PA 3: Association Analysis - Apriori Algorithm

Report on Bread Basket items

Student Details

Shivani Manojkumar Panchiwala – 1001982478

Kuldip Rameshbhai Savaliya – 1001832000

Meghaben Ghanshyambhai Patel – 1002006777

Apriori Algorithm: To extract frequent items from the itemset and derive association rules, the apriori dataset is used.

The Apriori algorithm has the following two parameters:

1) Support: This refers to how frequently the element appears in the dataset.
2) Confidence: A conditional Probability of that item is confidence.

The Given Dataset:

Transaction ID	Items
1	Bread, Scandinavian
2	Scandinavian, Cookies
3	Scandinavian, Hot chocolate
4	Bread, Scandinavian, Cookies
5	Bread, Hot chocolate
6	Scandinavian, Hot chocolate
7	Bread, Hot chocolate
8	Bread, Scandinavian, Hot chocolate, Muffin
9	Bread, Scandinavian, Hot chocolate

Minimum Support = 2, Minimum Confidence = 50%
Find the frequently occurring item sets and create association rules now.

Step 1:

We'll refer to this below table as S1.

ITEMS	SUPPORT COUNT
Bread	6
Scandinavian	7
Hot chocolate	5
Cookies	2
Muffin	1

The following step is to eliminate every element whose support count is less than the minimal support.

- We'll refer to this below table as T1.

ITEMS	SUPPORT COUNT
Bread	6
Scandinavian	7
Hot chocolate	5
Cookies	2

Step 2:

- We will now produce S2. In this, we will first create the pair and then record their frequency.
- We'll refer to this below table as S2.

ITEMS	SUPPORT COUNT
{Bread, Scandinavian}	4
{Bread, Hot chocolate}	4
{Bread, Cookies}	1
{Scandinavian, Hot chocolate}	4
{Scandinavian, Cookies}	2
{Cookies, Muffin}	0

- We will now eliminate all rows with support counts below the minimal support count.
- We'll refer to this below table as T2.

ITEMS	SUPPORT COUNT
{Bread, Scandinavian}	4
{Bread, Hot chocolate}	4
{Scandinavian, Hot chocolate}	4
{Scandinavian, Cookies}	2

Step 3:

- We will now create an S3 table. This will provide the cumulative frequency of the three things.

ITEMS	SUPPORT COUNT
{Bread, Scandinavian, Hot chocolate}	2
{Scandinavian, Hot chocolate, Cookies}	1
{Bread, Hot chocolate, Cookies}	0
{Bread, Scandinavian, Cookies}	0

- We will now once again eliminate any rows with a support count below the required minimum.
- There is only one row left after that.

{Bread, Scandinavian, Hot chocolate}	2

Step 4:

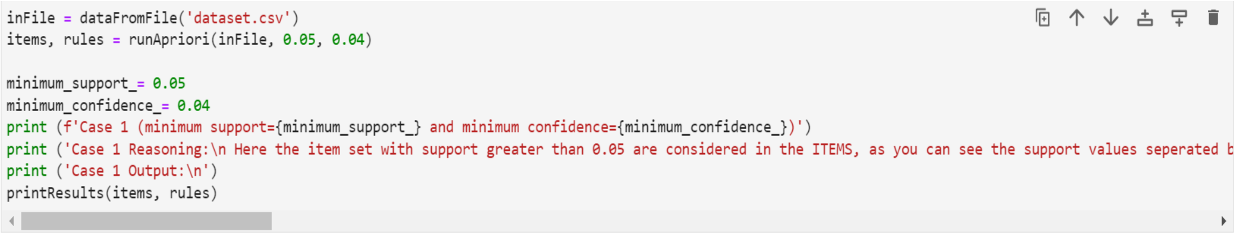
Finding the association rules for the subsets:

