Assignment 5 – Association Rule Mining

Problem Statement:

You work in XYZ Company as a Python. The company officials want you to write code for Association Rule Mining. Dataset: retail_dataset.csv

Tasks To Be Performed:

1. Using pandas import the dataset as DataFrame
2. Install the mixtend library to use Apriori and association rule mining
3. Using the Apriori algorithm generate a list of items frequently bought together
4. Generate the association rules for the given items from the Apriori algorithm

In [1]:

```
import pandas as pd
from sklearn.preprocessing import MultiLabelBinarizer
from mlxtend.frequent_patterns import apriori
from mlxtend.frequent_patterns import association_rules
```

In [2]:

```
retail_shopping = {'ID': [1,2,3,4,5,6,7,8,9,10], 'Basket': [['Pencils', 'Markers', 'Highlighters', 'Papers'],
                                                             ['Markers', 'Erasers'],
                                                             ['Staplerpins', 'Papers', 'Erasers', 'Cardholders', 'Highlighters'],
                                                             ['Papers', 'Erasers', 'Cardholders'],
                                                             ['Markers', 'Postit', 'Erasers'],
                                                             ['Envelop'],
                                                             ['Markers', 'Erasers'],
                                                             ['Pencils', 'Markers', 'Staplerpins', 'Postit', 'Highlighters', 'Papers', 'Erasers'],
                                                             ['Staplerpins', 'Postit', 'Markers', 'Erasers'],
                                                             ['Envelop']
                                                             ]
                    }
df = pd.DataFrame(retail_shopping)
df
```

Out[2]:

	ID	Basket
0	1	[Pencils, Markers, Highlighters, Papers]
1	2	[Markers, Erasers]
2	3	[Staplerpins, Papers, Erasers, Cardholders, Hi...
3	4	[Papers, Erasers, Cardholders]
4	5	[Markers, Postit, Erasers]
5	6	[Envelop]
6	7	[Markers, Erasers]
7	8	[Pencils, Markers, Staplerpins, Postit, Highli...
8	9	[Staplerpins, Postit, Markers, Erasers]
9	10	[Envelop]

In [3]:

```
pip install mlxtend
```

```
Requirement already satisfied: mlxtend in c:\users\roy\anaconda3\lib\site-packages (0.22.0)
Requirement already satisfied: matplotlib>=3.0.0 in c:\users\roy\anaconda3\lib\site-packages (from mlxtend) (3.7.0)
Requirement already satisfied: numpy>=1.16.2 in c:\users\roy\anaconda3\lib\site-packages (from mlxtend) (1.23.5)
Requirement already satisfied: setuptools in c:\users\roy\anaconda3\lib\site-packages (from mlxtend) (65.6.3)
Requirement already satisfied: pandas>=0.24.2 in c:\users\roy\anaconda3\lib\site-packages (from mlxtend) (1.5.3)
Requirement already satisfied: scipy>=1.2.1 in c:\users\roy\anaconda3\lib\site-packages (from mlxtend) (1.10.0)
Requirement already satisfied: scikit-learn>=1.0.2 in c:\users\roy\anaconda3\lib\site-packages (from mlxtend) (1.2.1)
Requirement already satisfied: joblib>=0.13.2 in c:\users\roy\anaconda3\lib\site-packages (from mlxtend) (1.1.1)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\roy\anaconda3\lib\site-packages (from matplotlib>=3.0.0->mlxtend) (1.0.5)
Requirement already satisfied: packaging>=20.0 in c:\users\roy\anaconda3\lib\site-packages (from matplotlib>=3.0.0->mlxtend) (22.0)
Requirement already satisfied: pillow>=6.2.0 in c:\users\roy\anaconda3\lib\site-packages (from matplotlib>=3.0.0->mlxtend) (9.4.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\roy\anaconda3\lib\site-packages (from matplotlib>=3.0.0->mlxtend) (1.4.4)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\roy\anaconda3\lib\site-packages (from matplotlib>=3.0.0->mlxtend) (3.0.9)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\roy\anaconda3\lib\site-packages (from matplotlib>=3.0.0->mlxtend) (4.25.0)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\roy\anaconda3\lib\site-packages (from matplotlib>=3.0.0->mlxtend) (2.8.2)
Requirement already satisfied: cycler>=0.10 in c:\users\roy\anaconda3\lib\site-packages (from matplotlib>=3.0.0->mlxtend) (0.11.0)
Requirement already satisfied: pytz>=2020.1 in c:\users\roy\anaconda3\lib\site-packages (from pandas>=0.24.2->mlxtend) (2022.7)
Requirement already satisfied: threadpoolctl>=2.0.0 in c:\users\roy\anaconda3\lib\site-packages (from scikit-learn>=1.0.2->mlxtend) (2.2.0)
Requirement already satisfied: six>=1.5 in c:\users\roy\anaconda3\lib\site-packages (from python-dateutil>=2.7->matplotlib>=3.0.0->mlxtend) (1.16.0)
Note: you may need to restart the kernel to use updated packages.
```

