Project Report

Market Basket Analysis

Name:Keshav Anand

Course: AI and ML

(Batch-4)

Duration: 12 months

Problem Statement: Market Basket Analysis using Association Rule Mining.

Prerequisites

What things you need to install the software and how to install them:

Python 3.6 This setup requires that your machine has latest version of python. The following url https://www.python.org/downloads/ can be referred to download python. Once you have python downloaded and installed, you will need to setup PATH variables (if you want to run python program directly, detail instructions are below in how to run software section). To do that check this: https://www.pythoncentral.io/add-python-to-path-python-is-not- recognized-as-an-internal-or-external-command/. Setting up PATH variable is optional as you can also run program without it and more instruction are given below on this topic.

Second and easier option is to download anaconda and use its anaconda prompt to run the commands. To install anaconda check this url https://www.anaconda.com/download/ You will also need to download and install below 3 packages after you install either python or anaconda from the steps above Sklearn (scikit-learn) numpy scipy if you have chosen to install python 3.6 then run below commands in command prompt/terminal to install these packages pip install -U scikit-learn pip install numpy pip install scipy if you have chosen to install anaconda then run below commands in anaconda prompt to install these packages conda install -c scikit-learn conda install -c anaconda numpy conda install -c anaconda scipy

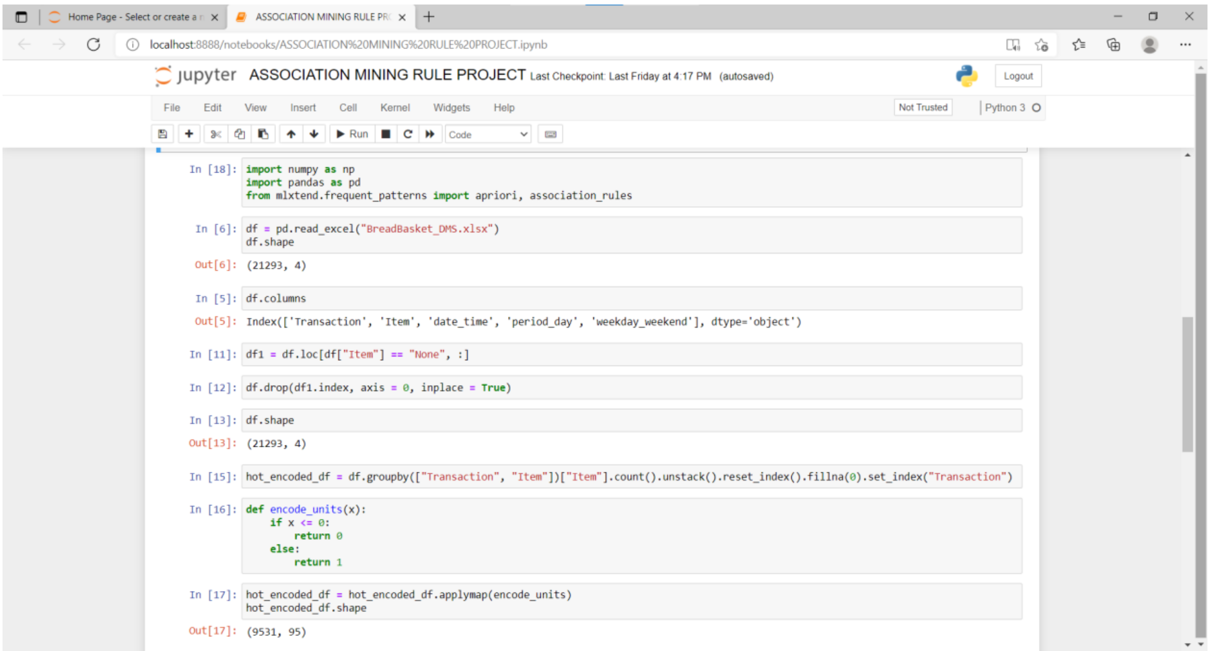
Dataset used:

The dataset used is the Bread-Basket Dataset available in website called Kaggle.com.