RULES	SUPPORT	CONFIDENCE
Bread ^ Scandinavian → Hot chocolate	2	Sup {(Bread ^ Scandinavian) ^ Hot chocolate}/Sup (Bread ^ Scandinavian) = 2/4 = 0.5 = 50%
Scandinavian ^ Hot chocolate → Bread	2	Sup {(Scandinavian ^ Hot chocolate) ^ Bread}/Sup (Scandinavian ^ Hot chocolate) = 2/4 = 0.5 = 50%
Bread ^ Hot chocolate → Scandinavian	2	Sup {(Bread ^ Hot chocolate) ^ Scandinavian}/Sup (Bread ^ Hot chocolate) = 2/4 = 0.5 = 50%
Hot chocolate → Bread ^ Scandinavian	2	Sup {(Hot chocolate ^ (Bread ^ Scandinavian)}/Sup (Hot chocolate) = 2/5 = 0.4 = 40%
Bread → Scandinavian ^ Hot chocolate	2	Sup {(Bread ^ (Scandinavian ^ Hot chocolate)}/Sup (Bread) = 2/6 = 0.33 = 33.33%
Scandinavian → Scandinavian ^ Hot chocolate	2	Sup {(Scandinavian ^ (Scandinavian ^ Hot chocolate)}/Sup (Scandinavian) = 2/7 = 0.28 = 28%

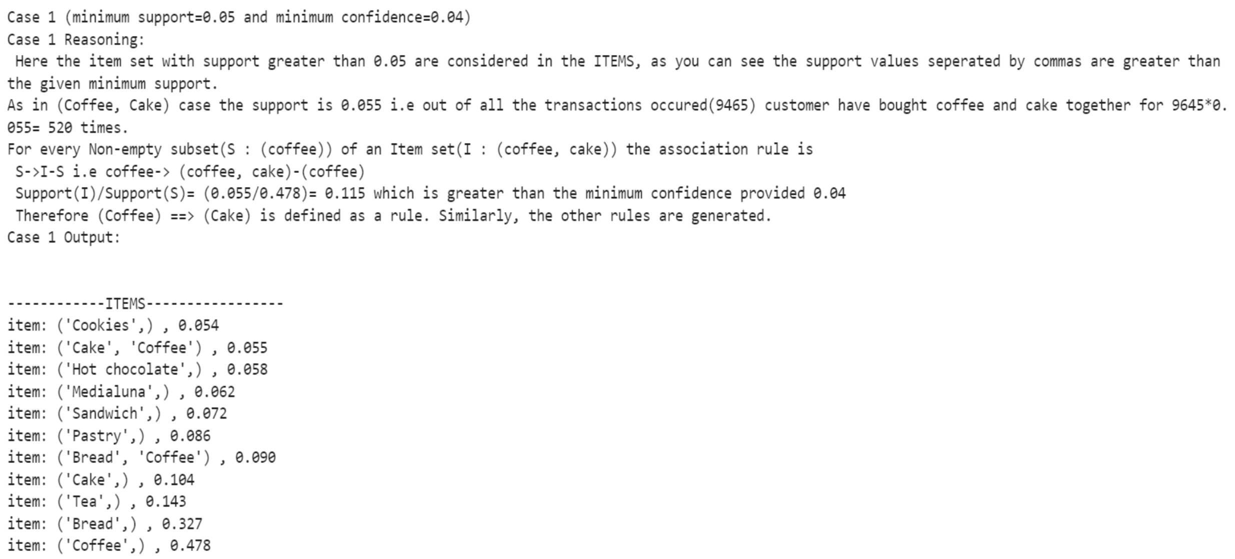
As a result, the stated minimum level of confidence is 50%, and the above table reveals that only three rows have a level of confidence more than or equal to the specified minimum level of confidence. So, the *Bread ^ Scandinavian → Scandinavian, Scandinavian ^ Scandinavian → Bread and Bread ^ Hot chocolate → Scandinavian,* can be considered as the **Strong association rules.**

Evaluated Results:

