Retail Analytics

Individual Retail Analytics Project: Shopping Cart Analysis

Hult International Business School

Summer 2023

Individual Retail Analytics Project: Shopping Cart Analysis

1. Summary of the dataset

The present analysis is considering two dataset, the first dataset is the initial dataset that contains 9835 transactions and the second dataset is a subset of the initial dataset and it is conformed by the transactions that contains the product 'bottled beer' which is the 13th most frequently bought product out the top-30 most frequently products from the initial dataset. The new subset dataset for 'bottled beer' contains 792 transactions.

Top 30th most frequent products

The top 30th most frequent products are composed by transactions that contains one or two products being the 13th most frequently product "bottled beer" with a support metric of 0.081 which is over 0.001 and makes it a good candidate to find interesting rules associated to it. The list of the 30th most frequent products can be found in the source of code part.

2. Solution: Most strongly associated product pairs

The 5 most strongly associated products pairs are the following and are composed by two products in the antecedents and one product in the consequent. Each of this associated products have a confidence level over 0.5, lift metric over 2, and a support metric equal or greater than 0.1

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction	zhangs_metric	antecedent_len
540	(bottled beer, soups)	(whole milk)	0.015152	0.253788	0.013889	0.916667	3.611940	0.010044	8.954545	0.734266	2
240	(hamburger meat, bottled beer)	(whole milk)	0.026515	0.253788	0.021465	0.809524	3.189765	0.014735	3.917614	0.705196	2
564	(detergent, bottled beer)	(whole milk)	0.016414	0.253788	0.011364	0.692308	2.727899	0.007198	2.425189	0.643988	2
424	(soft cheese, bottled beer)	(other vegetables)	0.015152	0.200758	0.010101	0.666667	3.320755	0.007059	2.397727	0.709615	2
430	(bottled beer, cream cheese)	(other vegetables)	0.022727	0.200758	0.015152	0.666667	3.320755	0.010589	2.397727	0.715116	2

In order to perform a cross-selling and promotion strategy the bottled beer should be sold combining the antecedents itemset and the consequent products which are whole and other vegetables. Also, another sell strategy would be to place products such as whole milk and vegetables products next to bottled beers.

From this analysis we can interpret also that buyers of 'bottled beer' might consider the beer as a basic consumer good because it is bought among products such as whole milk, other vegetables, soups, hamburger meat and other dairy products.

3. Methodology

For the present analysis, it has been chosen Python as programming language and Tableau for visualization.

First step:

In the first step the libraries pandas and 'mlxtend.frequent_patterns' were loaded, and from the 'mlxtend.frequent_patterns' library were imported the function 'apriori' and 'associated rules'.
Also in the first step two datasets were storage in two variables called as 'df_string' with rows of strings which represent the product names of each transaction. The second dataset was called 'df_trans' and contained the binary matrix representation of the first dataset, where each row represent a transaction and each column a product.

Second step:

In the second step I found the 30th most frequently bought products from the whole dataset as well as the 13th most frequent product which was 'bottled beer". To find the 30th most frequently bought products was used the dataset "df_trans" in which was applied the function 'apriori' with min. threshold of 0.005 and was sorted by the support metric in descendent order.
From this list of 30 itemsets, the 13th product was selected to perform the association products analysis. The 13th product was "bottled beer".

Third step:

In this step the initial dataset was subsetted and filtered with transactions that contained the product "bottled beer". The new dataset was called as 'product_13', to this dataset I applied the function "apriori "with a min support of of 0.01, which identifies the frequent individual items. The new dataset was labeled as 'frequent_items'.

To this dataset 'frequent_items' I applied the function 'association_rules' considering a confidence metric with a min threshold of 0.5. The displayed output was a table called 'rules', where each row represents association rules between the antecedent and the consequent. The antecedents are conformed by items contained in a shopping cart and the consequent are the products that could be bought together or be place next to the other.

However in order to have the most interesting association rules I applied certain filters. First for the antecedents I considered two items (An extra column 'antecedent_len' was created to display the numbers of item per antecedent) where support is at least 0.001, confidence is over 0.5 and lift is above 2. Besides of this I have useD Tableau to visualize the interesting rules, where the highest point between the axes of convenience and lift represent the most interesting rules.