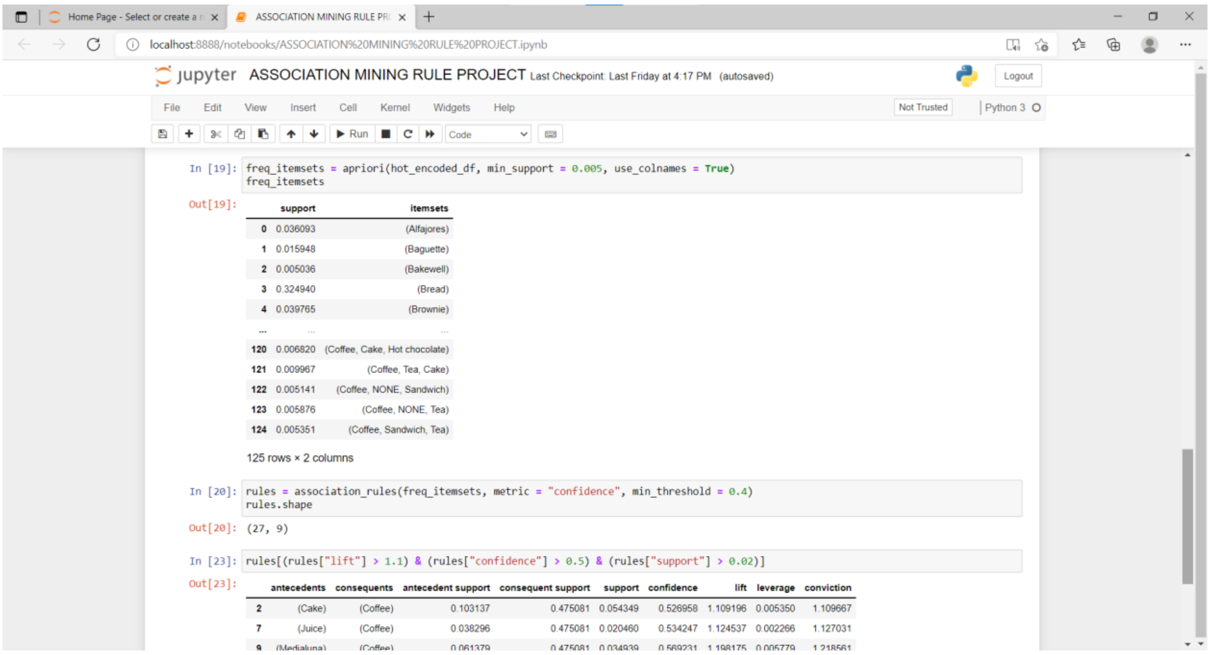
Method used for Analysis:

Association Rule Mining

Screenshots of Source Code and Output:


Home Page - Select or create a
ASSOCIATION MINING RULE PR
localhost:8888/notebooks/ASSOCIATION%20MINING%20RULE%20PROJECT.ipynb
jupyter ASSOCIATION MINING RULE PROJECT Last Checkpoint: Last Friday at 4:17 PM (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code

```
In [18]: import numpy as np
         import pandas as pd
         from mlxtend.frequent_patterns import apriori, association_rules

In [6]:  df = pd.read_excel("BreadBasket_DMS.xlsx")
         df.shape

Out[6]:  (21293, 4)

In [5]:  df.columns

Out[5]:  Index(['Transaction', 'Item', 'date_time', 'period_day', 'weekday_weekend'], dtype='object')

In [11]: df1 = df.loc[df["Item"] == "None", :]

In [12]: df.drop(df1.index, axis = 0, inplace = True)

In [13]: df.shape

Out[13]: (21293, 4)

In [15]: hot_encoded_df = df.groupby(["Transaction", "Item"])["Item"].count().unstack().reset_index().fillna(0).set_index("Transaction")

In [16]: def encode_units(x):
             if x <= 0:
                 return 0
             else:
                 return 1

In [17]: hot_encoded_df = hot_encoded_df.applymap(encode_units)
         hot_encoded_df.shape

Out[17]: (9531, 95)
```


Home Page - Select or create a
ASSOCIATION MINING RULE PR
localhost:8888/notebooks/ASSOCIATION%20MINING%20RULE%20PROJECT.ipynb
jupyter ASSOCIATION MINING RULE PROJECT Last Checkpoint: Last Friday at 4:17 PM (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code

```
In [19]: freq_itemsets = apriori(hot_encoded_df, min_support = 0.005, use_colnames = True)
         freq_itemsets
```

Out[19]:

