ASSOCIATION RULES – MARKET BASKET ANALYSIS

Susana Reche Rodríguez
Student Number: *17165628*
Date: *24.07.2020*

Data Mining
Higher Diploma in Data Analytics

Table of Contents

Table of Images

Part 1: *60 Marks*

Using the techniques outlined by the paper from Hadley Wickham, entitled "Tidy Data" (Wickham, 2014) and using RStudio, apply the principles of Tidy Data to the Groceries Dataset attached in Moodle and create a programme in RStudio to run the Association Rules Algorithm.
In addition to the code you are required to provide a detailed report, clearly identifying the techniques of Tidy Data used in the code and also to report the outcomes using standard statistical techniques that can be applied to Association Rules.
The final report should not include the code and be clearly sectioned, correctly referenced and not exceed 1000 words. Code must be uploaded using the dedicated link.

PART 1: Unsupervised Machine Learning – Association Rules

1. Executive Summary

The Association Rules algorithm is also called Market Basket Analysis as has been applied mostly to learn about purchasing patterns (Lantz, 2015), even if it can be applied to find patterns in many other cases.

During the project, the first step is cleaning the data set "Groceries.csv" and preparing it into a transactional format to be able to be used by the Apriori algorithm. Then the transactional data is explored to understand the most frequent item sets and fins appropriate rules (under certain values for the interest measures). Different visualizations are used in order to understand the data. Finally, some recommendations are given based on the selected rule.

2. Objective

The objective of the project is to find items that are frequently bought together to be able:

- Place frequently bought together items nearby to increase sales on-site
- Increase online revenue by offering related items to the user
- Creating promotional campaigns (email, social media,…) with those items which were bought together

3. Data Cleaning – Convert to Transactional Data

The "Groceries.csv" file is a comma-separated csv file with 1499 transactions and a maximum of 34 items per transaction.

Some data cleaning was performed :

- the white spaces have been trimmed, the empty rows ignored and the NA values marked as NA when importing to R
- the date, which was concatenated with the first item, was split and removed as is not needed (date range: January 2000 to February 2002, 28 months)
- the transactionID was added for each transaction

The Apriori algorithm needs transactional data which can be shaped in two formats: basket or single. The basket format contains a transaction per line with the items separated by commas in one column. The single format contains two columns, the first column is the transaction ID and the second is an item. In both cases, the duplicates were removed to ensure the same item is not twice in a transaction. Both formats have been tested, with the same results. The basket format can also use a binary matrix as per (Arnold, 2018), but has not been tested.

A series of transformations were done to obtain the single format:

- add transaction ID
- move the items one per row using the melt function.

The tidy data principle establish that each variable needs to be in a column, in this case, both formats follow the principle, but probably the single format is cleaner, as contains one item per row.

When prepared for single format the number of unique products was checked, there are 38.

	x	freq
1	all- purpose	1
2	aluminum foil	1
3	bagels	1
4	beef	1
5	butter	1
6	cereals	1
7	cheeses	1
8	coffee/tea	1
9	dinner rolls	1
10	dishwashing liquid/detergent	1
11	eggs	1
12	flour	1
13	fruits	1
14	hand soap	1
15	ice cream	1
16	individual meals	1
17	juice	1
18	ketchup	1
19	laundry detergent	1
20	lunch meat	1
21	milk	1
22	mixes	1
23	paper towels	1
24	pasta	1
25	pork	1
26	poultry	1
27	sandwich bags	1
28	sandwich loaves	1
29	shampoo	1
30	soap	1
31	soda	1
32	spaghetti sauce	1
33	sugar	1
34	toilet paper	1
35	tortillas	1
36	vegetables	1
37	waffles	1
38	yogurt	1

Figure 1: List of unique products contained in the data set

4. Transactional Data Exploration

The data frame has been successfully transformed into transactional, the number of transactions (1,499) and items (38) is exactly the same as before the transformation as seen in the summary [Fig. 2]. 38.36% of the cells from the sparse matrix (Jabeen, 2018) are filled in with products and the rest are empty.

The items within the transactions seem to be evenly distributed as per [Fig.3] and the size of the transactions normally distributed [Fig. 6]. 50% of the transactions contain between 10 and 19 items, the minimum number of unique items within a transaction are 4 and the maximum 27, as per [Fig. 3].

The most popular items are vegetables which appear in 1,089 transactions (72.64%), followed by poultry which appears in 613 transactions (40.89%)as seen in [Fig. 4] and [Fig.5].

```
transactions as itemMatrix in sparse format with
 1499 rows (elements/itemsets/transactions) and
 38 columns (items) and a density of 0.3836242

most frequent items:
                  vegetables                     poultry                     waffles dishwashing liquid/detergent
                        1089                         613                         587                         585
                   ice cream                    (Other)
                         584                       18394

element (itemset/transaction) length distribution:
sizes
  4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27
 15  57  56  53  71  74  72  79  67  72  89  86  84 105  95  94 114  78  67  36  24   7   3   1

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   4.00   10.00   15.00   14.58   19.00   27.00

includes extended item information - examples:
         labels
1  all- purpose
2 aluminum foil
3        bagels

includes extended transaction information - examples:
  transactionID
1             1
2            10
3           100
```