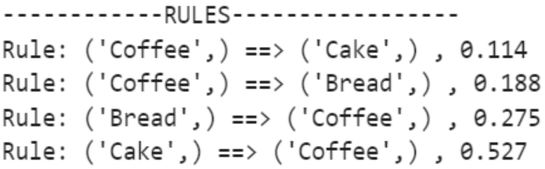
Case 1:



```
inFile = dataFromFile('dataset.csv')
items, rules = runApriori(inFile, 0.05, 0.04)

minimum_support_= 0.05
minimum_confidence_= 0.04
print (f'Case 1 (minimum support={minimum_support_} and minimum confidence={minimum_confidence_})')
print ('Case 1 Reasoning:\n Here the item set with support greater than 0.05 are considered in the ITEMS, as you can see the support values seperated b
print ('Case 1 Output:\n')
printResults(items, rules)
```

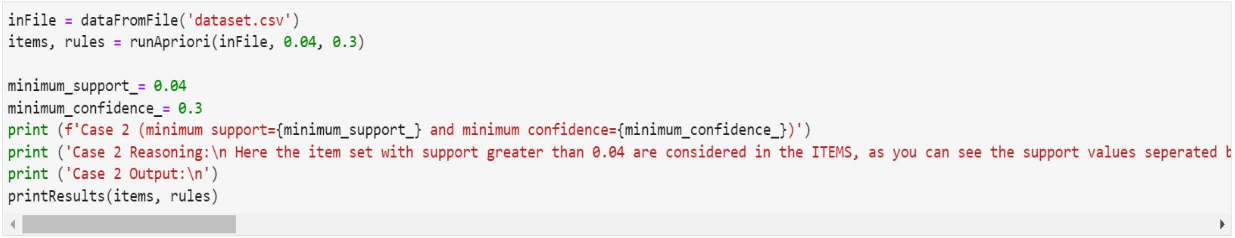


```
Case 1 (minimum support=0.05 and minimum confidence=0.04)
Case 1 Reasoning:
 Here the item set with support greater than 0.05 are considered in the ITEMS, as you can see the support values seperated by commas are greater than
the given minimum support.
As in (Coffee, Cake) case the support is 0.055 i.e out of all the transactions occured(9465) customer have bought coffee and cake together for 9645*0.
055= 520 times.
For every Non-empty subset(S : (coffee)) of an Item set(I : (coffee, cake)) the association rule is
 S->I-S i.e coffee-> (coffee, cake)-(coffee)
 Support(I)/Support(S)= (0.055/0.478)= 0.115 which is greater than the minimum confidence provided 0.04
 Therefore (Coffee) ==> (Cake) is defined as a rule. Similarly, the other rules are generated.
Case 1 Output:


------------ITEMS----------------
item: ('Cookies',) , 0.054
item: ('Cake', 'Coffee') , 0.055
item: ('Hot chocolate',) , 0.058
item: ('Medialuna',) , 0.062
item: ('Sandwich',) , 0.072
item: ('Pastry',) , 0.086
item: ('Bread', 'Coffee') , 0.090
item: ('Cake',) , 0.104
item: ('Tea',) , 0.143
item: ('Bread',) , 0.327
item: ('Coffee',) , 0.478
```

