A.Y. 2022-2023

DATA MINING AND WAREHOUSE AYUSH JAIN

COMPUTER ENGINEERING | TE - B2 | 60004200132

EXPERIMENT - 6

Aim: Implementation of Association rule mining Using:

- 1. Apriori Algorithm,
- 2. FP Tree

Theory:

Association rule learning is a type of unsupervised learning technique that checks for the dependency of one data item on another data item and maps accordingly so that it can be more profitable. It tries to find some interesting relations or associations among the variables of the dataset. It is based on different rules to discover the interesting relations between variables in the database. Association rule learning is one of the very important concepts of machine learning, and it is employed in Market Basket analysis, Web usage mining, continuous production, etc. Here market basket analysis is a technique used by various big retailers to discover the associations between items. We can understand it by taking an example of a supermarket, as in a supermarket, all products that are purchased together are put together. For example, if a customer buys bread, he most likely can also buy butter, eggs, or milk, so these products are stored on a shelf or mostly nearby.

Association rule learning can be divided into three types of algorithms:

- 1. Apriori
- 2. Eclat
- 3. F-P Growth Algorithm

Association rule learning works on the concept of If and Else Statements, such as if A then B. Here the If the element is called antecedent, then the statement is called as Consequent. These types of relationships where we can find out some association or relation between two items are known as single cardinality. It is all about creating rules, and if the number of items increases, then cardinality also increases accordingly. So, to measure the associations between thousands of data items, there are several metrics. These metrics are given below:

- 1. Support
- 2. Confidence
- 3. Lift

NAAC Accredited with "A" Grade (CGPA: 3.18)



A.Y. 2022-2023

Implementation:

Connecting to drive from google.colab import drive drive.mount('/content/drive')

Importing and installing required python packages or libraries

!pip install apyori !pip install mlxtend --upgrade import numpy as np import pandas as pd import matplotlib.pyplot as plt import seaborn as sns from datetime import datetime from apyori import apriori from mlxtend.frequent_patterns import fpgrowth, association_rules from mlxtend.preprocessing import TransactionEncoder

Pre-processing df =
pd.read_csv('/content/drive/MyDrive/Groceries_dataset.csv')

df.head()

0 1808 21-07-2015 tropical fruit
1 2552 05-01-2015 whole milk
2 2300 19-09-2015 pip fruit
3 1187 12-12-2015 other vegetables
4 3037 01-02-2015 whole milk

df.info()

for i in df.columns: print(i)

Member_number
Date
itemDescription



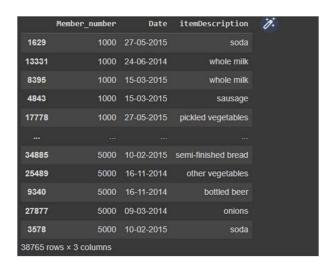


(Autonomous College Affiliated to the University of Mumbai) NAAC Accredited with "A" Grade (CGPA: 3.18)

A.Y. 2022-2023

Part A: Apriori

Sorting by members
df.sort_values(by = 'Member_number', inplace = True) df



Taking only required values

X = df.iloc[:,[0,2]].values

```
['soda',
   'whole milk',
   'whole milk',
   'sausage',
   'pickled vegetables',
   'canned beer',
   'yogurt',
   'misc. beverages',
   'salty snack',
   'sausage',
   'semi-finished bread',
   'hygiene articles',
   'pastry']
```

```
# Apriori
```

min_sup = float(input('Enter the minimum support: '))
min_con = float(input('Enter the minimum confidence: '))

```
Enter the minimum support: 0.002
Enter the minimum confidence: 0.2
```

DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING



(Autonomous College Affiliated to the University of Mumbai) NAAC Accredited with "A" Grade (CGPA: 3.18)