Figure 2: Summary of groceries data set once converted into transactional data

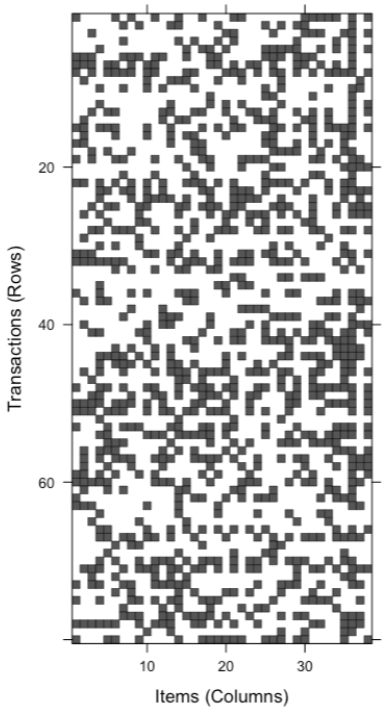


Figure 3: Visualization of 80 random transactions and its items

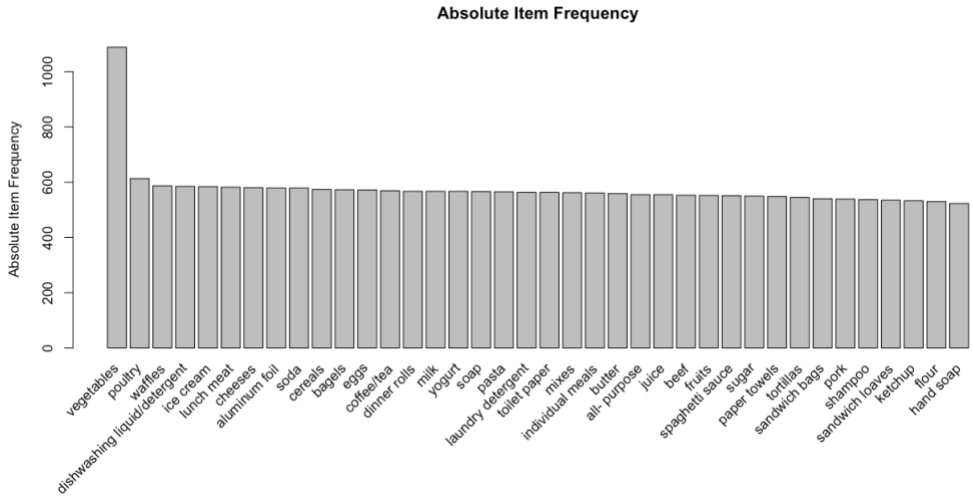


Figure 4: Absolute Item Frequency Plot

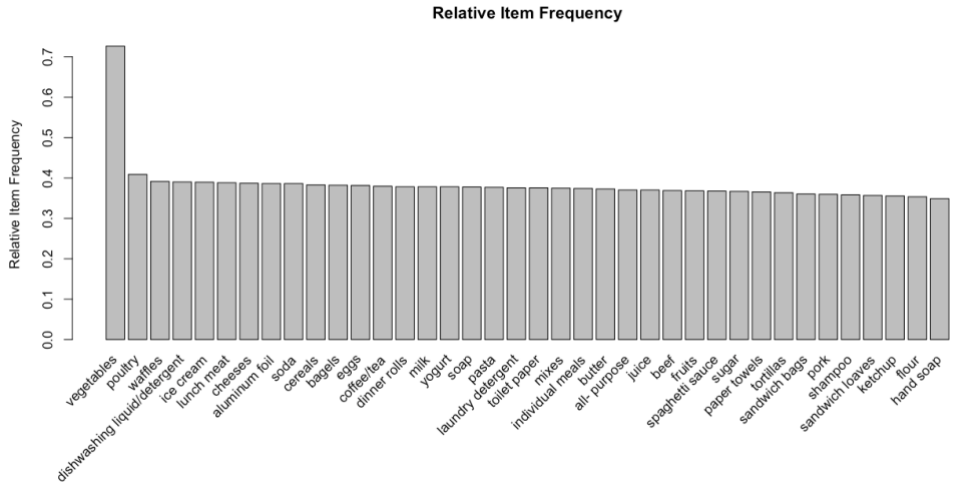


Figure 5: Relative Item Frequency Plot

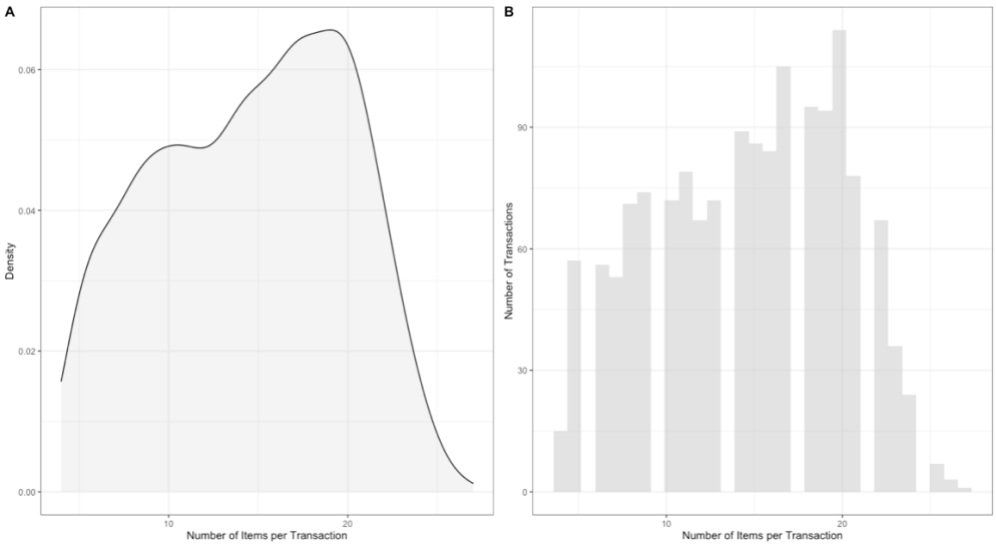


Figure 6: Distribution of items per transaction

5. Item sets

When setting the support to 0.01, the confidence to 0.8 and the minimum number of items within a set to 2, the result is 1,071,312 of item sets [Fig. 7]. As vegetables appear in 72.64% of the transaction the decision is to take vegetables out of the picture and evaluate the rest of the items, which result in 619,710 item sets [Fig. 8].

As there are still many item sets to evaluate, the result is restricted to have a minimum 3 items in the item set, the total number then reduces to 619,044 item sets (still a huge number)[Fig.9].

```
set of 1071312 itemsets

most frequent items:
                  vegetables                      poultry                        soda dishwashing liquid/detergent
                      451602                       182217                      168388                        166294
                  lunch meat                     (Other)
                      160092                      4698881

element (itemset/transaction) length distribution:sizes
     2      3      4      5      6      7      8
   703   8436  73815 470103 472875  45305     75

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   2.00    5.00    5.00    5.44    6.00    8.00

summary of quality measures:
    support        transIdenticalToItemsets     count
 Min.   :0.01001   Min.   :0.000e+00        Min.   : 15.00
 1st Qu.:0.01134   1st Qu.:0.000e+00        1st Qu.: 17.00
 Median :0.01334   Median :0.000e+00        Median : 20.00
 Mean   :0.01573   Mean   :5.730e-08        Mean   : 23.57
 3rd Qu.:0.01601   3rd Qu.:0.000e+00        3rd Qu.: 24.00
 Max.   :0.32021   Max.   :6.671e-04        Max.   :480.00

includes transaction ID lists: FALSE

mining info:
                  data ntransactions support confidence
 groceries_transactions          1499    0.01          1
```

Figure 7: Summary Apriori item sets (sup=0.01, conf = 0.8, minlen = 2)

```
Apriori

Parameter specification:
 confidence minval smax arem  aval originalSupport maxtime support minlen maxlen           target  ext
         NA    0.1    1 none FALSE            TRUE       5    0.01      2     10 frequent itemsets TRUE

Algorithmic control:
 filter tree heap memopt load sort verbose
    0.1 TRUE TRUE  FALSE TRUE    2    TRUE

Absolute minimum support count: 14

set item appearances ...[1 item(s)] done [0.00s].
set transactions ...[38 item(s), 1499 transaction(s)] done [0.00s].
sorting and recoding items ... [37 item(s)] done [0.00s].
creating transaction tree ... done [0.00s].
checking subsets of size 1 2 3 4 5 6 7 done [0.53s].
sorting transactions ... done [0.00s].
writing ... [619710 set(s)] done [0.17s].
creating S4 object  ... done [0.18s].
```

Figure 8: Summary Apriori item sets (sup=0.01, conf = 0.8, minlen = 2) without vegetables

```
set of 619044 itemsets

most frequent items:
                    poultry                       soda dishwashing liquid/detergent                    ice cream
                     106153                                                   97718                        97272                        94483
                  lunch meat                                                (Other)
                      93549                                                 2666105

element (itemset/transaction) length distribution:sizes
     3      4      5      6      7
  7770  66045 404058 140697    474

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  3.000   5.000   5.000   5.097   5.000   7.000

summary of quality measures:
    support        transIdenticalToItemsets     count
 Min.   :0.01001   Min.   :0.000e+00        Min.   : 15.00
 1st Qu.:0.01134   1st Qu.:0.000e+00        1st Qu.: 17.00
 Median :0.01334   Median :0.000e+00        Median : 20.00
 Mean   :0.01584   Mean   :1.035e-07        Mean   : 23.75
 3rd Qu.:0.01668   3rd Qu.:0.000e+00        3rd Qu.: 25.00
 Max.   :0.09273   Max.   :6.671e-04        Max.   :139.00

includes transaction ID lists: FALSE

mining info:
                 data ntransactions support confidence
 groceries_transactions          1499    0.01          1
```

Figure 9: Summary Apriori item sets (sup=0.01, conf = 0.8, minlen = 3) without vegetables

6. Rules and Interest Measures

With support of 0.01, confidence of 0.8, a minimum length of 3 items per item set and excluding vegetables the Apriori algorithm generate 9,660 rules [Fig.10]. The support is then modified to 0.02 and rules reduce to 14 [Fig.11].

```
Apriori

Parameter specification:
 confidence minval smax arem  aval originalSupport maxtime support minlen maxlen target  ext
        0.8    0.1    1 none FALSE            TRUE       5    0.01      3     10  rules TRUE

Algorithmic control:
 filter tree heap memopt load sort verbose
    0.1 TRUE TRUE  FALSE TRUE    2    TRUE

Absolute minimum support count: 14

set item appearances ...[1 item(s)] done [0.00s].
set transactions ...[38 item(s), 1499 transaction(s)] done [0.00s].
sorting and recoding items ... [37 item(s)] done [0.00s].
creating transaction tree ... done [0.00s].
checking subsets of size 1 2 3 4 5 6 7 done [0.53s].
writing ... [9660 rule(s)] done [0.07s].
creating S4 object  ... done [0.06s].
```

Figure 10: Summary Apriori item sets (sup=0.01, conf = 0.8, minlen = 3) without vegetables

Support

Supports refers to how many times the combination of items or itemset occurs in the dataset. So for support of 0.02 [Fig. 11] , the item set needs to appear in 29.98 transactions (0.02 * 1,499). Taking into consideration that in the data set there is an average of 53.53 transactions per month (1,499 / 28 months), 29.98 transactions can be considered relevant.