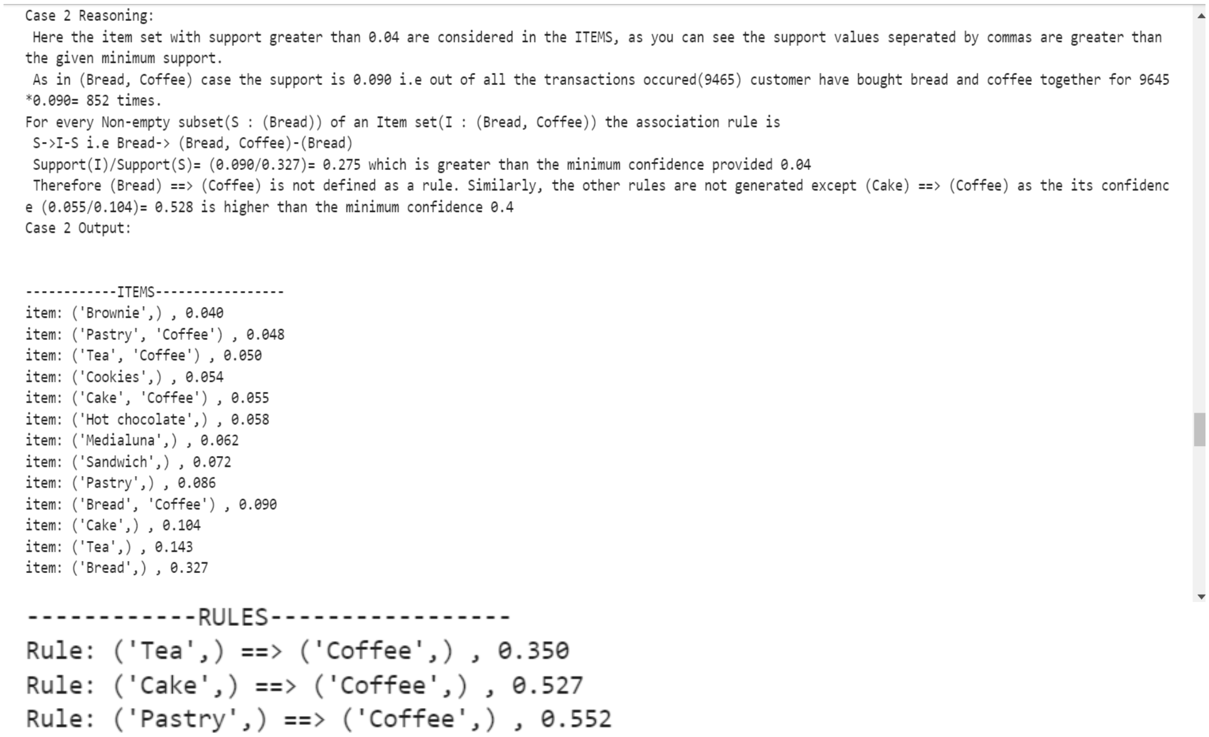


```
------------RULES-----------------
Rule: ('Coffee',) ==> ('Cake',) , 0.114
Rule: ('Coffee',) ==> ('Bread',) , 0.188
Rule: ('Bread',) ==> ('Coffee',) , 0.275
Rule: ('Cake',) ==> ('Coffee',) , 0.527
```

Case 2:

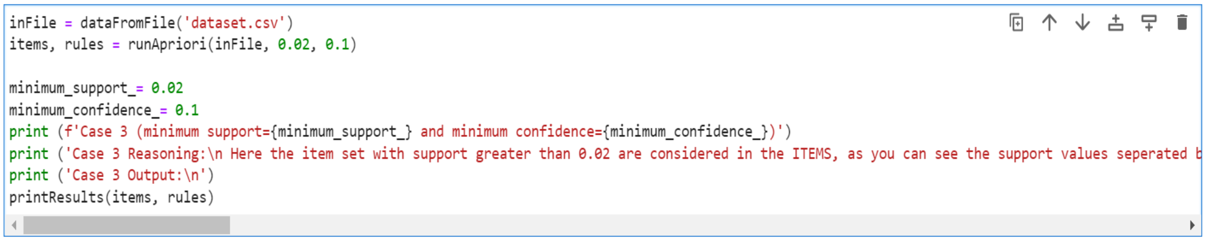


```
inFile = dataFromFile('dataset.csv')
items, rules = runApriori(inFile, 0.04, 0.3)

minimum_support_= 0.04
minimum_confidence_= 0.3
print (f'Case 2 (minimum support={minimum_support_} and minimum confidence={minimum_confidence_})')
print ('Case 2 Reasoning:\n Here the item set with support greater than 0.04 are considered in the ITEMS, as you can see the support values seperated b
print ('Case 2 Output:\n')
printResults(items, rules)
```