4. Source Code

I. Importing datasets

```
# importing the libraries
import pandas as pd
from mlxtend.frequent_patterns import apriori, association_rules
import matplotlib.pyplot as plt
```

```
# importing dataset transactions_strings.csv
df_string = pd.read_csv('transactions_strings.csv')
df_string
```

```
# calculating main statisitcs for transactions_strings.csv
df_string.describe()
```

	{citrus fruit,semi-finished bread,margarine,ready soups}
count	9834
unique	7010
top	{canned beer}
freq	260

```
# importing dataset transactions_binary.csv
df_trans = pd.read_csv('transactions_binary.csv')
df_trans
```

II. Task 1:

Top-30 most frequently bought products during the time period of the dataset

```
# top items by purchase frequency
freq_items = apriori(df_trans, min_support=0.005, use_colnames=True)
freq_item_30 = freq_items.sort_values(by = "support", ascending = False).head(30)
freq_item_30['Row Number'] = [i+1 for i, _ in enumerate(freq_item_30.index)]
print(freq_item_30)
### top 13th is bottled beer
```

	support	itemsets	Row Number
21	0.255516	(whole milk)	1
19	0.193493	(other vegetables)	2
47	0.183935	(rolls/buns)	3
80	0.174377	(soda)	4
26	0.139502	(yogurt)	5
79	0.110524	(bottled water)	6
16	0.108998	(root vegetables)	7
12	0.104931	(tropical fruit)	8
119	0.098526	(shopping bags)	9
1	0.093950	(sausage)	10
50	0.088968	(pastry)	11
11	0.082766	(citrus fruit)	12
83	0.080529	(bottled beer)	13
115	0.079817	(newspapers)	14
84	0.077682	(canned beer)	15
13	0.075648	(pip fruit)	16
394	0.074835	(other vegetables, whole milk)	17
82	0.072293	(fruit/vegetable juice)	18
27	0.071683	(whipped/sour cream)	19
49	0.064870	(brown bread)	20
46	0.063447	(domestic eggs)	21
0	0.058973	(frankfurter)	22
60	0.058566	(margarine)	23
77	0.058058	(coffee)	24
8	0.057651	(pork)	25
461	0.056634	(rolls/buns, whole milk)	26
449	0.056024	(whole milk, yogurt)	27
22	0.055414	(butter)	28
23	0.053279	(curd)	29
9	0.052466	(beef)	30

III. Task 2:

Top 5 most promising product association rules that involve the 13th most frequently bought product

```
## From the initial dataset subset the transaction that contains bottled beer
product_13 = df_trans.loc[df_trans['bottled beer']==1]
product_13
```

```
# creating a frequent itemset with the apriori function, the minimum threshold
# for the support metric is 0.01
frequent_items = apriori(product_13, min_support=0.01,use_colnames=True)
frequent_items
```

```
# genereting rules using the function assciation_rules and confidence as evaluation
# metric, setting the minimun threshold
rules = association_rules(frequent_items, metric='confidence', min_threshold=0.5)
rules
```

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction	zhangs_metric
0	(frankfurter)	(whole milk)	0.066919	0.253788	0.034091	0.509434	2.007322	0.017108	1.521125	0.537814

```
# creating an additional column called "antecedent_len" which will
# contain the number of items in the antecedent
rules["antecedent_len"] = rules["antecedents"].apply(lambda x: len(x))
rules
```

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction	zhangs_metric
0	(frankfurter)	(whole milk)	0.066919	0.253788	0.034091	0.509434	2.007322	0.017108	1.521125	0.537814
1	(frankfurter)	(bottled beer)	0.066919	1.000000	0.066919	1.000000	1.000000	0.000000	inf	0.000000
2	(sausage)	(bottled beer)	0.097222	1.000000	0.097222	1.000000	1.000000	0.000000	inf	0.000000

```
## filtering the new subset dataset that includes bottle beer with
## the following conditions:
## 1. number of item in the antecedent equal to 2
## 2. support metric greater than 0.001
## 3. confidence metric greater than 0.4
## 4. lift metric greater than 2
## ordering the dataset in descendent order according to the highest
## values for confidence and lift
interesting_rules = rules[ (rules['antecedent_len'] == 2) & # to consider 3 products
(rules['support'] >= 0.001) &
(rules['confidence'] > 0.5) &
(rules['lift'] > 2) ]
interesting_rules
interesting_rules = interesting_rules[ interesting_rules["antecedents"].apply(lambda x: "bottled beer" in x) ]
interesting_rules
#interesting_rules.sort_values(by=['antecedent_len'], ascending=False).head(5)
interesting_rules.sort_values(by=['confidence','lift'], ascending=False).head(5)
```

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction	zhangs_metric	antecedent_len
540	(bottled beer, soups)	(whole milk)	0.015152	0.253788	0.013889	0.916667	3.611940	0.010044	8.954545	0.734266	2
240	(hamburger meat, bottled beer)	(whole milk)	0.026515	0.253788	0.021465	0.809524	3.189765	0.014735	3.917614	0.705196	2
564	(detergent, bottled beer)	(whole milk)	0.016414	0.253788	0.011364	0.692308	2.727899	0.007198	2.425189	0.643988	2
424	(soft cheese, bottled beer)	(other vegetables)	0.015152	0.200758	0.010101	0.666667	3.320755	0.007059	2.397727	0.709615	2
430	(bottled beer, cream cheese)	(other vegetables)	0.022727	0.200758	0.015152	0.666667	3.320755	0.010589	2.397727	0.715116	2

Visualization of most interesting rules