Experiment 3.3

Implement Association Rules Mining

Student Name: SANSKAR AGRAWAL	**UID: 20BCS5914**
Branch: CSE	**Section/Group: MM_806-B**
Semester: 5th Semester	**Date of Performance: 06/11/2022**
Subject Name: Machine Learning Lab	**Subject Code: 20CSP-317**

Aim: Implement Association Rule Mining.

Objective: To prepare a model with Association Rule Mining.

Data Set Chosen: Association Rule Mining.

Result and output:

Implement Association Rules Minning

```
In [10]: import pandas as pd
         import numpy as np
         from mlxtend.frequent_patterns import apriori, association_rules
```

```
In [11]: df = pd.read_csv('GroceryStoreDataSet.csv', names = ['products'], sep = ',')
         df.head()
```

Out[11]:

	products
0	MILK,BREAD,BISCUIT
1	BREAD,MILK,BISCUIT,CORNFLAKES
2	BREAD,TEA,BOURNVITA
3	JAM,MAGGI,BREAD,MILK
4	MAGGI,TEA,BISCUIT

```
In [12]: df.shape
```

Out[12]: (20, 1)

```
In [13]: data = list(df["products"].apply(lambda x:x.split(",") ))
         data
```

```
Out[13]: [['MILK', 'BREAD', 'BISCUIT'],
         ['BREAD', 'MILK', 'BISCUIT', 'CORNFLAKES'],
         ['BREAD', 'TEA', 'BOURNVITA'],
         ['JAM', 'MAGGI', 'BREAD', 'MILK'],
         ['MAGGI', 'TEA', 'BISCUIT'],
         ['BREAD', 'TEA', 'BOURNVITA'],
         ['MAGGI', 'TEA', 'CORNFLAKES'],
         ['MAGGI', 'BREAD', 'TEA', 'BISCUIT'],
         ['JAM', 'MAGGI', 'BREAD', 'TEA'],
         ['BREAD', 'MILK'],
         ['COFFEE', 'COCK', 'BISCUIT', 'CORNFLAKES'],
         ['COFFEE', 'COCK', 'BISCUIT', 'CORNFLAKES'],
         ['COFFEE', 'SUGER', 'BOURNVITA'],
         ['BREAD', 'COFFEE', 'COCK'],
         ['BREAD', 'SUGER', 'BISCUIT'],
         ['COFFEE', 'SUGER', 'CORNFLAKES'],
         ['BREAD', 'SUGER', 'BOURNVITA'],
         ['BREAD', 'COFFEE', 'SUGER'],
         ['BREAD', 'COFFEE', 'SUGER'],
         ['TEA', 'MILK', 'COFFEE', 'CORNFLAKES']]
```

```
In [14]: from mlxtend.preprocessing import TransactionEncoder
         a = TransactionEncoder()
         a_data = a.fit(data).transform(data)
         df = pd.DataFrame(a_data,columns=a.columns_)
         df = df.replace(False,0)
         df
```

Out[14]:

	BISCUIT	BOURNVITA	BREAD	COCK	COFFEE	CORNFLAKES	JAM	MAGGI	MILK	SUGER	TEA
0	True	0	True	0	0	0	0	0	True	0	0
1	True	0	True	0	0	True	0	0	True	0	0
2	0	True	True	0	0	0	0	0	0	0	True
3	0	0	True	0	0	0	True	True	True	0	0
4	True	0	0	0	0	0	0	0	True	0	True
5	0	True	True	0	0	0	0	0	0	0	True
6	0	0	0	0	0	0	True	0	True	0	True
7	True	0	True	0	0	0	0	0	True	0	True
8	0	0	True	0	0	0	True	True	0	0	True
9	0	0	True	0	0	0	0	0	0	True	0
10	True	0	0	True	True	True	0	0	0	0	0
11	True	0	0	True	True	True	0	0	0	0	0
12	0	True	0	0	True	0	0	0	0	True	0
13	0	0	True	True	True	0	0	0	0	0	0

14	True	0	True	0	0	0	0	0	0	True	0	
15	0	0	0	0	True	True	0	0	0	True	0	
16	0	True	True	0	0	0	0	0	0	True	0	
17	0	0	True	0	True	0	0	0	0	True	0	
18	0	0	True	0	True	0	0	0	0	True	0	
19	0	0	0	0	True	True	0	0	True	0	True	

In [15]:
```python
df = apriori(df, min_support = 0.2, use_colnames = True, verbose = 1)
df
```

Processing 42 combinations | Sampling itemset size 3

/opt/anaconda3/lib/python3.9/site-packages/mlxtend/frequent_patterns/fpcommon.py:111: DeprecationWarning: DataFrames with non-bool types result in worse computationalperformance and their support might be discontinued in the future.Please use a DataFrame with bool type
  warnings.warn(

Out[15]:

	support	itemsets
0	0.35	(BISCUIT)
1	0.2	(BOURNVITA)
2	0.65	(BREAD)
3	0.4	(COFFEE)
4	0.3	(CORNFLAKES)
5	0.25	(MAGGI)
6	0.25	(MILK)
7	0.3	(SUGER)
8	0.35	(TEA)
9	0.2	(BISCUIT, BREAD)
10	0.2	(BREAD, MILK)
11	0.2	(BREAD, SUGER)
12	0.2	(BREAD, TEA)
13	0.2	(COFFEE, CORNFLAKES)
14	0.2	(SUGER, COFFEE)
15	0.2	(MAGGI, TEA)

In [16]:
```python
df_ar = association_rules(df, metric = "confidence", min_threshold = 0.6)
df_ar
```

Out[16]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
0	(MILK)	(BREAD)	0.25	0.65	0.2	0.800000	1.230769	0.0375	1.75
1	(SUGER)	(BREAD)	0.30	0.65	0.2	0.666667	1.025641	0.0050	1.05
2	(CORNFLAKES)	(COFFEE)	0.30	0.40	0.2	0.666667	1.666667	0.0800	1.80
3	(SUGER)	(COFFEE)	0.30	0.40	0.2	0.666667	1.666667	0.0800	1.80
4	(MAGGI)	(TEA)	0.25	0.35	0.2	0.800000	2.285714	0.1125	3.25