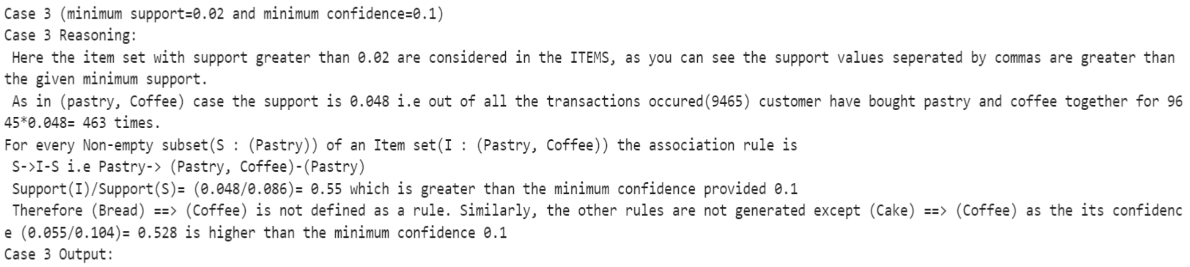


```
Case 2 Reasoning:
 Here the item set with support greater than 0.04 are considered in the ITEMS, as you can see the support values seperated by commas are greater than
the given minimum support.
 As in (Bread, Coffee) case the support is 0.090 i.e out of all the transactions occured(9465) customer have bought bread and coffee together for 9645
*0.090= 852 times.
For every Non-empty subset(S : (Bread)) of an Item set(I : (Bread, Coffee)) the association rule is
 S->I-S i.e Bread-> (Bread, Coffee)-(Bread)
 Support(I)/Support(S)= (0.090/0.327)= 0.275 which is greater than the minimum confidence provided 0.04
 Therefore (Bread) ==> (Coffee) is not defined as a rule. Similarly, the other rules are not generated except (Cake) ==> (Coffee) as the its confidenc
e (0.055/0.104)= 0.528 is higher than the minimum confidence 0.4
Case 2 Output:


------------ITEMS----------------
item: ('Brownie',) , 0.040
item: ('Pastry', 'Coffee') , 0.048
item: ('Tea', 'Coffee') , 0.050
item: ('Cookies',) , 0.054
item: ('Cake', 'Coffee') , 0.055
item: ('Hot chocolate',) , 0.058
item: ('Medialuna',) , 0.062
item: ('Sandwich',) , 0.072
item: ('Pastry',) , 0.086
item: ('Bread', 'Coffee') , 0.090
item: ('Cake',) , 0.104
item: ('Tea',) , 0.143
item: ('Bread',) , 0.327

------------RULES-----------------
Rule: ('Tea',) ==> ('Coffee',) , 0.350
Rule: ('Cake',) ==> ('Coffee',) , 0.527
Rule: ('Pastry',) ==> ('Coffee',) , 0.552
```

