**Lab #4: Association Rule Mining Using Python**

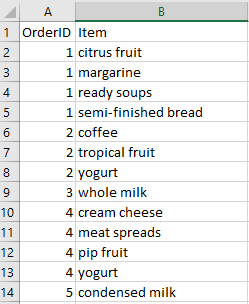
**What to submit:** a single word/pdf file with answers for the questions in **Part 5**.

# Before You Start

You’ll need two files to do this exercise: **Lab 4.py** (the Python script file) and **Groceries.csv** (the data file), both available on the Blackboard. (*Download both files and save them to the Working Directory folder. Also make sure you are connected to the Internet when you do this exercise!*)

# Part 1: Look at the Data File

1. The input file for an Association Rule analysis follows this general format. The first value is the “basket” and the second value is the “event.” If multiple events occur within a basket, then the basket number is repeated for each event. Both the basket and event are discrete values representing a particular basket and a particular event.
2. Now open the Groceries.csv data file. If it warns you, that’s ok. Just click “Yes” or “OK.”
3. You’ll see something like this:



The data file contains 43,367 rows of order data covering 9,835 orders and the items in each order.

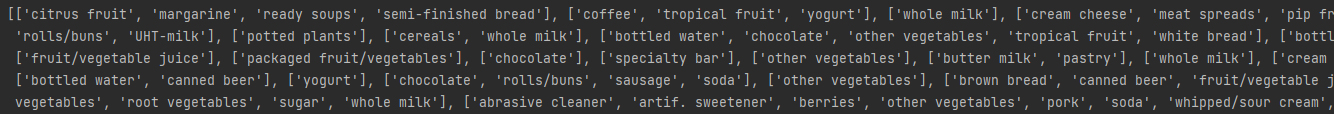
Look at the contents of the file. Each row represents an order/product pair. The first value is the order number (OrderID) and the second value is the product/item name. We can see from the excerpt that Order 1 contains the item citrus fruit, margarine, ready soups and semi-finished bread.

We will use this dataset to predict which items customers are likely to buy together.

1. Close the Groceries.csv file. If it asks you to save the file, choose “Don’t Save”.

# Part 2: Explore the Lab 4.py Script

1. Open the Lab 4.py file. This contains the Python script that performs the association mining analysis.
2. Look at lines 2 and 7. These install (when needed) the packages, which are needed for computing the association rules.
3. Look at line 13 to 19. These help input and transform the data you will use for the analysis. Note that, at this moment your data is like below:



1. Look at line 30 to 32. These code transform transaction data in a tabular format into a one-hot encoded NumPy array suitable for use with the apriori algorithm

Now, you get the below new data:

Table

Description automatically generated

1. Now let’s look at the apriori() function that computes the association rules. Scroll down to line 19:

frequent\_itemsets = apriori(df, min\_support=0.01, use\_colnames=True)

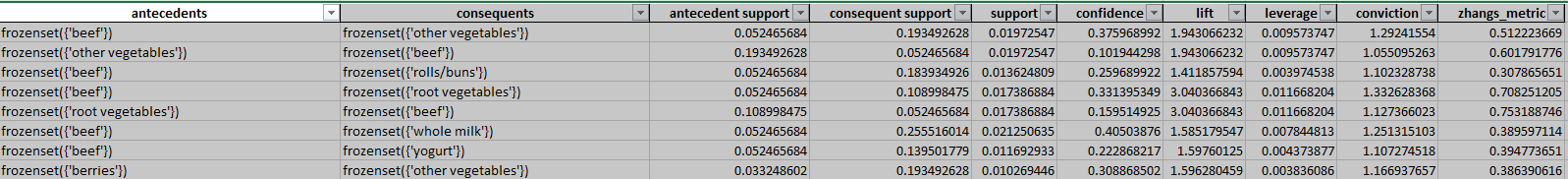
You can see a few things at work:

* The apriori() function is used to compute the association rules (and the results are stored in frequent\_itemsets).
* df is the transaction data read from the data file.
* min\_support defines the minimum threshold for support; any association below this threshold won't appear in the output.

1. Now let’s look at the association\_rules( ) function which generates association rules from frequent itemsets. Here, we set another threshold for confidence; any association below this threshold won't appear in the output.

rules = association\_rules(frequent\_itemsets, metric="confidence", min\_threshold=0.1)

# Part 3: Execute the Lab4.py Script

1. Then select Run/Run/Lab4.py.
2. You’ll see find a file association\_rules.xlsx generated in your working directory.
3. Close the Lab4.py script. If it asks you to save the file, click “Save.”
4. Open association\_rules.xlsx. You’ll see your rules in a spreadsheet:  
     
   

Column A lists the “if” part, and Column B presents “then” part. Columns C through G contain the support, confidence and lift values of each association rule.

1. Click on the icon next to the lift column header and select “Sort Largest to Smallest.”
2. You’ll now see the list sorted by lift values:  
     
   Graphical user interface

   Description automatically generated with medium confidence
3. Save your workbook.
4. Now let’s look at rule 1 (in row 2):



We see that this is the rule with the highest lift {whole milk, yogurt} => {curd}. It doesn’t occur that often (support = 0.010066), but the relationship is fairly strong (confidence = 0.179673) and the rule has high predictive power (lift = 3.372647>1). The high lift indicates that if a person bought whole milk, yogurt in an order, it is much more likely than chance (lift>1) that she/he will also get a curd.

1. Now scroll down to the end of the table:



We see that this is the rule with a low lift { canned beer } => { rolls/buns }. It doesn’t occur that often (support = 0.011286), and the relationship is fairly strong (confidence = 0.14528). The low lift (0.789887) indicates that if a person bought canned beer in an order, it is less likely than chance (lift<1) that she/he will also get a rolls or buns.

# Part 5: Try it yourself

Looking at your Excel worksheet with the imported rules and answer the following questions:  
  
  
a) How many rules are there with a lift value between 0.95 and 1.05 (excluding those with a lift of exactly 1)? Explain in business terms what the lift value means to you?

There are 14 rules with a lift value between 0.95 and 1.05. This indicates that these item pairs are not strongly associated with each other and have low predictive power. Lift values are often used in market basket analysis to identify relationships between products that are frequently purchased together.

b) What products are customers with a soda most likely to also have (or be interested in having)? Explain your answer. (confidence)  
  
 **(HINT: Sort the rules in alphabetical order to make those rules easier to find.)**

According to confidence, the answer will be whole milk

Soda and whole milk have the highest confidence of 0.229, thus the probability that a customer with soda will also buy whole milk is 22.9%

c) Compare rule 302 {canned beer} => {shopping bags} and rule 461: {canned beer} => {roll/buns}. Explain in business terms what it means and what you, as a supermarket manager, should do with that information.

Support and confidence are similar concepts, but they differ from lift. Rule 461 has a lift value less than 1, while rule 302 has a lift value greater than 1 This means that the customers who purchase canned beer are less likely to buy roll/buns but more likely to buy shopping bags compared to random chance. There is no single answer.

If a manager wants to boost the sale of milk, she could target customers buying yogurt and curd with a campaign promoting milk