Confidence

Confidence refers to the probability of the item B being part of the itemset if another item A is present. In the default setting the confidence is set as 0.8, B needs to appear in a minimum of 80% of the item sets which contain A. The objective is generating a rule where both items appear most of the time together.

Lift

The lift can be defined as per [Fig. 12], confidence divided the expected confidence. When support is higher than confidence then lift is negative correlated, and vice versa. If confidence is equal to support then the lift is 1, which means that the items are independent (Raj, 2020). If the items are independent it won't be possible to create any rule (Wikipedia, n.d.).

In this case, the goal is to have a lift higher than 1 (and the higher the better), which imply a positive relationship, and high confidence that the items appear most of the times together.

$$Support = \frac{Number\ of\ transactions\ with\ both\ A\ and\ B}{Total\ number\ of\ transactions} = P(A \cap B)$$

$$Confidence = \frac{Number\ of\ transactions\ with\ both\ A\ and\ B}{Total\ number\ of\ transactions\ with\ A} = \frac{P(A \cap B)}{P(A)}$$

$$ExpectedConfidence = \frac{Number\ of\ transactions\ with\ B}{Total\ number\ of\ transactions} = P(B)$$

$$Lift = \frac{Confidence}{Expected\ Confidence} = \frac{P(A \cap B)}{P(A) \,.\, P(B)}$$

Figure 11: Support, Confidence and Lift (Prabhakaran, n.d.)

There are many other measures as per (Wikipedia, n.d.), which can help to find the most appropriate rules as Conviction, All-confidence, Collective strength or Leverage among others.

```
Apriori

Parameter specification:
 confidence minval smax arem  aval originalSupport maxtime support minlen maxlen target  ext
        0.8    0.1    1 none FALSE            TRUE       5    0.02      3     10  rules TRUE

Algorithmic control:
 filter tree heap memopt load sort verbose
    0.1 TRUE TRUE  FALSE TRUE    2    TRUE

Absolute minimum support count: 29

set item appearances ...[1 item(s)] done [0.00s].
set transactions ...[38 item(s), 1499 transaction(s)] done [0.00s].
sorting and recoding items ... [37 item(s)] done [0.00s].
creating transaction tree ... done [0.00s].
checking subsets of size 1 2 3 4 5 6 done [0.09s].
writing ... [14 rule(s)] done [0.01s].
creating S4 object  ... done [0.01s].
```

Figure 12: Summary Apriori rules (sup=0.02, conf = 0.8, minlen = 3) without vegetables

	rules	support	confidence	coverage	lift	count	conviction
1	{fruits,hand soap,poultry,sugar} => {beef}	0.02134757	0.8000000	0.02668446	2.168535	32	3.155437
2	{beef,coffee/tea,hand soap,sugar} => {poultry}	0.02268179	0.8292683	0.02735157	2.027852	34	3.461927
3	{dinner rolls,hand soap,spaghetti sauce,sugar} => {poultry}	0.02935290	0.8301887	0.03535690	2.030102	44	3.480691
4	{coffee/tea,dinner rolls,hand soap,spaghetti sauce} => {poultry}	0.02468312	0.8222222	0.03002001	2.010622	37	3.324716
5	{dinner rolls,hand soap,soap,spaghetti sauce} => {poultry}	0.02134757	0.8000000	0.02668446	1.956281	32	2.955304
6	{cereals,paper towels,sandwich bags,sugar} => {cheeses}	0.02401601	0.8000000	0.03002001	2.067586	36	3.065377
7	{fruits,poultry,sugar,toilet paper} => {beef}	0.02668446	0.8000000	0.03335557	2.168535	40	3.155437
8	{beef,dinner rolls,spaghetti sauce,sugar} => {poultry}	0.02601734	0.8125000	0.03202135	1.986847	39	3.152324
9	{beef,dinner rolls,ice cream,spaghetti sauce} => {poultry}	0.02334890	0.8139535	0.02868579	1.990402	35	3.176951
10	{dinner rolls,laundry detergent,spaghetti sauce,sugar} => {poultry}	0.02801868	0.8235294	0.03402268	2.013818	42	3.349344
11	{laundry detergent,milk,paper towels,spaghetti sauce} => {dishwashing liquid/detergent}	0.02401601	0.8372093	0.02868579	2.145259	36	3.745545
12	{bagels,beef,spaghetti sauce,sugar,toilet paper} => {poultry}	0.02001334	0.8823529	0.02268179	2.157662	30	5.024016
13	{bagels,beef,poultry,spaghetti sauce,toilet paper} => {sugar}	0.02001334	0.8571429	0.02334890	2.336104	30	4.431621
14	{bagels,poultry,spaghetti sauce,sugar,toilet paper} => {beef}	0.02001334	0.8333333	0.02401601	2.258891	30	3.786524

Figure 13: 14 Apriori rules (sup=0.02, conf = 0.8, minlen = 3) without vegetables + conviction

By looking at the final 14 rules, in general, most of the rules move around “poultry” and “beef” [Fig. 13][Fig. 16]. Even though the highest support is for poultry (size of the arrow), the lines with stronger lift (colour) are the ones pointing to sugar and beef [Fig. 17][Fig. 18].

Rules 13 and 14 have the highest lift [Fig. 14][Fig. 16] and some of the highest conviction, which means that there is a high dependency between the items (mlxtend, n.d.).

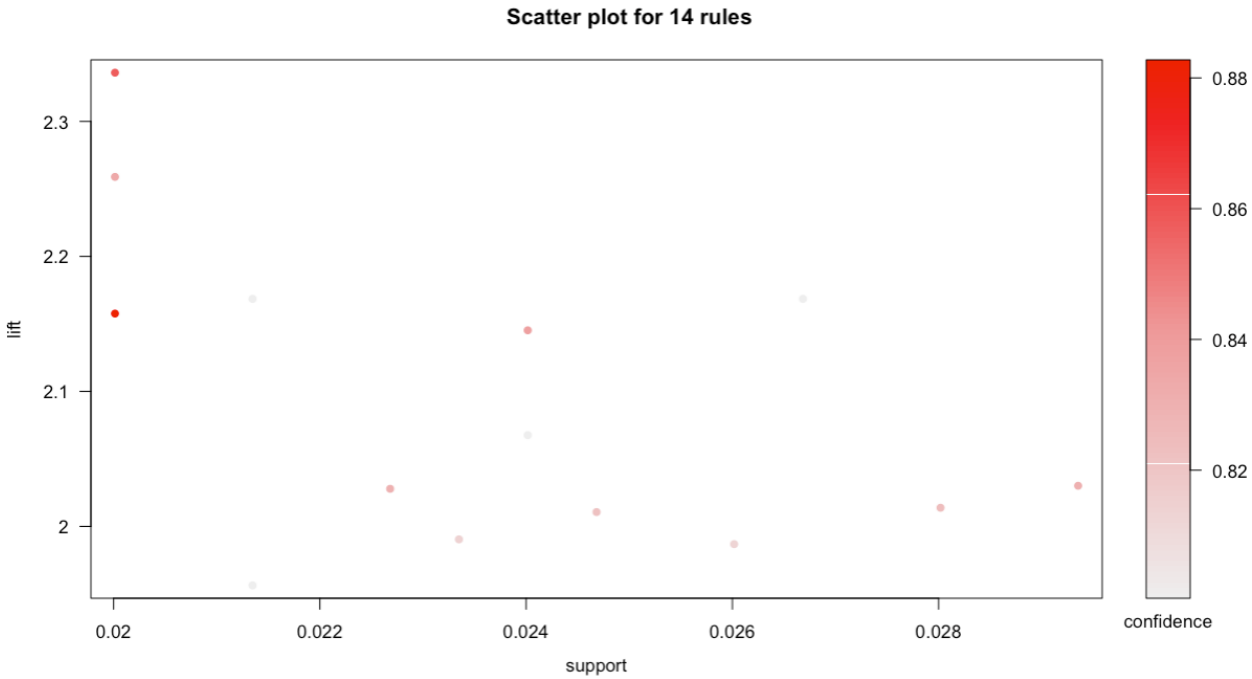


Figure 14: Scatter plot for 14 rules (support - lift)