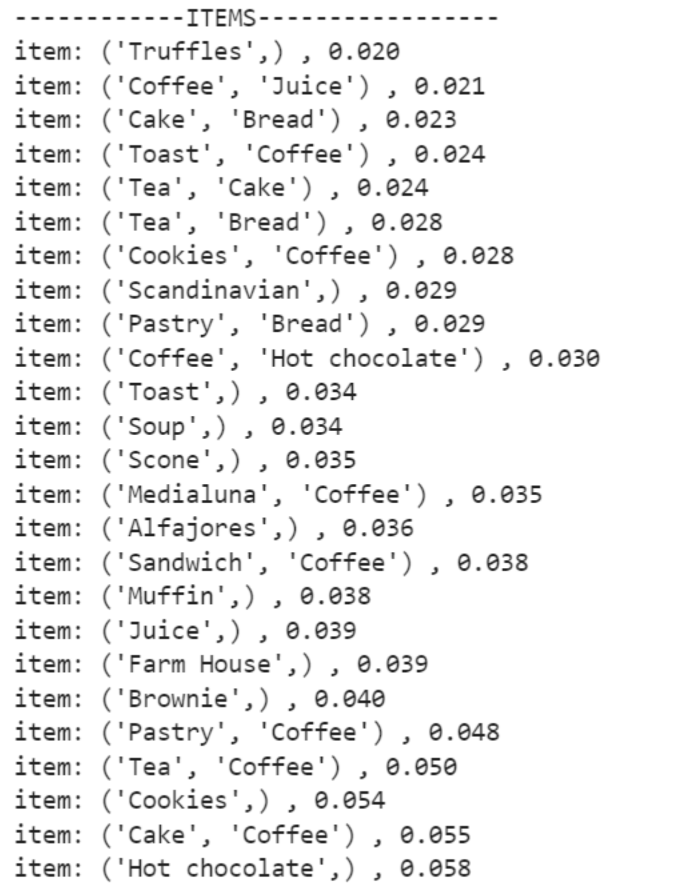
Case 3:



