1tqii0cft

April 29, 2024

1 ****1. We import the necessary libraries required for the implementation****

[ ]:

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
```

2 ****2. Load the dataset****

****Now we have to proceed by reading the dataset we have, that is in a csv format. We do that using pandas module's read_csv function [6].****

[ ]:

```
dataset = pd.read_csv("../input/market-basket/Market_Basket_Optimisation.csv",␣
  ↪header = None)
transactions = []
for i in range(0, 7501):
    transactions.append([str(dataset.values[i,j]) for j in range(0,20)])
```

[ ]:

```

```

3 ****3. Take a glance at the records****

[ ]:

```
dataset
```

[ ]:

```
                   0                  1            2                  3  \
0             shrimp            almonds      avocado     vegetables mix
1            burgers          meatballs         eggs                NaN
2            chutney                NaN          NaN                NaN
3             turkey            avocado          NaN                NaN
4      mineral water               milk   energy bar  whole wheat rice
…                  …                  …            …                  …
7496          butter         light mayo  fresh bread                NaN
7497         burgers  frozen vegetables         eggs       french fries
7498         chicken                NaN          NaN                NaN
7499       escalope           green tea          NaN                NaN
```

```
7500           eggs    frozen smoothie  yogurt cake    low fat yogurt

                 4                 5     6               7             8  \
0     green grapes  whole weat flour  yams  cottage cheese  energy drink
1              NaN               NaN   NaN             NaN           NaN
2              NaN               NaN   NaN             NaN           NaN
3              NaN               NaN   NaN             NaN           NaN
4        green tea               NaN   NaN             NaN           NaN
…            …               …   …             …           …
7496           NaN               NaN   NaN             NaN           NaN
7497     magazines         green tea   NaN             NaN           NaN
7498           NaN               NaN   NaN             NaN           NaN
7499           NaN               NaN   NaN             NaN           NaN
7500           NaN               NaN   NaN             NaN           NaN

                 9              10         11     12     13             14  \
0     tomato juice  low fat yogurt  green tea  honey  salad  mineral water
1              NaN             NaN        NaN    NaN    NaN            NaN
2              NaN             NaN        NaN    NaN    NaN            NaN
3              NaN             NaN        NaN    NaN    NaN            NaN
4              NaN             NaN        NaN    NaN    NaN            NaN
…            …             …        …    …    …            …
7496           NaN             NaN        NaN    NaN    NaN            NaN
7497           NaN             NaN        NaN    NaN    NaN            NaN
7498           NaN             NaN        NaN    NaN    NaN            NaN
7499           NaN             NaN        NaN    NaN    NaN            NaN
7500           NaN             NaN        NaN    NaN    NaN            NaN

          15                 16               17       18         19
0     salmon  antioxydant juice  frozen smoothie  spinach  olive oil
1        NaN                NaN              NaN      NaN        NaN
2        NaN                NaN              NaN      NaN        NaN
3        NaN                NaN              NaN      NaN        NaN
4        NaN                NaN              NaN      NaN        NaN
…      …                …              …      …        …
7496     NaN                NaN              NaN      NaN        NaN
7497     NaN                NaN              NaN      NaN        NaN
7498     NaN                NaN              NaN      NaN        NaN
7499     NaN                NaN              NaN      NaN        NaN
7500     NaN                NaN              NaN      NaN        NaN

[7501 rows x 20 columns]
```

4 ****4. Look at the shape****

[ ]:
```
dataset.shape
```

[ ]:
```
(7501, 20)
```

5 ****5. Convert Pandas DataFrame into a list of lists****

[ ]:
```
for i in range(0, 7501):
    transactions.append([str(dataset.values[i,j]) for j in range(0,20)])
```

[ ]:
```
from apyori import apriori
rules = apriori(transactions = transactions, min_support = 0.003,␣
 ↪min_cinfidence = 0.2, min_lift = 3, min_length = 2, max_length = 2)
```

6 ****6. Print out the number of rules as list****

[ ]:
```
results = list(rules)
```

7 ****7. Visualizing the results****

****In the LHS variable, we store the first item from all the results, from which we obtain the second item that is bought after that item is already bought, which is now stored in the RHS variable. The supports, confidences and lifts store all the support, confidence and lift values from the results [6].****

[ ]:
```
def inspect(results):
    lhs         =[tuple(result[2][0][0])[0] for result in results]
    rhs         =[tuple(result[2][0][1])[0] for result in results]
    supports    =[result[1] for result in results]
    confidences =[result[2][0][2] for result in results]
    lifts        =[result[2][0][3] for result in results]
    return list (zip(lhs, rhs, supports, confidences, lifts))
resultsinDataFrame = pd.DataFrame(inspect(results), columns = ["Left hand␣
 ↪side", "Right hand side", "Support", "Confidence", "Lift"])
```

****Finally, we store these variables into one dataframe, so that they are easier to visualize.****

[ ]:
```
resultsinDataFrame
```

[ ]:
```
   Left hand side       Right hand side   Support  Confidence      Lift
0        brownies        cottage cheese  0.003466    0.102767  3.225330
1         chicken           light cream  0.004533    0.075556  4.843951
2        escalope  mushroom cream sauce  0.005733    0.072269  3.790833
3        escalope                 pasta  0.005866    0.073950  4.700812
```

```
4     fresh bread         tomato juice  0.004266    0.099071  3.259356
5      fresh tuna                honey  0.003999    0.179641  3.785070
6   fromage blanc                honey  0.003333    0.245098  5.164271
7     ground beef        herb & pepper  0.015998    0.162822  3.291994
8     ground beef         tomato sauce  0.005333    0.054274  3.840659
9     light cream            olive oil  0.003200    0.205128  3.114710
10      olive oil    whole wheat pasta  0.007999    0.121457  4.122410
11          pasta               shrimp  0.005066    0.322034  4.506672
```

8 ****Now, we sort these final outputs in the descending order of lifts.****

```
[ ]: resultsinDataFrame.nlargest(n = 10, columns = "Lift")
```

```
[ ]:    Left hand side       Right hand side   Support  Confidence      Lift
     6   fromage blanc                 honey  0.003333    0.245098  5.164271
     1         chicken           light cream  0.004533    0.075556  4.843951
     3        escalope                 pasta  0.005866    0.073950  4.700812
     11          pasta                shrimp  0.005066    0.322034  4.506672
     10      olive oil     whole wheat pasta  0.007999    0.121457  4.122410
     8     ground beef          tomato sauce  0.005333    0.054274  3.840659
     2        escalope  mushroom cream sauce  0.005733    0.072269  3.790833
     5      fresh tuna                 honey  0.003999    0.179641  3.785070
     7     ground beef         herb & pepper  0.015998    0.162822  3.291994
     4     fresh bread          tomato juice  0.004266    0.099071  3.259356
```

****This is the final result of our apriori implementation in python. The SuperMarket will use this data to boost their sales and prioritize giving offers on the pair of items with greater Lift values [6].****