In [4]:

```
mlb = MultiLabelBinarizer()
encoded_basket = mlb.fit_transform(df['Basket'])
encoded_df = pd.DataFrame(encoded_basket, columns=mlb.classes_)
encoded_df
```

Out[4]:

	Cardholders	Envelop	Erasers	Highlighters	Markers	Papers	Pencils	Postit	Staplerpins
0	0	0	0	1	1	1	1	0	0
1	0	0	1	0	1	0	0	0	0
2	1	0	1	1	0	1	0	0	1
3	1	0	1	0	0	1	0	0	0
4	0	0	1	0	1	0	0	1	0
5	0	1	0	0	0	0	0	0	0
6	0	0	1	0	1	0	0	0	0
7	0	0	1	1	1	1	1	1	1
8	0	0	1	0	1	0	0	1	1
9	0	1	0	0	0	0	0	0	0

In [5]:

```
# Apply Apriori algorithm to find frequent itemsets
frequent_itemsets = apriori(encoded_df, min_support=0.2, use_colnames=True)
frequent_itemsets
```

```
C:\Users\Roy\anaconda3\lib\site-packages\mlxtend\frequent_patterns\fpcommon.py:110: 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[5]:

	support	itemsets
0	0.2	(Cardholders)
1	0.2	(Envelop)
2	0.7	(Erasers)
3	0.3	(Highlighters)
4	0.6	(Markers)
5	0.4	(Papers)
6	0.2	(Pencils)
7	0.3	(Postit)
8	0.3	(Staplerpins)
9	0.2	(Cardholders, Erasers)
10	0.2	(Cardholders, Papers)
11	0.2	(Highlighters, Erasers)
12	0.5	(Markers, Erasers)
13	0.3	(Erasers, Papers)
14	0.3	(Postit, Erasers)
15	0.3	(Staplerpins, Erasers)
16	0.2	(Markers, Highlighters)
17	0.3	(Highlighters, Papers)
18	0.2	(Pencils, Highlighters)
19	0.2	(Staplerpins, Highlighters)
20	0.2	(Markers, Papers)
21	0.2	(Markers, Pencils)
22	0.3	(Markers, Postit)
23	0.2	(Markers, Staplerpins)
24	0.2	(Pencils, Papers)
25	0.2	(Staplerpins, Papers)
26	0.2	(Staplerpins, Postit)
27	0.2	(Cardholders, Erasers, Papers)
28	0.2	(Highlighters, Erasers, Papers)
29	0.2	(Staplerpins, Highlighters, Erasers)
30	0.3	(Markers, Postit, Erasers)
31	0.2	(Markers, Staplerpins, Erasers)
32	0.2	(Staplerpins, Erasers, Papers)
33	0.2	(Staplerpins, Postit, Erasers)
34	0.2	(Markers, Highlighters, Papers)
35	0.2	(Pencils, Markers, Highlighters)
36	0.2	(Pencils, Highlighters, Papers)
37	0.2	(Staplerpins, Highlighters, Papers)
38	0.2	(Markers, Pencils, Papers)
39	0.2	(Markers, Staplerpins, Postit)
40	0.2	(Staplerpins, Highlighters, Erasers, Papers)
41	0.2	(Markers, Staplerpins, Postit, Erasers)
42	0.2	(Pencils, Markers, Highlighters, Papers)

In [6]:

```
association_rules_df = association_rules(frequent_itemsets, metric="lift", min_threshold=1.0)
association_rules_df
```

Out[6]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction	zhangs_metric
0	(Cardholders)	(Erasers)	0.2	0.7	0.2	1.000000	1.428571	0.06	inf	0.375
1	(Erasers)	(Cardholders)	0.7	0.2	0.2	0.285714	1.428571	0.06	1.12	1.000
2	(Cardholders)	(Papers)	0.2	0.4	0.2	1.000000	2.500000	0.12	inf	0.750
3	(Papers)	(Cardholders)	0.4	0.2	0.2	0.500000	2.500000	0.12	1.60	1.000
4	(Markers)	(Erasers)	0.6	0.7	0.5	0.833333	1.190476	0.08	1.80	0.400
...	...	...	...	...	...	...	...	...	...	...
145	(Highlighters, Papers)	(Markers, Pencils)	0.3	0.2	0.2	0.666667	3.333333	0.14	2.40	1.000
146	(Pencils)	(Markers, Highlighters, Papers)	0.2	0.2	0.2	1.000000	5.000000	0.16	inf	1.000
147	(Markers)	(Pencils, Highlighters, Papers)	0.6	0.2	0.2	0.333333	1.666667	0.08	1.20	1.000
148	(Highlighters)	(Markers, Pencils, Papers)	0.3	0.2	0.2	0.666667	3.333333	0.14	2.40	1.000
149	(Papers)	(Markers, Pencils, Highlighters)	0.4	0.2	0.2	0.500000	2.500000	0.12	1.60	1.000

150 rows × 10 columns

In [ ]: