PERFORMANCE ASSESSMENT_ D212

OFM3 TASK 3: Lift Analysis and Association Rules

March 7th,2022

Table of Contents

Performance Evaluation of OFM3: LIFT ANALYSIS, D212

Name: Rekha Alle
Student ID: 000778673
Course: Masters Data Analytics
Date: 03/07/2022
Program Mentor: Kesselly kamara
Contact: 385-428-4395
Email: Kesselly.kamara@wgu.edu

PART I. Introduction to Scenario

Understanding customers is one of the most important aspects of customer relationship management that directly influences a company's long-term success. When a corporation has a greater understanding of its consumers' traits, this could good target promotion and advertising campaigns for them, resulting in higher long-term earnings.

You operate as an investigator for a telecoms business that wants to understand more about its customers' characteristics. You've been given with conducting market basket research on customer research to discover critical relationships between consumer purchases, enabling for better operational and organizational selection.

A1. Analytical Question:

Which items of interest are likely to reduce client churn when combined with discounts? That is, could we learn about which things will endear us to clients if we provide them as a discount with our services by researching a list of transactions?

The **market basket analysis** method will be used to solve this analysis.

A2. Goals and Objectives:

Everybody in the organization will profit from recognizing, with some degree of certainty, which customers will be able to churn, since it will give importance to selling enhanced services to consumers with these traits and previous user experiences. This data analysis' purpose is to give numerical information to company stakeholders to assist them better understand their customers.

PART II. Justification for the method

B1. Assumptions Summary: Market Basket

Li remarked, "One of the most important tools for discovering correlations between commodities is market basket analysis. It works by looking for "often occurring item combinations" in transactions." " (Li, p. 1).

The goal of this study is to figure out which telecom peripherals and ICT tools people prefer and buy together the most. We'll try to figure out which things are bought together the most frequently and show the connections between them.

We anticipate finding the best combination of things to provide at a discount in conjunction with our services.

Market based analysis analyzes the selected dataset using Apriori algorithm as follows:

- Load the "teleco market basket.csv" file into a dataframe.
- To use the Apriori algorithm, convert the loaded dataset into a list.
- "teleco market basket prepared.csv" was used to extract and prepare the dataset.
- Apriori algorithm to generate association rules
- Individual dataframes should have their own support.
- And to get the association rule table, we need to loop thru the results and populate four list variables, convert them to dataframe and concatenate into single dataframe.
- And set column names like lhs, 1, 2, rhs, 1, 2, support, confidence, lift
- Check the rules for confidence, support, and lift that have the highest values.
- Recommend a course of action based on the findings of our investigation.

B2. Example of a Transaction:

Transactions are easily distinct from one another when looking at the dataset. A broader list of twenty items is included in the first transaction, as are included

- Logitech M510 Wireless mouse
- HP 63 Ink
- HP 65 ink
- nonda USB C to USB Adapter
- 10ft iPHone Charger Cable
- HP 902XL ink
- Creative Pebble 2.0 Speakers
- Cleaning Gel Universal Dust Cleaner
- Micro Center 32GB Memory card
- YUNSONG 3pack 6ft Nylon Lightning Cable
- TopMate C5 Laptop Cooler pad
- Apple USB-C Charger cable
- HyperX Cloud Stinger Headset
- TONOR USB Gaming Microphone
- Dust-Off Compressed Gas 2 pack
- 3A USB Type C Cable 3 pack 6FT
- HOVAMP iPhone charger
- SanDisk Ultra 128GB card
- FEEL2NICE 5 pack 10ft Lighning cable
- FEIYOLD Blue light Blocking Glasses

1 shopper bought all 20 things at the same time.

B3. Assumption of Market Basket:

Making decisions based on building association rules is one of Market Basket Analysis assumptions. Dr. Susan Sivek recommends following these principles "are just statements that link an 'antecedent' and a ‘consequent' item.

Association rules do not imply causal ties, merely that they occur together " (Sivek, p. 1).

As an example, in our study project, we would like to discover products that would be acquired prior to subscribing to a telecom service, or items that might be utilized in conjunction with telecom services.

Looking at the list associated items, those are all substitute items the user was looking for. As complements are commodities that are consumed in conjunction with one another. Substitutes are items that can be consumed instead of the original. The demand curve is also shifted by the prices of complementary or replacement items. Not only did our market basket analysis (MBA) of this transaction dataset reveal little significance, but none of the pairings showed that customers who utilized telecom services would want or need a related item.

PART III. Data Objectives:

C1. The following will be data transformation:

1. **Include standard imports all the required references:**

```
In [1]: # Standard data science imports
        import numpy as np
        import pandas as pd
        # Visualization libraries
        import seaborn as sns
        import matplotlib.pyplot as plt
        %matplotlib inline
```

2. **Change font and color of the Matplotlib:**

```
In [2]: # Change color of Matplotlib font
        import matplotlib as mpl

        COLOR = 'white'
        mpl.rcParams['text.color'] = COLOR
        mpl.rcParams['axes.labelcolor'] = COLOR
        mpl.rcParams['xtick.color'] = COLOR
        mpl.rcParams['ytick.color'] = COLOR
```

3. **Increase display cell-width**

```
In [3]: # Increase Jupyter display cell-width
        from IPython.core.display import display, HTML
        display(HTML("<style>.container { width:75% !important; }</style>"))
```

4. **Ignore warning codes**

```
In [4]: # Ignore Warning Code
        import warnings
        warnings.filterwarnings('ignore')
```

5. **Dataset**

```
In [5]: # Load data set into Pandas dataframe
        teleco = pd.read_csv('C:/Kailash/Rekha/D212/data/teleco_market_basket.csv')
```

6. Dataset

In [6]:
```
# Examine the features of the dataset
teleco.columns
```

Out[6]:
```
Index(['Item01', 'Item02', 'Item03', 'Item04', 'Item05', 'Item06', 'Item07',
       'Item08', 'Item09', 'Item10', 'Item11', 'Item12', 'Item13', 'Item14',
       'Item15', 'Item16', 'Item17', 'Item18', 'Item19', 'Item20'],
      dtype='object')
```

7. Data set size

In [7]:
```
# Get an idea of dataset size
teleco.shape
```

Out[7]:
```
(15002, 20)
```

8. Data frame Info

In [8]:
```
# Examine first few records of dataset
teleco.head()
```

Out[8]:

	Item01	Item02	Item03	Item04	Item05	Item06	Item07	Item08	Item09	Item10	Item11	Item12	Item13	Item14	Item15	Item16	Item17	Item18	Item19	Item20
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	Logitech M510 Wireless mouse	HP 63 Ink	HP 65 ink	nonda USB C to USB Adapter	10ft iPHone Charger Cable	HP 902XL ink	Creative Pebble 2.0 Speakers	Cleaning Gel Universal Dust Cleaner	Micro Center 32GB Memory card	YUNSONG 3pack 6ft Nylon Lightning Cable	TopMate C5 Laptop Cooler pad	Apple USB-C Charger cable	HyperX Cloud Stinger Headset	TONOR USB Gaming Microphone	Dust-Off Compressed Gas 2 pack	3A USB Type C Cable 3 pack 6FT	HOVAMP iPhone charger	SanDisk Ultra 128GB card	FEEL2NICE 5 pack 10ft Lighning cable	FEIYOLD Blue light Blocking Glasses
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
3	Apple Lightning to Digital AV Adapter	TP-Link AC1750 Smart WiFi Router	Apple Pencil	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

In [9]:
```
# View DataFrame info
# teleco.info
```

9. Data types

In [11]:
```
# Get data types of features
teleco.dtypes
```

Out[11]:
```
Item01    object
Item02    object
Item03    object
Item04    object
Item05    object
Item06    object
Item07    object
Item08    object
Item09    object
Item10    object
Item11    object
Item12    object
Item13    object
Item14    object
Item15    object
Item16    object
Item17    object
Item18    object
Item19    object
Item20    object
dtype: object
```

10. Data set

```
In [10]: # Get an overview of descriptive statistics
teleco.describe()
```

Out[10]:

	Item01	Item02	Item03	Item04	Item05	Item06	Item07	Item08	Item09	Item10
count	7501	5747	4389	3345	2529	1864	1369	981	654	395
unique	115	117	115	114	110	106	102	97	88	80
top	Dust-Off Compressed Gas 2 pack	Dust-Off Compressed Gas 2 pack	Dust-Off Compressed Gas 2 pack	Dust-Off Compressed Gas 2 pack	Apple USB-C Charger cable	USB 2.0 Printer cable	Apple USB-C Charger cable	Apple USB-C Charger cable	Apple USB-C Charger cable	Apple USB-C Charger cable
freq	577	484	375	201	153	107	96	67	57	31

	Item11	Item12	Item13	Item14	Item15	Item16	Item17	Item18	Item19	Item2(
count	256	154	87	47	25	8	4	4	3	1
unique	66	50	43	28	19	8	3	3	3	1
top	TopMate C5 Laptop Cooler pad	Apple USB-C Charger cable	Apple USB-C Charger cable	Apple USB-C Charger cable	ARRIS SURFboard SB8200 Cable Modem	HP 61 ink	SanDisk Ultra 128GB card	Brother Genuine High Yield Toner Cartridge	iPhone Charger Cable Anker 6ft	FEIYOLD Blue light Blocking Glasses
freq	22	15	8	4	3	1	2	2	1	1

11. Missing Data Points

```
In [12]: # Discover missing data points within dataset
data_nulls = teleco.isnull().sum()
print(data_nulls)
```

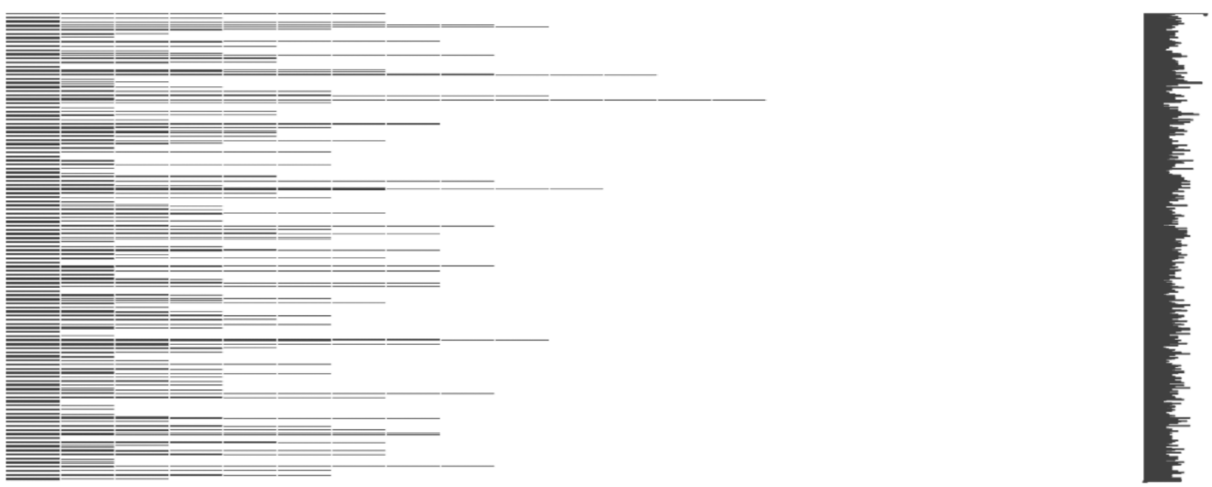
```
Item01     7501
Item02     9255
Item03    10613
Item04    11657
Item05    12473
Item06    13138
Item07    13633
Item08    14021
Item09    14348
Item10    14607
Item11    14746
Item12    14848
Item13    14915
Item14    14955
Item15    14977
Item16    14994
Item17    14998
Item18    14998
Item19    14999
Item20    15001
dtype: int64
```

C1: Imagination/Visualization

```
In [13]: # Check for missing data & visualize missing values in dataset
         # Install appropriate library
         !pip install missingno
         # Importing the libraries
         import missingno as msno
         # Visualize missing values as a matrix
         msno.matrix(teleco);
         """(GeeksForGeeks, p. 1)"""
```

```
Requirement already satisfied: missingno in c:\users\kaila\anaconda3\lib\site-packages (0.5.0)
Requirement already satisfied: seaborn in c:\users\kaila\anaconda3\lib\site-packages (from missingno) (0.11.1)
Requirement already satisfied: scipy in c:\users\kaila\anaconda3\lib\site-packages (from missingno) (1.6.2)
Requirement already satisfied: matplotlib in c:\users\kaila\anaconda3\lib\site-packages (from missingno) (3.3.4)
Requirement already satisfied: numpy in c:\users\kaila\anaconda3\lib\site-packages (from missingno) (1.20.1)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\kaila\anaconda3\lib\site-packages (from matplotlib->missingno) (1.3.1)
Requirement already satisfied: pillow>=6.2.0 in c:\users\kaila\anaconda3\lib\site-packages (from matplotlib->missingno) (8.2.0)
Requirement already satisfied: python-dateutil>=2.1 in c:\users\kaila\anaconda3\lib\site-packages (from matplotlib->missingno) (2.8.1)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.3 in c:\users\kaila\anaconda3\lib\site-packages (from matplotlib->missingno) (2.4.7)
Requirement already satisfied: cycler>=0.10 in c:\users\kaila\anaconda3\lib\site-packages (from matplotlib->missingno) (0.10.0)
Requirement already satisfied: six in c:\users\kaila\anaconda3\lib\site-packages (from cycler>=0.10->matplotlib->missingno) (1.15.0)
Requirement already satisfied: pandas>=0.23 in c:\users\kaila\anaconda3\lib\site-packages (from seaborn->missingno) (1.2.4)
Requirement already satisfied: pytz>=2017.3 in c:\users\kaila\anaconda3\lib\site-packages (from pandas>=0.23->seaborn->missingno) (2021.1)
```

```
Out[13]: '(GeeksForGeeks, p. 1)'
```



In [14]:
```
# Drop records with missing values
teleco.dropna(how='all', inplace=True)
# Review changes
teleco.head()
```

Out[14]:

	Item01	Item02	Item03	Item04	Item05	Item06	Item07	Item08	Item09	Item10	Item11	Item12	Item13	Item14	Item15	Item16	Item17	Item18	Item19	Item20
1	Logitech M510 Wireless mouse	HP 63 Ink	HP 65 ink	nonda USB C to USB Adapter	10ft iPHone Charger Cable	HP 902XL ink	Creative Pebble 2.0 Speakers	Cleaning Gel Universal Dust Cleaner	Micro Center 32GB Memory card	YUNSONG 3pack 6ft Nylon Lightning Cable	TopMate C5 Laptop Cooler pad	Apple USB-C Charger cable	HyperX Cloud Stinger Headset	TONOR USB Gaming Microphone	Dust-Off Compressed Gas 2 pack	3A USB Type C Cable 3 pack 6FT	HOVAMP iPhone charger	SanDisk Ultra 128GB card	FEEL2NICE 5 pack 10ft Lighning cable	FEIYOLD Blue light Blocking Glasses
3	Apple Lightning to Digital AV Adapter	TP-Link AC1750 Smart WiFi Router	Apple Pencil	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
5	UNEN Mfi Certified 5-pack Lightning Cable	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
7	Cat8 Ethernet Cable	HP 65 ink	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
9	Dust-Off Compressed Gas 2 pack	Screen Mom Screen Cleaner kit	Moread HDMI to VGA Adapter	HP 62XL Tri-Color ink	Apple USB-C Charger cable	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

In [15]:
```
# Replace empty values with 0
teleco.fillna(0, inplace=True)
```

In [16]:
```
# Get an idea of dataset size after changes
teleco.shape
```

Out[16]: (7501, 20)

In [17]:
```
# Review changes to DataFrame
teleco.head()
```

Out[17]:

	Item01	Item02	Item03	Item04	Item05	Item06	Item07	Item08	Item09	Item10	Item11	Item12	Item13	Item14	Item15	Item16	Item17	Item18	Item19	Item20
1	Logitech M510 Wireless mouse	HP 63 Ink	HP 65 ink	nonda USB C to USB Adapter	10ft iPHone Charger Cable	HP 902XL ink	Creative Pebble 2.0 Speakers	Cleaning Gel Universal Dust Cleaner	Micro Center 32GB Memory card	YUNSONG 3pack 6ft Nylon Lightning Cable	TopMate C5 Laptop Cooler pad	Apple USB-C Charger cable	HyperX Cloud Stinger Headset	TONOR USB Gaming Microphone	Dust-Off Compressed Gas 2 pack	3A USB Type C Cable 3 pack 6FT	HOVAMP iPhone charger	SanDisk Ultra 128GB card	FEEL2NICE 5 pack 10ft Lighning cable	FEIYOLD Blue light Blocking Glasses
3	Apple Lightning to Digital AV Adapter	TP-Link AC1750 Smart WiFi Router	Apple Pencil	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	UNEN Mfi Certified 5-pack Lightning Cable	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	Cat8 Ethernet Cable	HP 65 ink	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	Dust-Off Compressed Gas 2 pack	Screen Mom Screen Cleaner kit	Moread HDMI to VGA Adapter	HP 62XL Tri-Color ink	Apple USB-C Charger cable	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

In [18]:
```python
# Confirm no null values
teleco.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 7501 entries, 1 to 15001
Data columns (total 20 columns):
 #   Column  Non-Null Count  Dtype 
---  ------  --------------  ----- 
 0   Item01  7501 non-null   object
 1   Item02  7501 non-null   object
 2   Item03  7501 non-null   object
 3   Item04  7501 non-null   object
 4   Item05  7501 non-null   object
 5   Item06  7501 non-null   object
 6   Item07  7501 non-null   object
 7   Item08  7501 non-null   object
 8   Item09  7501 non-null   object
 9   Item10  7501 non-null   object
 10  Item11  7501 non-null   object
 11  Item12  7501 non-null   object
 12  Item13  7501 non-null   object
 13  Item14  7501 non-null   object
 14  Item15  7501 non-null   object
 15  Item16  7501 non-null   object
 16  Item17  7501 non-null   object
 17  Item18  7501 non-null   object
 18  Item19  7501 non-null   object
 19  Item20  7501 non-null   object
dtypes: object(20)
memory usage: 1.2+ MB
```

In [20]:
```python
# Convert dataset into list format for use with Apriori algorithm
teleco_list = []
for i in range(0, 7501):
    teleco_list.append([str(teleco.values[i, j]) for j in range(0, 20)])
teleco_cleaned = pd.DataFrame(teleco_list)
```

In [21]:
```python
# Review DataFrame
teleco_cleaned.head()
```

Out[21]:

	0	1	2	3	4	5	6	7	8	9
0	Logitech M510 Wireless mouse	HP 63 Ink	HP 65 ink	nonda USB C to USB Adapter	10ft iPHone Charger Cable	HP 902XL ink	Creative Pebble 2.0 Speakers	Cleaning Gel Universal Dust Cleaner	Micro Center 32GB Memory card	YUNSONG 3pack 6ft Nylon Lightning Cable
1	Apple Lightning to Digital AV Adapter	TP-Link AC1750 Smart WiFi Router	Apple Pencil	0	0	0	0	0	0	0
2	UNEN Mfi Certified 5-pack Lightning Cable	0	0	0	0	0	0	0	0	0
3	Cat8 Ethernet Cable	HP 65 ink	0	0	0	0	0	0	0	0
4	Dust-Off Compressed Gas 2 pack	Screen Mom Screen Cleaner kit	Moread HDMI to VGA Adapter	HP 62XL Tri-Color ink	Apple USB-C Charger cable	0	0	0	0	0

	10	11	12	13	14	15	16	17	18	19
0	TopMate C5 Laptop Cooler pad	Apple USB-C Charger cable	HyperX Cloud Stinger Headset	TONOR USB Gaming Microphone	Dust-Off Compressed Gas 2 pack	3A USB Type C Cable 3 pack 6FT	HOVAMP iPhone charger	SanDisk Ultra 128GB card	FEEL2NICE 5 pack 10ft Lighning cable	FEIYOLD Blue light Blocking Glasses
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0

In [22]:
```python
# Extract prepared dataset
teleco_cleaned.to_csv('C:/Kailash/Rekha/D212/data/teleco_market_basket_prepared.csv')
```

In [22]:
```
# Extract prepared dataset
teleco_cleaned.to_csv('C:/Kailash/Rekha/D212/data/teleco_market_basket_prepared.csv')
```

In [23]:
```
teleco_list[:1]
```

Out[23]:
```
[['Logitech M510 Wireless mouse',
  'HP 63 Ink',
  'HP 65 ink',
  'nonda USB C to USB Adapter',
  '10ft iPHone Charger Cable',
  'HP 902XL ink',
  'Creative Pebble 2.0 Speakers',
  'Cleaning Gel Universal Dust Cleaner',
  'Micro Center 32GB Memory card',
  'YUNSONG 3pack 6ft Nylon Lightning Cable',
  'TopMate C5 Laptop Cooler pad',
  'Apple USB-C Charger cable',
  'HyperX Cloud Stinger Headset',
  'TONOR USB Gaming Microphone',
  'Dust-Off Compressed Gas 2 pack',
  '3A USB Type C Cable 3 pack 6FT',
  'HOVAMP iPhone charger',
  'SanDisk Ultra 128GB card',
  'FEEL2NICE 5 pack 10ft Lighning cable',
  'FEIYOLD Blue light Blocking Glasses']]
```

C2: Execution of Code:

In [34]:
```
# Generate association rules from Apriori algorithm
!pip install apyori
from apyori import apriori

# Train Apriori algorithm on the dataset
rule_list = apriori(teleco_list, min_support = 0.003, min_confidence = 0.3, min_lift = 3, min_length = 2)
```

```
Collecting apyori
  Downloading apyori-1.1.2.tar.gz (8.6 kB)
Building wheels for collected packages: apyori
  Building wheel for apyori (setup.py): started
  Building wheel for apyori (setup.py): finished with status 'done'
  Created wheel for apyori: filename=apyori-1.1.2-py3-none-any.whl size=5975 sha256=dada2363d4b8e36f80f955a2d78afb37a5b43a4d751364cc5c944e7072bfc0c5
  Stored in directory: c:\users\kaila\appdata\local\pip\cache\wheels\1b\02\6c\a45230be8603bd95c0a51cd2b289aefdd860c1a100eab73661
Successfully built apyori
Installing collected packages: apyori
Successfully installed apyori-1.1.2
```

In [35]:
```
# Review generate rules
rule_list = list(rule_list)
print(rule_list[0])
```

```
RelationRecord(items=frozenset({'5pack Nylon Braided USB C cables', 'HP 63XL Ink'}), support=0.005732568990801226, ordered_statistics=[OrderedStatistic(items_base=fr
ozenset({'5pack Nylon Braided USB C cables'}), items_add=frozenset({'HP 63XL Ink'}), confidence=0.3006993006993007, lift=3.790832696715049)])
```

In [36]:
```
# Print number of rules
print(len(rule_list))
```

```
102
```

In [37]:
```
# Transform results into DataFrame structure
results = pd.DataFrame(rule_list)
```

Out[38]:

	items	support	ordered_statistics
0	(5pack Nylon Braided USB C cables, HP 63XL Ink)	0.005733	[((5pack Nylon Braided USB C cables), (HP 63XL...
1	(AutoFocus 1080p Webcam, SanDisk Ultra 64GB card)	0.005333	[((AutoFocus 1080p Webcam), (SanDisk Ultra 64G...
2	(iPhone 11 case, HP 63XL Ink)	0.005866	[((iPhone 11 case), (HP 63XL Ink), 0.372881355...
3	(iPhone 11 case, Logitech M510 Wireless mouse)	0.005066	[((iPhone 11 case), (Logitech M510 Wireless mo...
4	(SanDisk 128GB Ultra microSDXC card, SanDisk U...	0.015998	[((SanDisk 128GB Ultra microSDXC card), (SanDi...
...	...	...	...
97	(Dust-Off Compressed Gas 2 pack, VIVO Dual LCD...	0.004399	[((Nylon Braided Lightning to USB cable, Dust-...
98	(Screen Mom Screen Cleaner kit, SanDisk Ultra ...	0.003200	[((SanDisk Ultra 128GB card, Dust-Off Compress...
99	(VIVO Dual LCD Monitor Desk mount, Nylon Braid...	0.003066	[((Nylon Braided Lightning to USB cable, HP 61...
100	(Screen Mom Screen Cleaner kit, VIVO Dual LCD ...	0.003466	[((Nylon Braided Lightning to USB cable, HP 61...
101	(Screen Mom Screen Cleaner kit, VIVO Dual LCD ...	0.003066	[((Screen Mom Screen Cleaner kit, Nylon Braide...

102 rows × 3 columns

In [39]:
```
# Separate support to indiviual DataFrame
support = results.support
```

In [40]:
```
# Instantiate four empty lists to contain lhs, rhs, confidence and lift
first_values = []
second_values = []
third_values = []
fourth_values = []
```

In [42]:
```python
# Create for loop to iterate over list
for i in range(results.shape[0]):
    single_list = results['ordered_statistics'][i][0]
    first_values.append(list(single_list[0]))
    second_values.append(list(single_list[1]))
    third_values.append(single_list[2])
    fourth_values.append(single_list[3])
```

In [43]:
```python
# Convert lists into DataFrame
lhs = pd.DataFrame(first_values)
rhs = pd.DataFrame(second_values)
confidence = pd.DataFrame(third_values, columns=['confidence'])
lift = pd.DataFrame(fourth_values, columns=['lift'])
```

In [44]:
```python
# Concatenate lists into single DataFrame
results_final = pd.concat([lhs, rhs, support, confidence, lift], axis=1)
results_final.fillna(value=' ', inplace=True)
```

In [45]:
```python
# View final results
results_final
```

Out[45]:

	0	1	2	0	1	2	support	confidence	lift
0	5pack Nylon Braided USB C cables			HP 63XL Ink			0.005733	0.300699	3.790833
1	AutoFocus 1080p Webcam			SanDisk Ultra 64GB card			0.005333	0.377358	3.840659
2	iPhone 11 case			HP 63XL Ink			0.005866	0.372881	4.700812
3	iPhone 11 case			Logitech M510 Wireless mouse			0.005066	0.322034	4.506672
4	SanDisk 128GB Ultra microSDXC card			SanDisk Ultra 64GB card			0.015998	0.323450	3.291994
...	...	...	...	...	...	...	...	...	...
97	Nylon Braided Lightning to USB cable	Dust-Off Compressed Gas 2 pack	VIVO Dual LCD Monitor Desk mount	0	SanDisk Ultra 64GB card		0.004399	0.366667	3.731841
98	SanDisk Ultra 128GB card	Dust-Off Compressed Gas 2 pack	VIVO Dual LCD Monitor Desk mount	Screen Mom Screen Cleaner kit	0		0.003200	0.470588	3.631566
99	Nylon Braided Lightning to USB cable	HP 61 ink	SanDisk Ultra 64GB card	0	VIVO Dual LCD Monitor Desk mount		0.003066	0.534884	3.072100
100	Nylon Braided Lightning to USB cable	HP 61 ink	VIVO Dual LCD Monitor Desk mount	Screen Mom Screen Cleaner kit	0		0.003466	0.440678	3.400746
101	Screen Mom Screen Cleaner kit	Nylon Braided Lightning to USB cable	SanDisk Ultra 64GB card	0	VIVO Dual LCD Monitor Desk mount		0.003066	0.534884	3.072100

102 rows × 9 columns

C3: Table of Rules for an Association:

In [46]:
```
# Set column names
results_final.columns = ['lhs', 1, 2, 'rhs', 1, 2, 'support', 'confidence', 'lift']
results_final_1 = results_final[['lhs', 'rhs', 'support', 'confidence', 'lift']]
results_final_1
```

Out[46]:

	lhs	rhs	support	confidence	lift
0	5pack Nylon Braided USB C cables	HP 63XL Ink	0.005733	0.300699	3.790833
1	AutoFocus 1080p Webcam	SanDisk Ultra 64GB card	0.005333	0.377358	3.840659
2	iPhone 11 case	HP 63XL Ink	0.005866	0.372881	4.700812
3	iPhone 11 case	Logitech M510 Wireless mouse	0.005066	0.322034	4.506672
4	SanDisk 128GB Ultra microSDXC card	SanDisk Ultra 64GB card	0.015998	0.323450	3.291994
...	...	...	...	...	...
97	Nylon Braided Lightning to USB cable	0	0.004399	0.366667	3.731841
98	SanDisk Ultra 128GB card	Screen Mom Screen Cleaner kit	0.003200	0.470588	3.631566
99	Nylon Braided Lightning to USB cable	0	0.003066	0.534884	3.072100
100	Nylon Braided Lightning to USB cable	Screen Mom Screen Cleaner kit	0.003466	0.440678	3.400746
101	Screen Mom Screen Cleaner kit	0	0.003066	0.534884	3.072100

102 rows × 5 columns

Highest combination of Support, Confidence and LIft

After running the final results to create the association rules table, we can demonstate mathematically that "5pack Nylon Braided USB C cables" and "HP 63XL Ink" have the highest combination of values for our three metrics:

For "5pack Nylon Braided USB C cables" – "HP 63XL Ink"

- Support = 0.0057
- Confidence = 0.3007
- Lift = 3.7908

In [48]:
```
# Visualize the list of rules
results = list(rule_list)
for i in results:
    print('\n')
    print(i)
    print('**********')
```

```
RelationRecord(items=frozenset({'SanDisk 128GB Ultra microSDXC card', '0', 'SanDisk Ultra 64GB card'}), support=0.015997866951073192, ordered_statistics=[OrderedS
tatistic(items_base=frozenset({'SanDisk 128GB Ultra microSDXC card'}), items_add=frozenset({'0', 'SanDisk Ultra 64GB card'}), confidence=0.3234501347708895, lift=
3.2919938411349285), OrderedStatistic(items_base=frozenset({'SanDisk 128GB Ultra microSDXC card', '0'}), items_add=frozenset({'SanDisk Ultra 64GB card'}), confide
nce=0.3234501347708895, lift=3.2919938411349285)])
**********


RelationRecord(items=frozenset({'Screen Mom Screen Cleaner kit', 'Anker USB C to HDMI Adapter', '10ft iPhone Charger Cable 2 Pack'}), support=0.003066257832289028
2, ordered_statistics=[OrderedStatistic(items_base=frozenset({'Anker USB C to HDMI Adapter', '10ft iPhone Charger Cable 2 Pack'}), items_add=frozenset({'Screen Mo
m Screen Cleaner kit'}), confidence=0.44230769230769235, lift=3.4133230452674903)])
**********


RelationRecord(items=frozenset({'Screen Mom Screen Cleaner kit', '10ft iPhone Charger Cable 2 Pack', 'FEIYOLD Blue light Blocking Glasses'}), support=0.0035995200
639914677, ordered_statistics=[OrderedStatistic(items_base=frozenset({'10ft iPhone Charger Cable 2 Pack', 'FEIYOLD Blue light Blocking Glasses'}), items_add=froze
nset({'Screen Mom Screen Cleaner kit'}), confidence=0.4029850746268656, lift=3.1098673300165833)])
**********
```

C4: The First Three Rules:

The First three rules are as follows

1. If "5pack Nylon Braided USB C cables" then "HP 63XL Ink" with:

- Support = 0.0057
- Confidence = 0.3007 = 30%
- Lift = 3.7908

  Our confidence in this rule demonstrates that out of all customers who purchased the "5pack Nylon Braided USB C cables", 30% also purchased the "HP 63XL Ink". The simplest metric of support, with a value of 0.0057, demonstrates that a little more than half a percentage of all transactions contain both items. A lift value of 3.7908 demonstrates that once a customer has purchased the "5pack Nylon Braided USB C cables", they are 3.8 times more likely to also purchase the "HP 63XL Ink".

  #2. If "AutoFocus 1080p Webcam" then "SanDisk Ultra 64GB card" with:
- Support = 0.0053
- Confidence = 0.3774 = 38% of customers also purchased consequent
- Lift = 3.8407 = 3.8 times more likely to purchase consequent following purchase of antecedent

  #3. If "iPhone 11 case" then "HP 63XL Ink" with:
- Support = 0.0051
- Confidence = 0.3729 = 37% of customers also purchased consequent
- Lift = 4.7008 = 4.7 times more likely to purchase consequent following purchase of antecedent

PART IV. Analysis/Research

D1. The Importance of Support, Lift, and Confidence:

The metrics are compared using our top three rules.

- $Support = \frac{frequency(X,Y)}{N}$ = Giving us the number of total transactions containing this particular itemset.
- $Confidence = \frac{frequency(X,Y)}{frequency(X)}$ = Giving us a probability of the consequent given the antecedent.
- $Lift = \frac{Support}{Support(X)*Support(Y)}$ = Giving us the coefficient of likelihood given the antecedent; that is, how many more times likely is the consequent to be purchased once the antecedent has been purchased.

This study's findings seem unconvincing. None of the rules have a certainty level higher than 40%, much less than the prescribed value of 80% for importance.

The second rule has the highest level of confidence, at 38 percent, whereas the first rule has just 30 percent (based on its study in conjunction with our other metrics of relevance).

Support for the pairing of any of the top three rules' item sets occurs in less than half of one percent of all transactions and is hence not persuasive.

Finally, the lift ratio gives us some hope that if a buyer buys the preceding item, they will buy the subsequent item as well. The association between purchasing a "iPhone 11 cover" and then getting some "HP 63XL Ink" demonstrates our highest lift metric of "4.7 times more likely."

D2. The Importance of the Findings in Practice:

We don't believe these findings are useful because we can't guarantee that any item will be purchased even half of the time. Do we not have a better probability of correctly anticipating the outcome of a coin flip? When one of the antecedents, such as a camera, is purchased, the customer is about four times more likely to buy the consequent, such as a memory card.

So, if a half-percentage point of consumers buys a five-pack of USB cables, they are roughly four times as likely to buy HP ink for the printer.

These findings don't give us much to go on. Perhaps additional data is needed before we can make any conclusions. Of course, further investigation is advised.

D3. Plan of Action/Initiatives:

As a result, based on the preceding research and significant remark, we do not recommend that company decision-makers pursue the original notion of promoting our cellular service by offering discounted or even free goods in exchange for subscribing to our service. Our market basket analysis of this transaction dataset revealed not only little significance, but none of the pairings showed that customers who utilized telecom services would want or need a related item.

That is, if we discovered a substantial association between, say, we may recommend one things for a possible marketing offer and customer discount based on a number of transactions where customers acquired two linked telecoms accessories. That was not discovered. We discovered ink when we were hunting for a relationship where a webcam and an ethernet cable were both acquired at the same time.

There is currently no need to take any action. Prior to our data science team being able to firmly state, advise suggestions, little more data must be processed.

PART V. Documentary Evidence

E. Panopto recording:

https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=f112a31c-5dd1-4f58-8f44-ae5a0169b695

F. Third Party Evidence:

Title: Implementing Apriori algorithm in Python. GeeksForGeeks
Date: 2021
Author: A. Gupta.
URL: https://www.geeksforgeeks.org/implementing-apriori-algorithm-in-python/

Title: Hands on guide to MBA analysis with python.
Date: May 11th, 2020
Author: V. Kumar.
URL: https://analyticsindiamag.com/hands-on-guide-to-market-basket-analysis-with-python-codes/

Title: Guide to association rules mining from scratch
Date: Nov 30th, 2020.
Author: R. Umredkar.
URL: https://analyticsindiamag.com/guide-to-association-rule-mining-from-scratch/

G.Sources:

Title: A gentle introduction to Market Basket analysis- Association Rules
Date: Sept 24, 2017
Author: S Li.
URL: https://towardsdatascience.com/a-gentle-introduction-on-market-basket-analysis-association-rules-fa4b986a40ce

Title: Market Basket Analysis 101-Key concepts
Date: Nov 16th, 2020
Author: S. Sivek.
URL: https://towardsdatascience.com/market-basket-analysis-101-key-concepts-1ddc6876cd00