**Experiment 06**

**Aim:** Implementation of Association rule mining

**Apriori Algorithm**

**Code:**

import pandas as pd

import numpy as np

from mlxtend.frequent\_patterns import apriori, association\_rules

from mlxtend.preprocessing import TransactionEncoder

df = pd.read\_csv('../content/sample\_data/GroceryStoreDataSet.csv', names = ['products'], sep = ',')

df.head()

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

data

a = TransactionEncoder()

a\_data = a.fit(data).transform(data)

df = pd.DataFrame(a\_data,columns=a.columns\_)

df = df.replace(False,0)

df

df = apriori(df, min\_support = 0.2, use\_colnames = True, verbose = 1)

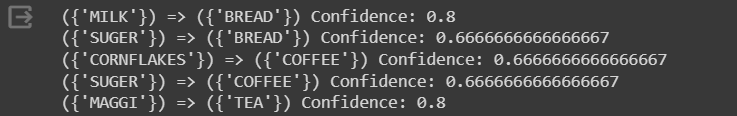
df

df\_ar = association\_rules(df, metric = "confidence", min\_threshold = 0.6)

df\_ar

for i in range(len(df\_ar)):

    print("",str(df\_ar.loc[i, "antecedents"]).replace("frozenset",""),"=>",str(df\_ar.loc[i, "consequents"]).replace("frozenset",""),"Confidence:",str(df\_ar.loc[i, "confidence"]).replace("frozenset",""))

**Output:**

**FP Tree**

**Code:**

import pandas as pd

from mlxtend.preprocessing import TransactionEncoder

from mlxtend.frequent\_patterns import fpgrowth

from mlxtend.frequent\_patterns import association\_rules

dataset = [['f', 'a', 'c', 'd', 'g', 'i', 'm', 'p'],

           ['a', 'b', 'c', 'f', 'l', 'm', 'o'],

           ['b', 'f', 'h', 'j', 'o', 'w'],

           ['b', 'c', 'k', 's', 'p'],

           ['a', 'f', 'c', 'e', 'l', 'p','m', 'n']]

te = TransactionEncoder()

te\_ary = te.fit(dataset).transform(dataset)

df = pd.DataFrame(te\_ary, columns=te.columns\_)

df

result = fpgrowth(df, min\_support=0.6,use\_colnames=True)

df\_ar = association\_rules(result, metric="confidence", min\_threshold=0.8)

for i in range(len(df\_ar)):

    print("",str(df\_ar.loc[i, "antecedents"]).replace("frozenset",""),"=>",str(df\_ar.loc[i, "consequents"]).replace("frozenset",""),"Confidence:",str(df\_ar.loc[i, "confidence"]).replace("frozenset",""))

**Output:**