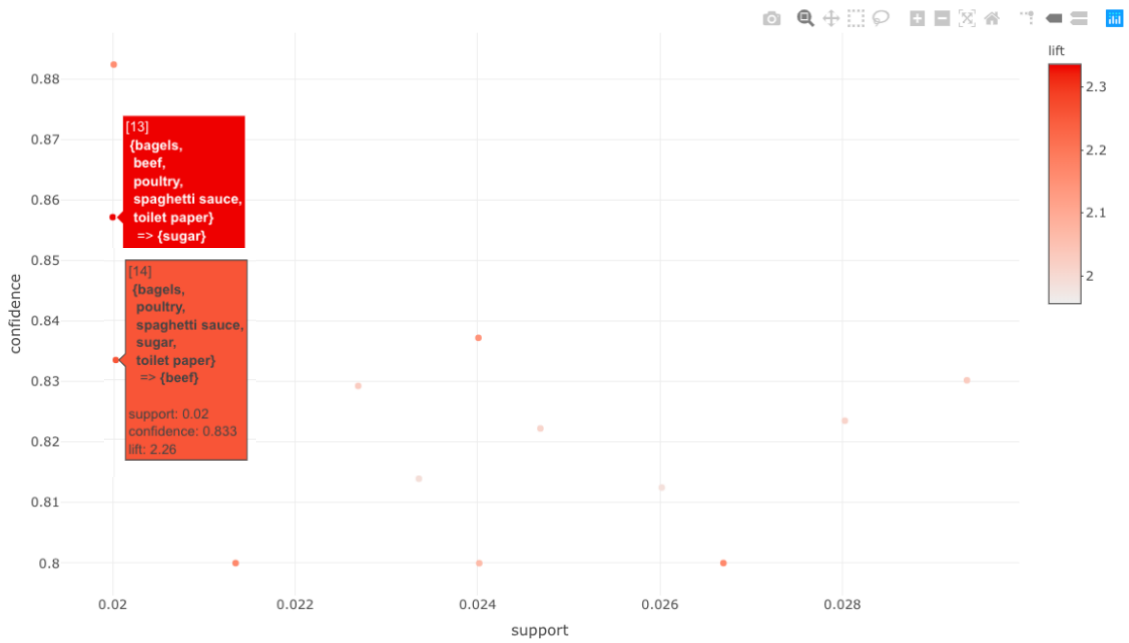


Figure 15: Scatter plot for 14 rules (support - confidence)

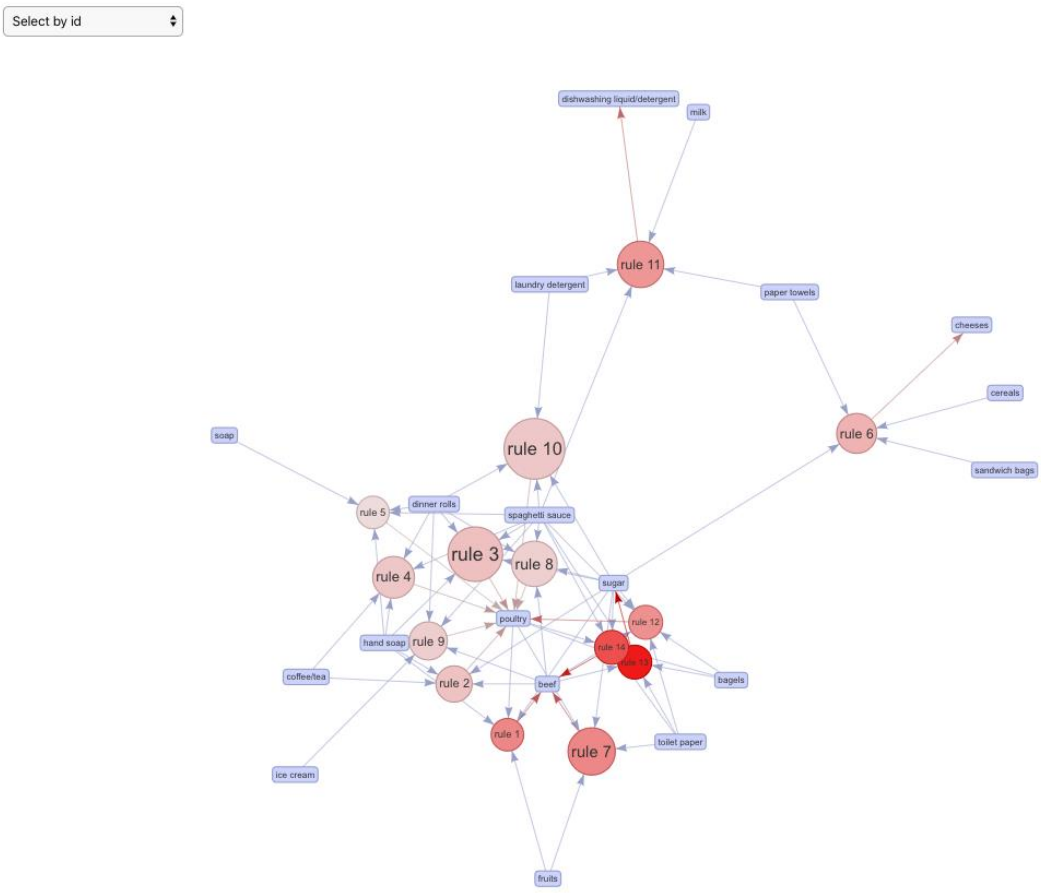


Figure 16: Graph for 14 rules - Interaction between Items (color > lift, size > support)