A.Y. 2022-2023

```
# Forming rules
rules = apriori(transactions = transactions, min_support = min_sup,
min_confidence = min_con, min_lift = 3, min_length = 2, max_length = 2) result
= list(rules) for i in result:
print(i)
RelationRecord(items=frozenset({'UHT-milk', 'kitchen towels'}),
support=0.002052861175263023,
ordered_statistics=[OrderedStatistic(items_base=frozenset({'kitchen towels'}),
items add=frozenset({'UHT-milk'}), confidence=0.32, lift=4.437864768683275)])
RelationRecord(items=frozenset(\(\frac{\text{rice}}{\text{rice}}\), support=0.0028226841159866563,
ordered statistics=[OrderedStatistic(items base=frozenset({'rice'}),
items_add=frozenset({'UHT-milk'}), confidence=0.2391304347826087,
lift=3.3163391613801645)])
RelationRecord(items=frozenset({beef', 'potato products'}),
support=0.002052861175263023,
ordered_statistics=[OrderedStatistic(items_base=frozenset({'potato products'}),
items add=frozenset({'beef'}), confidence=0.4210526315789474,
lift=3.8248067721751937)]) RelationRecord(items=frozenset({canned fruit', 'coffee'}),
support=0.002052861175263023,
ordered_statistics=[OrderedStatistic(items_base=frozenset({canned fruit}),
lift=4.287128712871287)]) RelationRecord(items=frozenset({'nuts/prunes', 'coffee'}),
support=0.002052861175263023,
ordered_statistics=[OrderedStatistic(items_base=frozenset({'nuts/prunes})
'}), items add=frozenset({'coffee'}), confidence=0.32, lift=3.086732673267327)])
RelationRecord(items=frozenset(\'\'napkins', 'rice'\'\), support=0.0028226841159866563,
ordered statistics=[OrderedStatistic(items base=frozenset({'rice'}),
items_add=frozenset({'napkins'}), confidence=0.2391304347826087,
lift=3.292902135504686)])
RelationRecord(items=frozenset({'waffles', 'sparkling wine'}),
support=0.0023094688221709007,
ordered_statistics=[OrderedStatistic(items_base=frozenset({\parkling wine})),
items_add=frozenset({'waffles'}), confidence=0.21951219512195122,
lift=3.491587854654057)])# Inspecting the result def inspect(result):
 lhs = [tuple(i[2][0][0])[0] for i in result]
rhs = [tuple(i[2][0][1])[0] for i in result]
support = [i[1] for i in result]
confidence = [i[2][0][2] for i in result]
lift = [i[2][0][3] for i in result
 return list(zip(lhs, rhs, support, confidence, lift))
```

data = pd.DataFrame(inspect(result), columns = ['Left Hand Side', 'Right Hand Side',



Shri Vile Parle Kelavani Mandal's

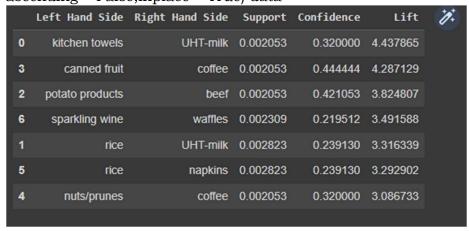
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING



(Autonomous College Affiliated to the University of Mumbai) NAAC Accredited with "A" Grade (CGPA: 3.18)

A.Y. 2022-2023

'Support', 'Confidence', 'Lift']) data.sort_values(by = 'Lift', ascending = False,inplace = True) data



Part B: FP Tree



fp_result = fpgrowth(fp_df, min_support = min_sup, use_colnames = True)
fp_result.tail(10)



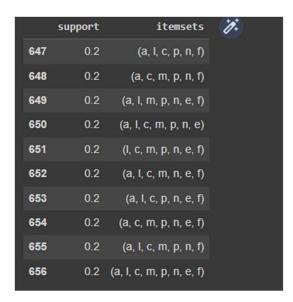
Shri Vile Parle Kelavani Mandal's

DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING



(Autonomous College Affiliated to the University of Mumbai) NAAC Accredited with "A" Grade (CGPA: 3.18)

A.Y. 2022-2023



rules_fp = association_rules(fp_result, metric="confidence", min_threshold=0.8)
rules_fp

* y-	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction	D.	
0	(a)	(f)	0.6	0.8	0.6	1.0	1.250000	0.12	inf		
1	(a)	(c)	0.6	0.8	0.6	1.0	1.250000	0.12	inf		
2	(c, a)	(f)	0.6	0.8	0.6	1.0	1.250000	0.12	inf		
3	(c, f)	(a)	0.6	0.6	0.6	1.0	1.666667	0.24	inf		
4	(a, f)	(c)	0.6	0.8	0.6	1.0	1.250000	0.12	inf		
9732	(n, e)	(a, I, c, m, p, f)	0.2	0.2	0.2	1.0	5.000000	0.16	inf		
9733	(n, f)	(a, I, c, m, p, e)	0.2	0.2	0.2	1.0	5.000000	0.16	inf		
9734	(e, f)	(a, I, c, m, p, n)	0.2	0.2	0.2	1.0	5.000000	0.16	inf		
9735	(n)	(a, I, c, m, p, e, f)	0.2	0.2	0.2	1.0	5.000000	0.16	inf		
9736	(e)	(a, I, c, m, p, n, f)	0.2	0.2	0.2	1.0	5.000000	0.16	inf		
9737 rows x 9 columns											

Conclusion:

Implemented Apriori and algorithm for a market basket analysis dataset and made an FP Tree for the given dataset.

.