick'}) confidence: 0.8333333333333334
frozenset({'Swimming Shirt'}) => frozenset({'Rash Guard'}) confidence: 0.9090909090909091
frozenset({'Rash Guard'}) => frozenset({'Swimming Shirt'}) confidence: 0.8333333333333334
frozenset({'Modern Pants'}) => frozenset({'Sweatshirts'}) confidence: 1.0
frozenset({'Sweatshirts'}) => frozenset({'Modern Pants'}) confidence: 0.7692307692307692
frozenset({'Socks'}) => frozenset({'Sweatshirts'}) confidence: 0.9166666666666667
frozenset({'Sweatshirts'}) => frozenset({'Socks'}) confidence: 0.8461538461538461
frozenset({'Running Shoe'}) => frozenset({'Socks'}) confidence: 0.7857142857142858
frozenset({'Socks'}) => frozenset({'Running Shoe'}) confidence: 0.9166666666666667
frozenset({'Running Shoe', 'Socks'}) => frozenset({'Sweatshirts'}) confidence:
0.9090909090909091
frozenset({'Running Shoe', 'Sweatshirts'}) => frozenset({'Socks'}) confidence:
0.9090909090909091
frozenset({'Socks', 'Sweatshirts'}) => frozenset({'Running Shoe'}) confidence:
0.9090909090909091
frozenset({'Socks'}) => frozenset({'Running Shoe', 'Sweatshirts'}) confidence:
0.8333333333333334
frozenset({'Running Shoe'}) => frozenset({'Socks', 'Sweatshirts'}) confidence:
0.7142857142857143
frozenset({'Sweatshirts'}) => frozenset({'Running Shoe', 'Socks'}) confidence:
0.7692307692307692

```

For CustomData:

(base) Himanis-MacBook-Air:midterm-project himanipatel\$ python algo.py

Hello, Welcome to association rule mining

Please choose the dataset for which you would like to get the Association rules output

Please type

1 for Amazon

2 for BestBuy

3 for KMART

4 for Nike

5 for Custom

5

You selected (5)Custom Dataset

Enter the minimum support: 0.2

Enter the minimum Confidence: 0.1

frequent 9-itemsets support

frozenset({'TEA'}) 0.35
frozenset({'SUGER'}) 0.3
frozenset({'COFFEE'}) 0.4
frozenset({'MILK'}) 0.25
frozenset({'CORNFLAKES'}) 0.3
frozenset({'BISCUIT'}) 0.35
frozenset({'MAGGI'}) 0.25
frozenset({'BOURNVITA'}) 0.2
frozenset({'BREAD'}) 0.65

frequent 9-itemsets support

frozenset({'COFFEE', 'SUGER'}) 0.2
frozenset({'BREAD', 'BISCUIT'}) 0.2
frozenset({'TEA', 'BREAD'}) 0.2
frozenset({'BREAD', 'MILK'}) 0.2
frozenset({'BREAD', 'SUGER'}) 0.2
frozenset({'COFFEE', 'CORNFLAKES'}) 0.2
frozenset({'TEA', 'MAGGI'}) 0.2

frequent 9-itemsets support

#####

Association Rules with Confidence

#####

frozenset({'COFFEE'}) => frozenset({'SUGER'}) confidence: 0.5
frozenset({'SUGER'}) => frozenset({'COFFEE'}) confidence: 0.6666666666666667
frozenset({'BREAD'}) => frozenset({'BISCUIT'}) confidence: 0.3076923076923077
frozenset({'BISCUIT'}) => frozenset({'BREAD'}) confidence: 0.5714285714285715
frozenset({'BREAD'}) => frozenset({'TEA'}) confidence: 0.3076923076923077
frozenset({'TEA'}) => frozenset({'BREAD'}) confidence: 0.5714285714285715
frozenset({'BREAD'}) => frozenset({'MILK'}) confidence: 0.3076923076923077
frozenset({'MILK'}) => frozenset({'BREAD'}) confidence: 0.8
frozenset({'BREAD'}) => frozenset({'SUGER'}) confidence: 0.3076923076923077

frozenset({'SUGER'}) => frozenset({'BREAD'}) confidence: 0.6666666666666667
frozenset({'CORNFLAKES'}) => frozenset({'COFFEE'}) confidence: 0.6666666666666667
frozenset({'COFFEE'}) => frozenset({'CORNFLAKES'}) confidence: 0.5
frozenset({'MAGGI'}) => frozenset({'TEA'}) confidence: 0.8
frozenset({'TEA'}) => frozenset({'MAGGI'}) confidence: 0.5714285714285715

Github Link: <https://github.com/Hpp5/CS-634-Midterm-project>