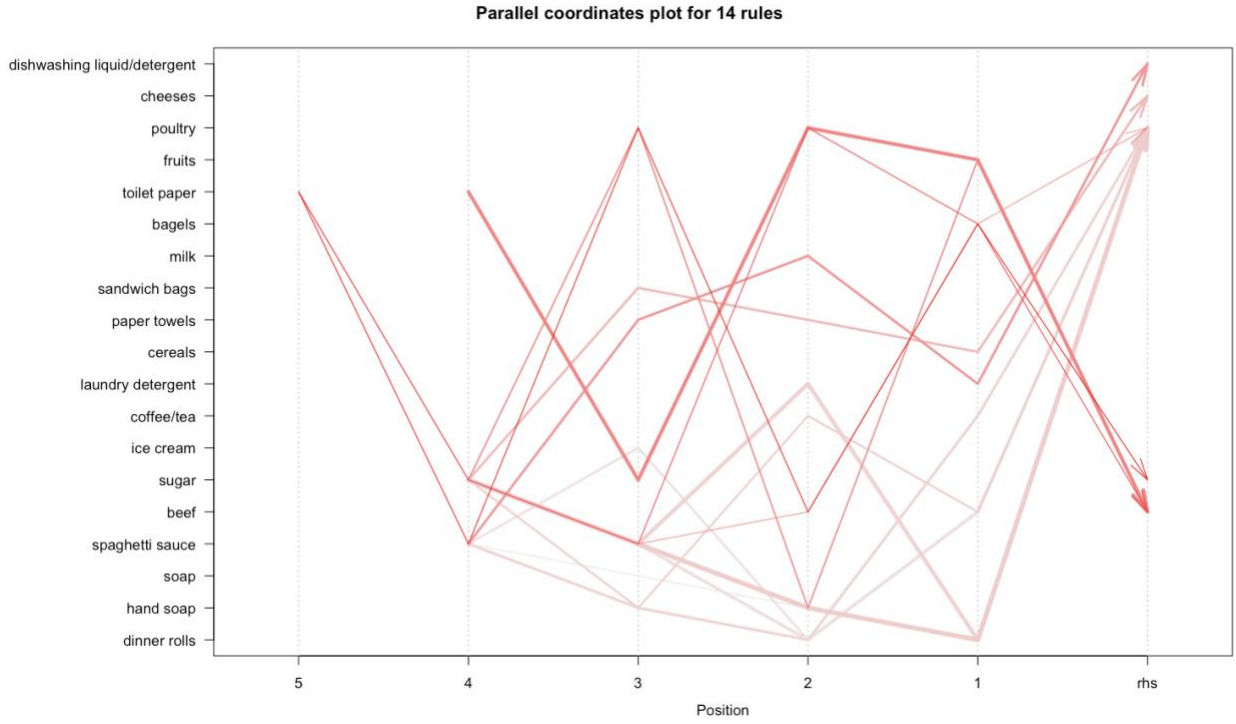


Figure 17: Parallel coordinates plot for 14 rules

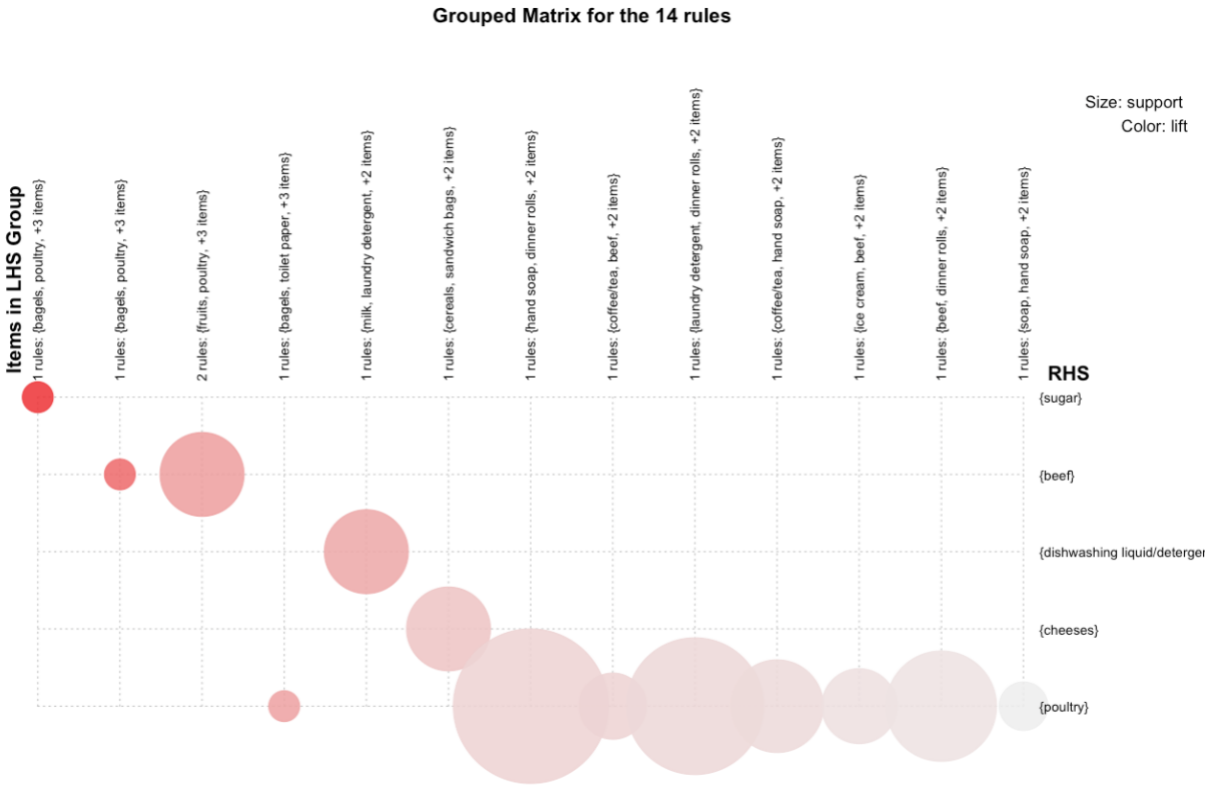


Figure 18: Grouped Matrix for the 14 selected rules (Dinov, 2020)

7. Rules with the highest lift

The two rules with higher lift are selected to inspect them in isolation.

```
    lhs                                                  rhs     support    confidence coverage   lift     count
[1] {bagels,beef,poultry,spaghetti sauce,toilet paper}  => {sugar} 0.02001334 0.8571429  0.02334890 2.336104 30
[2] {bagels,poultry,spaghetti sauce,sugar,toilet paper} => {beef}  0.02001334 0.8333333  0.02401601 2.258891 30
```

Figure 19: Subset of rules with the highest lift (lift > 2.2)

As per [Fig. 20] and [Fig. 21] the higher confidence get achieved when all the elements are combined together for both rules.

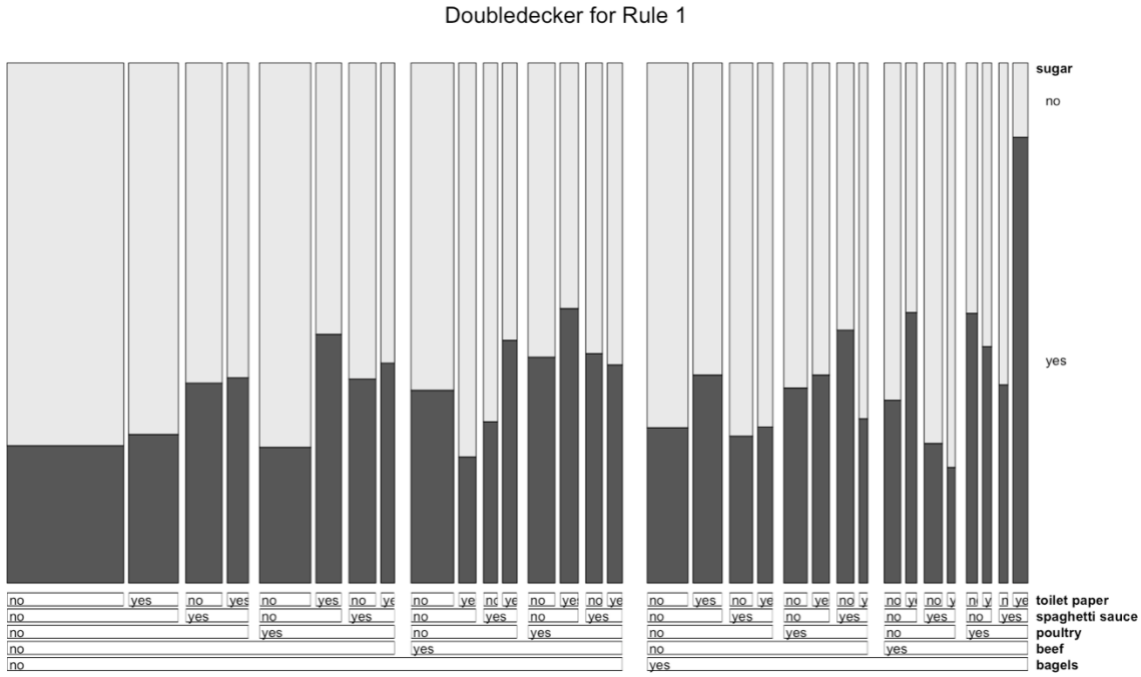


Figure 20: Double-decker graph for rule 1 (area > support, high > confidence)

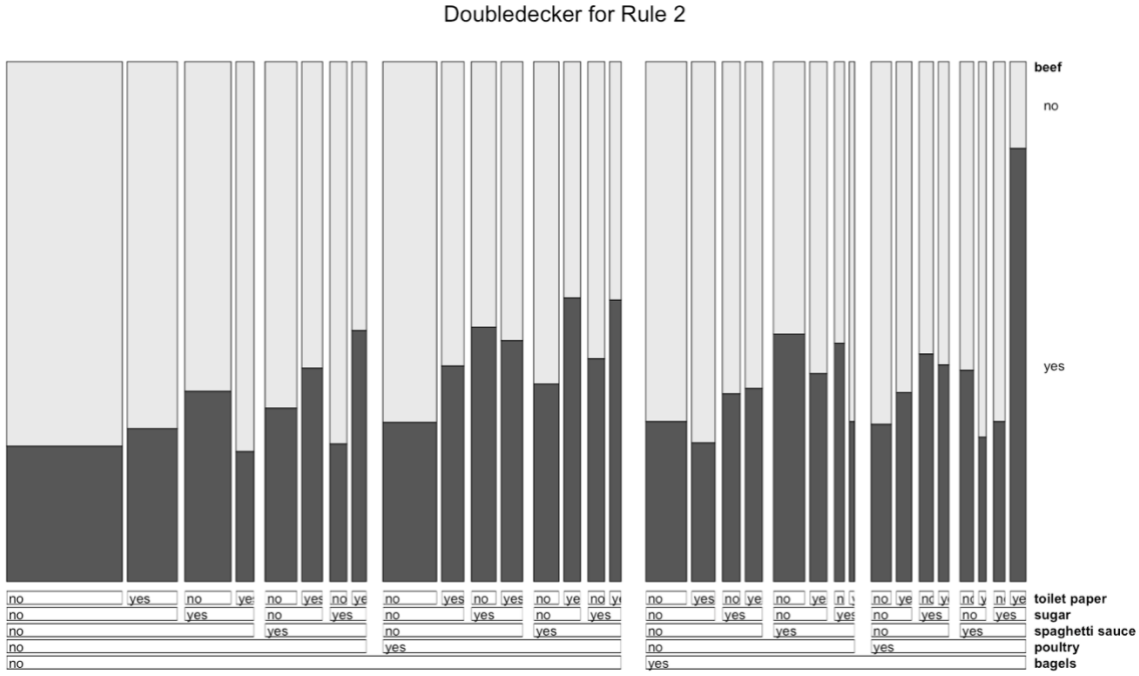
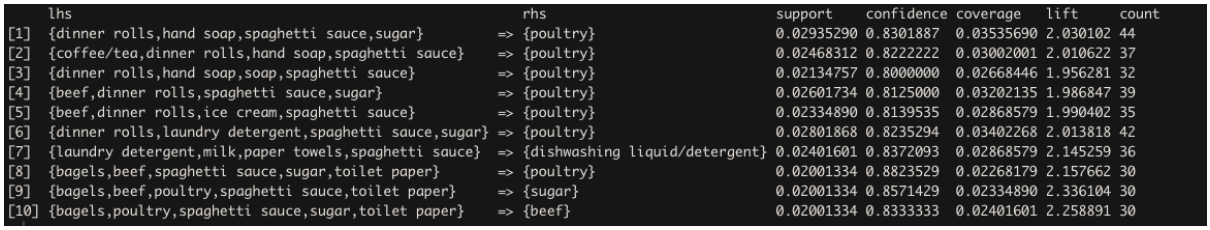