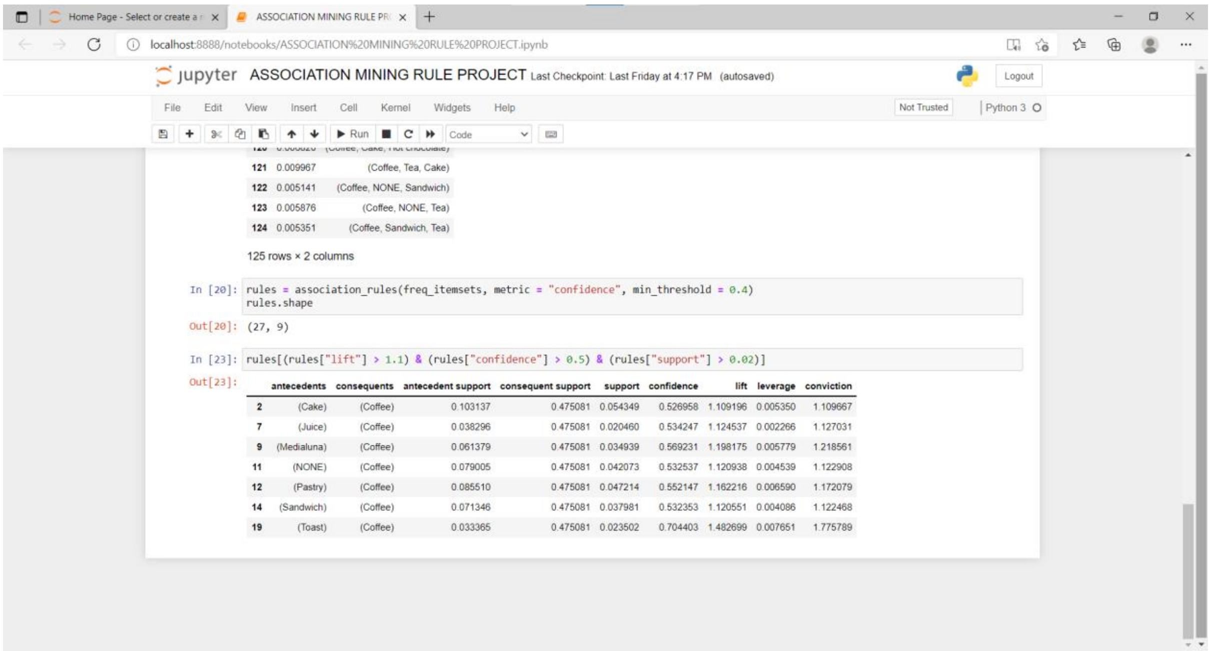
	support	itemsets
0	0.036093	(Alfajores)
1	0.015948	(Baguette)
2	0.005036	(Bakewell)
3	0.324940	(Bread)
4	0.039765	(Brownie)
...	...	...
120	0.006820	(Coffee, Cake, Hot chocolate)
121	0.009967	(Coffee, Tea, Cake)
122	0.005141	(Coffee, NONE, Sandwich)
123	0.005876	(Coffee, NONE, Tea)
124	0.005351	(Coffee, Sandwich, Tea)

125 rows × 2 columns

```
In [20]: rules = association_rules(freq_itemsets, metric = "confidence", min_threshold = 0.4)
         rules.shape

Out[20]: (27, 9)

In [23]: rules[(rules["lift"] > 1.1) & (rules["confidence"] > 0.5) & (rules["support"] > 0.02)]
```

Out[23]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
2	(Cake)	(Coffee)	0.103137	0.475081	0.054349	0.526958	1.109196	0.005350	1.109667
7	(Juice)	(Coffee)	0.038296	0.475081	0.020460	0.534247	1.124537	0.002266	1.127031
9	(Medialuna)	(Coffee)	0.061379	0.475081	0.034939	0.569231	1.198175	0.005779	1.218561


Home Page - Select or create a
ASSOCIATION MINING RULE PR
localhost:8888/notebooks/ASSOCIATION%20MINING%20RULE%20PROJECT.ipynb
jupyter
ASSOCIATION MINING RULE PROJECT Last Checkpoint: Last Friday at 4:17 PM (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
Python 3
Run
Code

121	0.009967	(Coffee, Tea, Cake)
122	0.005141	(Coffee, NONE, Sandwich)
123	0.005876	(Coffee, NONE, Tea)
124	0.005351	(Coffee, Sandwich, Tea)

125 rows × 2 columns

```
In [20]: rules = association_rules(freq_itemsets, metric = "confidence", min_threshold = 0.4)
         rules.shape

Out[20]: (27, 9)
```

```
In [23]: rules[(rules["lift"] > 1.1) & (rules["confidence"] > 0.5) & (rules["support"] > 0.02)]

Out[23]:
```

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
2	(Cake)	(Coffee)	0.103137	0.475081	0.054349	0.526958	1.109196	0.005350	1.109667
7	(Juice)	(Coffee)	0.038296	0.475081	0.020460	0.534247	1.124537	0.002266	1.127031
9	(Medialuna)	(Coffee)	0.061379	0.475081	0.034939	0.569231	1.198175	0.005779	1.218561
11	(NONE)	(Coffee)	0.079005	0.475081	0.042073	0.532537	1.120938	0.004539	1.122908
12	(Pastry)	(Coffee)	0.085510	0.475081	0.047214	0.552147	1.162216	0.006590	1.172079
14	(Sandwich)	(Coffee)	0.071346	0.475081	0.037981	0.532353	1.120551	0.004086	1.122468
19	(Toast)	(Coffee)	0.033365	0.475081	0.023502	0.704403	1.482699	0.007651	1.775789