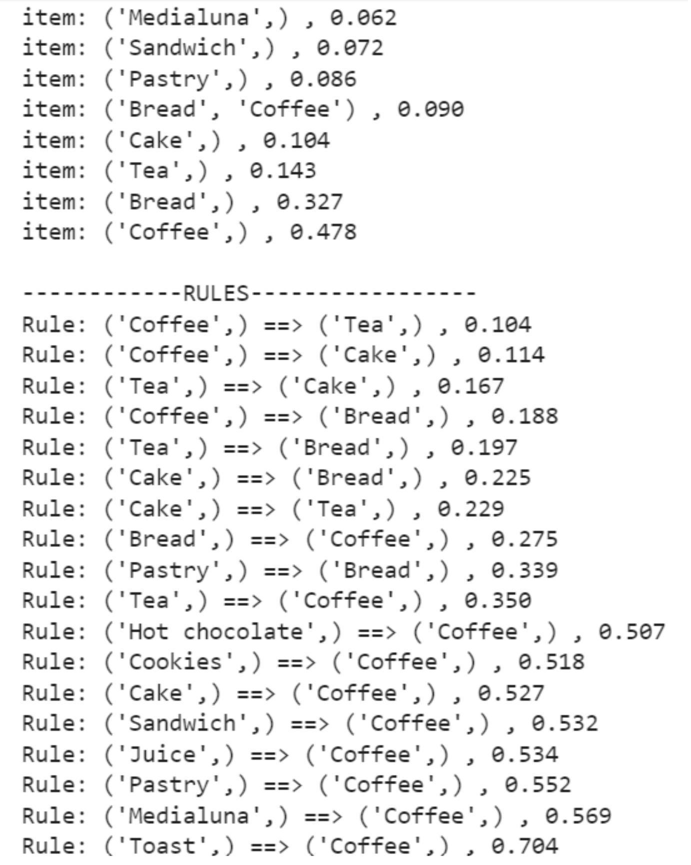
```
inFile = dataFromFile('dataset.csv')
items, rules = runApriori(inFile, 0.02, 0.1)

minimum_support_= 0.02
minimum_confidence_= 0.1
print (f'Case 3 (minimum support={minimum_support_} and minimum confidence={minimum_confidence_})')
print ('Case 3 Reasoning:\n Here the item set with support greater than 0.02 are considered in the ITEMS, as you can see the support values seperated b
print ('Case 3 Output:\n')
printResults(items, rules)
```



```
Case 3 (minimum support=0.02 and minimum confidence=0.1)
Case 3 Reasoning:
 Here the item set with support greater than 0.02 are considered in the ITEMS, as you can see the support values seperated by commas are greater than
the given minimum support.
 As in (pastry, Coffee) case the support is 0.048 i.e out of all the transactions occured(9465) customer have bought pastry and coffee together for 96
45*0.048= 463 times.
For every Non-empty subset(S : (Pastry)) of an Item set(I : (Pastry, Coffee)) the association rule is
 S->I-S i.e Pastry-> (Pastry, Coffee)-(Pastry)
 Support(I)/Support(S)= (0.048/0.086)= 0.55 which is greater than the minimum confidence provided 0.1
 Therefore (Bread) ==> (Coffee) is not defined as a rule. Similarly, the other rules are not generated except (Cake) ==> (Coffee) as the its confidenc
e (0.055/0.104)= 0.528 is higher than the minimum confidence 0.1
Case 3 Output:
```



```
------------ITEMS-----------------
item: ('Truffles',) , 0.020
item: ('Coffee', 'Juice') , 0.021
item: ('Cake', 'Bread') , 0.023
item: ('Toast', 'Coffee') , 0.024
item: ('Tea', 'Cake') , 0.024
item: ('Tea', 'Bread') , 0.028
item: ('Cookies', 'Coffee') , 0.028
item: ('Scandinavian',) , 0.029
item: ('Pastry', 'Bread') , 0.029
item: ('Coffee', 'Hot chocolate') , 0.030
item: ('Toast',) , 0.034
item: ('Soup',) , 0.034
item: ('Scone',) , 0.035
item: ('Medialuna', 'Coffee') , 0.035
item: ('Alfajores',) , 0.036
item: ('Sandwich', 'Coffee') , 0.038
item: ('Muffin',) , 0.038
item: ('Juice',) , 0.039
item: ('Farm House',) , 0.039
item: ('Brownie',) , 0.040
item: ('Pastry', 'Coffee') , 0.048
item: ('Tea', 'Coffee') , 0.050
item: ('Cookies',) , 0.054
item: ('Cake', 'Coffee') , 0.055
item: ('Hot chocolate',) , 0.058
```



```
item: ('Medialuna',) , 0.062
item: ('Sandwich',) , 0.072
item: ('Pastry',) , 0.086
item: ('Bread', 'Coffee') , 0.090
item: ('Cake',) , 0.104
item: ('Tea',) , 0.143
item: ('Bread',) , 0.327
item: ('Coffee',) , 0.478

------------RULES-----------------
Rule: ('Coffee',) ==> ('Tea',) , 0.104
Rule: ('Coffee',) ==> ('Cake',) , 0.114
Rule: ('Tea',) ==> ('Cake',) , 0.167
Rule: ('Coffee',) ==> ('Bread',) , 0.188
Rule: ('Tea',) ==> ('Bread',) , 0.197
Rule: ('Cake',) ==> ('Bread',) , 0.225
Rule: ('Cake',) ==> ('Tea',) , 0.229
Rule: ('Bread',) ==> ('Coffee',) , 0.275
Rule: ('Pastry',) ==> ('Bread',) , 0.339
Rule: ('Tea',) ==> ('Coffee',) , 0.350
Rule: ('Hot chocolate',) ==> ('Coffee',) , 0.507
Rule: ('Cookies',) ==> ('Coffee',) , 0.518
Rule: ('Cake',) ==> ('Coffee',) , 0.527
Rule: ('Sandwich',) ==> ('Coffee',) , 0.532
Rule: ('Juice',) ==> ('Coffee',) , 0.534
Rule: ('Pastry',) ==> ('Coffee',) , 0.552
Rule: ('Medialuna',) ==> ('Coffee',) , 0.569
Rule: ('Toast',) ==> ('Coffee',) , 0.704
```

References:

https://www.kaggle.com/datasets/mittalvasu95/the-bread-basket

https://www.geeksforgeeks.org/apriori-algorithm/

https://www.softwaretestinghelp.com/apriori-algorithm

https://www.educative.io/edpresso/what-is-the-apriori-algorithm