Figure 21: Double-decker graph for rule 1 (area > support, high > confidence)

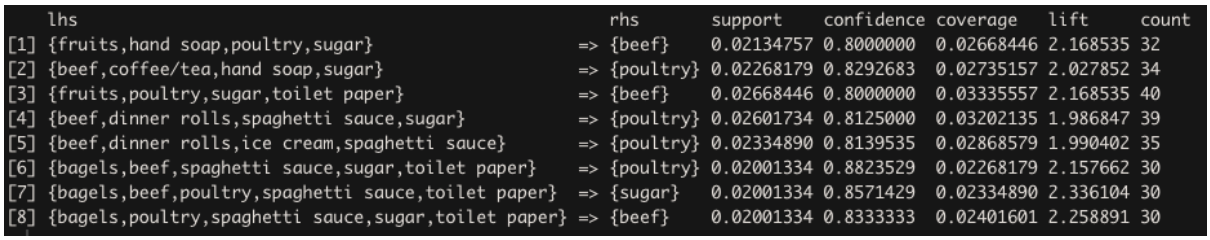
8. Rule 2 - Beef

When finding the rules for the elements contained in the previous rule 2 (“bagels”, ”poultry”, “spaghetti sauce”, “sugar” and ”toilet paper”), many of them seem to be bought together in all the cases, which confirm that there is some kind of relation.



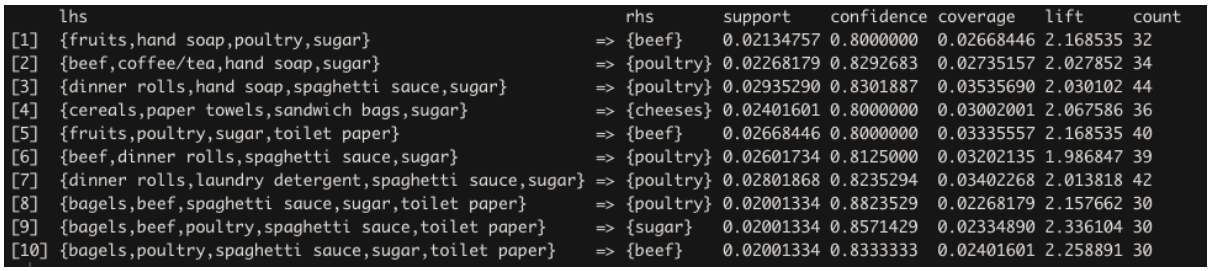
	lhs	rhs	support	confidence	coverage	lift	count
[1]	{dinner rolls,hand soap,spaghetti sauce,sugar}	=> {poultry}	0.02935290	0.8301887	0.03535690	2.030102	44
[2]	{coffee/tea,dinner rolls,hand soap,spaghetti sauce}	=> {poultry}	0.02468312	0.8222222	0.03002001	2.010622	37
[3]	{dinner rolls,hand soap,soap,spaghetti sauce}	=> {poultry}	0.02134757	0.8000000	0.02668446	1.956281	32
[4]	{beef,dinner rolls,spaghetti sauce,sugar}	=> {poultry}	0.02601734	0.8125000	0.03202135	1.986847	39
[5]	{beef,dinner rolls,ice cream,spaghetti sauce}	=> {poultry}	0.02334890	0.8139535	0.02868579	1.990402	35
[6]	{dinner rolls,laundry detergent,spaghetti sauce,sugar}	=> {poultry}	0.02801868	0.8235294	0.03402268	2.013818	42
[7]	{laundry detergent,milk,paper towels,spaghetti sauce}	=> {dishwashing liquid/detergent}	0.02401601	0.8372093	0.02868579	2.145259	36
[8]	{bagels,beef,spaghetti sauce,sugar,toilet paper}	=> {poultry}	0.02001334	0.8823529	0.02268179	2.157662	30
[9]	{bagels,beef,poultry,spaghetti sauce,toilet paper}	=> {sugar}	0.02001334	0.8571429	0.02334890	2.336104	30
[10]	{bagels,poultry,spaghetti sauce,sugar,toilet paper}	=> {beef}	0.02001334	0.8333333	0.02401601	2.258891	30

Figure 22: Rules which include Spaghetti Sauce



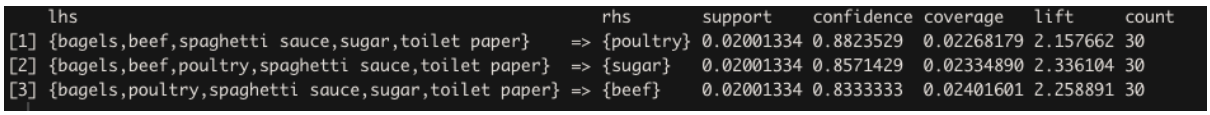
	lhs	rhs	support	confidence	coverage	lift	count
[1]	{fruits,hand soap,poultry,sugar}	=> {beef}	0.02134757	0.8000000	0.02668446	2.168535	32
[2]	{beef,coffee/tea,hand soap,sugar}	=> {poultry}	0.02268179	0.8292683	0.02735157	2.027852	34
[3]	{fruits,poultry,sugar,toilet paper}	=> {beef}	0.02668446	0.8000000	0.03335557	2.168535	40
[4]	{beef,dinner rolls,spaghetti sauce,sugar}	=> {poultry}	0.02601734	0.8125000	0.03202135	1.986847	39
[5]	{beef,dinner rolls,ice cream,spaghetti sauce}	=> {poultry}	0.02334890	0.8139535	0.02868579	1.990402	35
[6]	{bagels,beef,spaghetti sauce,sugar,toilet paper}	=> {poultry}	0.02001334	0.8823529	0.02268179	2.157662	30
[7]	{bagels,beef,poultry,spaghetti sauce,toilet paper}	=> {sugar}	0.02001334	0.8571429	0.02334890	2.336104	30
[8]	{bagels,poultry,spaghetti sauce,sugar,toilet paper}	=> {beef}	0.02001334	0.8333333	0.02401601	2.258891	30

Figure 23: Rules which include Beef



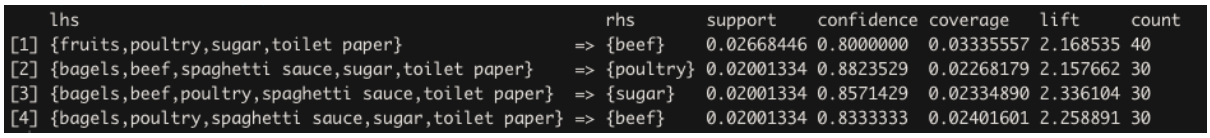
	lhs	rhs	support	confidence	coverage	lift	count
[1]	{fruits,hand soap,poultry,sugar}	=> {beef}	0.02134757	0.8000000	0.02668446	2.168535	32
[2]	{beef,coffee/tea,hand soap,sugar}	=> {poultry}	0.02268179	0.8292683	0.02735157	2.027852	34
[3]	{dinner rolls,hand soap,spaghetti sauce,sugar}	=> {poultry}	0.02935290	0.8301887	0.03535690	2.030102	44
[4]	{cereals,paper towels,sandwich bags,sugar}	=> {cheeses}	0.02401601	0.8000000	0.03002001	2.067586	36
[5]	{fruits,poultry,sugar,toilet paper}	=> {beef}	0.02668446	0.8000000	0.03335557	2.168535	40
[6]	{beef,dinner rolls,spaghetti sauce,sugar}	=> {poultry}	0.02601734	0.8125000	0.03202135	1.986847	39
[7]	{dinner rolls,laundry detergent,spaghetti sauce,sugar}	=> {poultry}	0.02801868	0.8235294	0.03402268	2.013818	42
[8]	{bagels,beef,spaghetti sauce,sugar,toilet paper}	=> {poultry}	0.02001334	0.8823529	0.02268179	2.157662	30
[9]	{bagels,beef,poultry,spaghetti sauce,toilet paper}	=> {sugar}	0.02001334	0.8571429	0.02334890	2.336104	30
[10]	{bagels,poultry,spaghetti sauce,sugar,toilet paper}	=> {beef}	0.02001334	0.8333333	0.02401601	2.258891	30

Figure 24: Rules which include Sugar



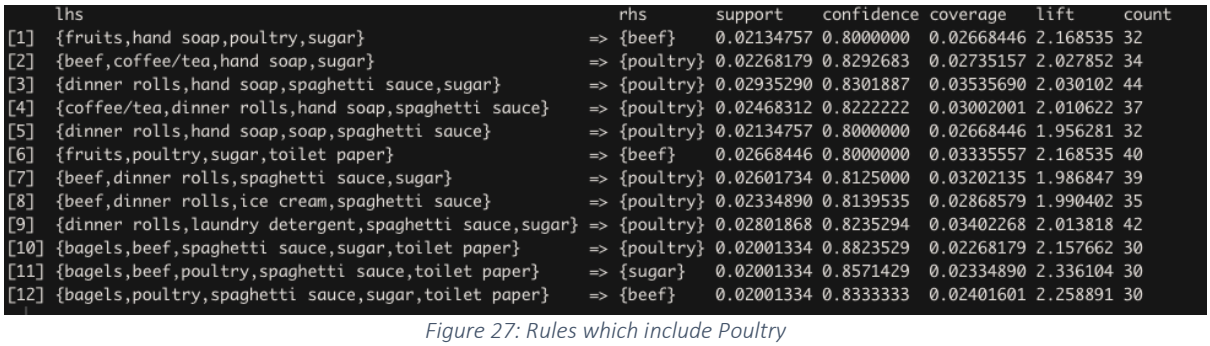
	lhs	rhs	support	confidence	coverage	lift	count
[1]	{bagels,beef,spaghetti sauce,sugar,toilet paper}	=> {poultry}	0.02001334	0.8823529	0.02268179	2.157662	30
[2]	{bagels,beef,poultry,spaghetti sauce,toilet paper}	=> {sugar}	0.02001334	0.8571429	0.02334890	2.336104	30
[3]	{bagels,poultry,spaghetti sauce,sugar,toilet paper}	=> {beef}	0.02001334	0.8333333	0.02401601	2.258891	30

Figure 25: Rules which include Bagels



	lhs	rhs	support	confidence	coverage	lift	count
[1]	{fruits,poultry,sugar,toilet paper}	=> {beef}	0.02668446	0.8000000	0.03335557	2.168535	40
[2]	{bagels,beef,spaghetti sauce,sugar,toilet paper}	=> {poultry}	0.02001334	0.8823529	0.02268179	2.157662	30
[3]	{bagels,beef,poultry,spaghetti sauce,toilet paper}	=> {sugar}	0.02001334	0.8571429	0.02334890	2.336104	30
[4]	{bagels,poultry,spaghetti sauce,sugar,toilet paper}	=> {beef}	0.02001334	0.8333333	0.02401601	2.258891	30

Figure 26: Rules which include Toilet Paper



	lhs	rhs	support	confidence	coverage	lift	count
[1]	{fruits,hand soap,poultry,sugar}	=> {beef}	0.02134757	0.8000000	0.02668446	2.168535	32
[2]	{beef,coffee/tea,hand soap,sugar}	=> {poultry}	0.02268179	0.8292683	0.02735157	2.027852	34
[3]	{dinner rolls,hand soap,spaghetti sauce,sugar}	=> {poultry}	0.02935290	0.8301887	0.03535690	2.030102	44
[4]	{coffee/tea,dinner rolls,hand soap,spaghetti sauce}	=> {poultry}	0.02468312	0.8222222	0.03002001	2.010622	37
[5]	{dinner rolls,hand soap,soap,spaghetti sauce}	=> {poultry}	0.02134757	0.8000000	0.02668446	1.956281	32
[6]	{fruits,poultry,sugar,toilet paper}	=> {beef}	0.02668446	0.8000000	0.03335557	2.168535	40
[7]	{beef,dinner rolls,spaghetti sauce,sugar}	=> {poultry}	0.02601734	0.8125000	0.03202135	1.986847	39
[8]	{beef,dinner rolls,ice cream,spaghetti sauce}	=> {poultry}	0.02334890	0.8139535	0.02868579	1.990402	35
[9]	{dinner rolls,laundry detergent,spaghetti sauce,sugar}	=> {poultry}	0.02801868	0.8235294	0.03402268	2.013818	42
[10]	{bagels,beef,spaghetti sauce,sugar,toilet paper}	=> {poultry}	0.02001334	0.8823529	0.02268179	2.157662	30
[11]	{bagels,beef,poultry,spaghetti sauce,toilet paper}	=> {sugar}	0.02001334	0.8571429	0.02334890	2.336104	30
[12]	{bagels,poultry,spaghetti sauce,sugar,toilet paper}	=> {beef}	0.02001334	0.8333333	0.02401601	2.258891	30

Figure 27: Rules which include Poultry

9. Other Rules – Beef

With support of 0.02 and confidence of 0.08, and choosing “beef” as “rhs” the obtained rules are 4 [Fig. 28], all with relatively low support. The one with the highest lift is the rule examined in the previous sections.
All contain poultry and sugar, then 3 of them contain fruits and toilet paper.

	lhs		rhs	support	confidence	coverage	lift	count
[1]	{fruits,hand soap,poultry,sugar}	=>	{beef}	0.02134757	0.8000000	0.02668446	2.168535	32
[2]	{fruits,poultry,sugar,toilet paper}	=>	{beef}	0.02668446	0.8000000	0.03335557	2.168535	40
[3]	{fruits,poultry,sugar,toilet paper,vegetables}	=>	{beef}	0.02268179	0.8095238	0.02801868	2.194351	34
[4]	{bagels,poultry,spaghetti sauce,sugar,toilet paper}	=>	{beef}	0.02001334	0.8333333	0.02401601	2.258891	30

Figure 28: Rules with support 0.02 and confidence of 0.08 for users that bought other items and then beef

10. Summary

As per rule 2, a consumer that has previously bought bagels, poultry, spaghetti sauce, sugar, toilet paper has 80% chances to buy beef.

Rule 2: bagels, poultry, spaghetti sauce, sugar, toilet paper > beef

The occurrence of rule 2 is small (support > 0.02) due to the high number of items involved but if it happens there are 80% chances that beef can be sold as well. The rule can be effective online, where a personalized experience is possible. Different promotions on beef can be offered to those users who bought the rest of the items for example.

This rule can be hardly applied in a physical location, as the support is very low. The cost of applying it will be higher than the benefit generated. Applying it on-site will consist of placing the elements nearby.

There seems to be some kind of relationship between poultry, sugar and beef, which would need to be examined in more detail.

In general, the support for the “Groceries” data set is very low but some possible next step, would be to see if there are relationships/dependencies with a smaller amount of items that occur more times. Getting the most benefit from association rules might need some time to play around with the interest measures.

Appendix I - Apriori

Apriori algorithm was developed in 1994 by Agravwal and Srikant (Borgelt, 2017), and is the most popular algorithm to generate Association Rules. There are many algorithms as FP-Gowth (Martínez, 2020), Eclat, Hash-Based, partitioning, etc (Rodrigo, 2018) which in some cases can be more efficient dealing with large data sets than Apriori. Apriori is unsupervised, and for this reason, the data doesn't need to be split into training and validation.

A transaction contains different items. Apriori starts looking for frequency of individual items, then for item sets with 2 items, then 3 items … as seen in [Fig. 28]. If a small itemset is not frequent, will be considered that the item sets that contain this small itemset are not frequent either.

Some measures can be applied in order to restrict the number of the resultant rules as well as to improve efficiency, as seen in "Rules and Interest Measures".

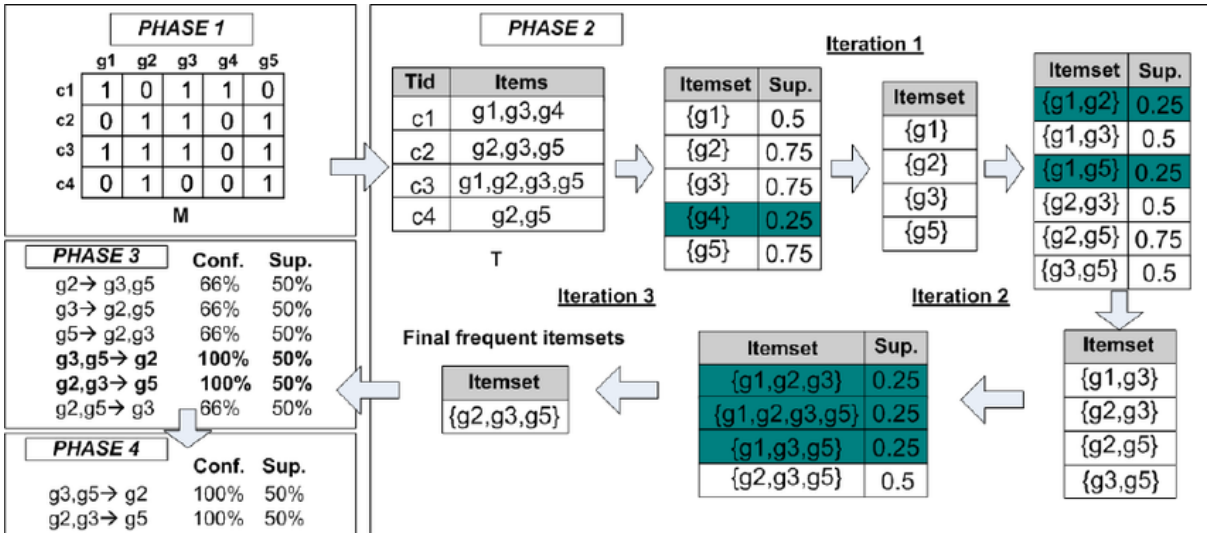


Figure 29: Example of Apriori algorithm with support 2 (Baena, 2009)

Bibliography

Analytics Vidhya, 2017. *Mining frequent items bought together using Apriori Algorithm (with code in R).* [Online]
Available at: https://www.analyticsvidhya.com/blog/2017/08/mining-frequent-items-using-apriori-algorithm/
[Accessed 13 July 2020].

Arnold, B. K., 2018. *Building the "transactions" Class for Association Rule Mining in R using arules and apriori.* [Online]
Available at: https://blog.aptitive.com/building-the-transactions-class-for-association-rule-mining-in-r-using-arules-and-apriori-c6be64268bc4
[Accessed 13 July 2020].

Baena, D. S. R., 2009. *APRIORI algorithm with support set to 2.* [Online]
Available at: https://www.researchgate.net/figure/Example-of-the-A-PRIORI-algorithm-with-support-set-to-2-Therefore-every-itemset-to-be_fig3_26881622
[Accessed 13 July 2020].

Borgelt, C., 2017. *Apriori.* [Online]
Available at: http://www.borgelt.net/doc/apriori/apriori.html
[Accessed 03 July 2020].

Chelluboina, M. H. &. S., n.d. *Visualizing Association Rules: Introduction to the R-extension Package arulesViz.* [Online]
Available at: https://cran.r-project.org/web/packages/arulesViz/vignettes/arulesViz.pdf
[Accessed 12 July 2020].

Dinov, I., 2020. *Data Science and Predictive Analytics (UMich HS650).* [Online]
Available at:
http://www.socr.umich.edu/people/dinov/courses/DSPA_notes/11_Apriory_AssocRuleLearning.html
[Accessed 13 July 2020].

DS 501 - Introduction to Data Science, 2015. *Association Rules.* [Online]
Available at: https://rstudio-pubs-static.s3.amazonaws.com/80871_46067c5667e242d9bfb0dc4d212532dc.html
[Accessed 12 July 2020].

Jabeen, H., 2018. *Market Basket Analysis using R.* [Online]
Available at: https://www.datacamp.com/community/tutorials/market-basket-analysis-r
[Accessed 3 July 2020].

JDATALAB - IMPERFECT, 2018. *A Guide to Association Rules in R - Part 1 The Transactions Class in arules.* [Online]
Available at:
https://www.jdatalab.com/data_science_and_data_mining/2018/10/10/association-rule-transactions-class.html
[Accessed 13 July 2020].

Kadimisetty, A., 2018. *Association Rule Mining in R.* [Online]
Available at: https://towardsdatascience.com/association-rule-mining-in-r-ddf2d044ae50
[Accessed 13 July 2020].

Lantz, B., 2015. *Machine Learning with R.* 2nd Edition ed. Birmingham: PACKT Publishing.

Martínez, C. G., 2020. *REGLAS DE ASOCIACIÓN.* [Online]
Available at: https://rpubs.com/Cristina_Gil/Reglas_Asociacion
[Accessed 11 July 2020].
Michael Hahsler, B. G. K. H. &. C. B., n.d. *Introduction to arules – A computational environment for mining association rules and frequent item sets.* [Online]
Available at: https://cran.r-project.org/web/packages/arules/vignettes/arules.pdf
[Accessed 13 July 2020].
mlxtend, n.d. *Association Rules Generation from Frequent Itemsets.* [Online]
Available at:
http://rasbt.github.io/mlxtend/user_guide/frequent_patterns/association_rules/
[Accessed 13 July 2020].
Prabhakaran, S., n.d. *Association Mining (Market Basket Analysis).* [Online]
Available at: http://r-statistics.co/Association-Mining-With-R.html
[Accessed 13 July 2020].
Raj, R., 2020. *Machine Learning | Lift.* [Online]
Available at: https://www.youtube.com/watch?v=FGGlscdeMOQ
[Accessed 3 July 2020].
Rodrigo, J. A., 2018. *Reglas de asociación y algoritmo Apriori con R.* [Online]
Available at: https://www.cienciadedatos.net/documentos/43_reglas_de_asociacion
[Accessed 29 June 2020].
RStudio, 2018. *What is data wrangling? Intro, Motivation, Outline, Setup -- Pt. 1 Data Wrangling Introduction.* [Online]
Available at: https://www.youtube.com/watch?v=jOd65mR1zfw&list=PL9HYL-VRX0oQOWAFoKHFQAsWAI3ImbNPk
[Accessed 11 July 2020].
StackOverflow, 2016. *R Arules: how to remove certain itemsets from lhs/rhs.* [Online]
Available at: https://stackoverflow.com/questions/41303266/r-arules-how-to-remove-certain-itemsets-from-lhs-rhs
[Accessed 12 July 2020].
Stackoverflow, 2020. *Cleaning Data & Association Rules - R.* [Online]
Available at: https://stackoverflow.com/questions/60232051/cleaning-data-association-rules-r
[Accessed 13 July 2020].
UC Business Analytics R Programming Guide, 2018. *Dealing with Missing Values.* [Online]
Available at: https://uc-r.github.io/missing_values
[Accessed 11 July 2020].
Wikipedia, n.d. *Association rule learning.* [Online]
Available at: https://en.wikipedia.org/wiki/Association_rule_learning
[Accessed 29 June 2020].
Wikipedia, n.d. *Lift (data mining).* [Online]
Available at: https://en.wikipedia.org/wiki/Lift_(data_mining)
[Accessed 12 July 2020].