A1: Retail Analytics Project_Shopping Cart Analysis

Tetsuya Mano

6/11/2021

Overview

This document is my work for an individual assignment in Retail Analytics - DAT-4182 - SFO1. Although there were three options given to a student, I have chosen Shopping Cart Analysis. In this analysis, I used the same data that students used for a group task.

This paper consists of three parts written in the following with source code in R.

1. A brief summary of the dataset you were provided
2. My proposed solution
3. A brief (up to 1 page) overview of the approach/ methodology you have chosen

1. Data Summary

The data is formed with 9,835 transactions and 169 items.

```r
# read transactions dataset and convert to arules sparse matrix format
fcsv <- "https://raw.githubusercontent.com/multidis/hult-retail-analytics/main/shopping_cart/transactio
df_trans <- read_csv(fcsv)
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   .default = col_double()
## )
## i Use `spec()` for the full column specifications.
```

```r
#s <- spec(df_trans)
#cols_condense(s)
trans <- transactions(as.matrix(df_trans))
trans
```

```
## transactions in sparse format with
##  9835 transactions (rows) and
##  169 items (columns)
```

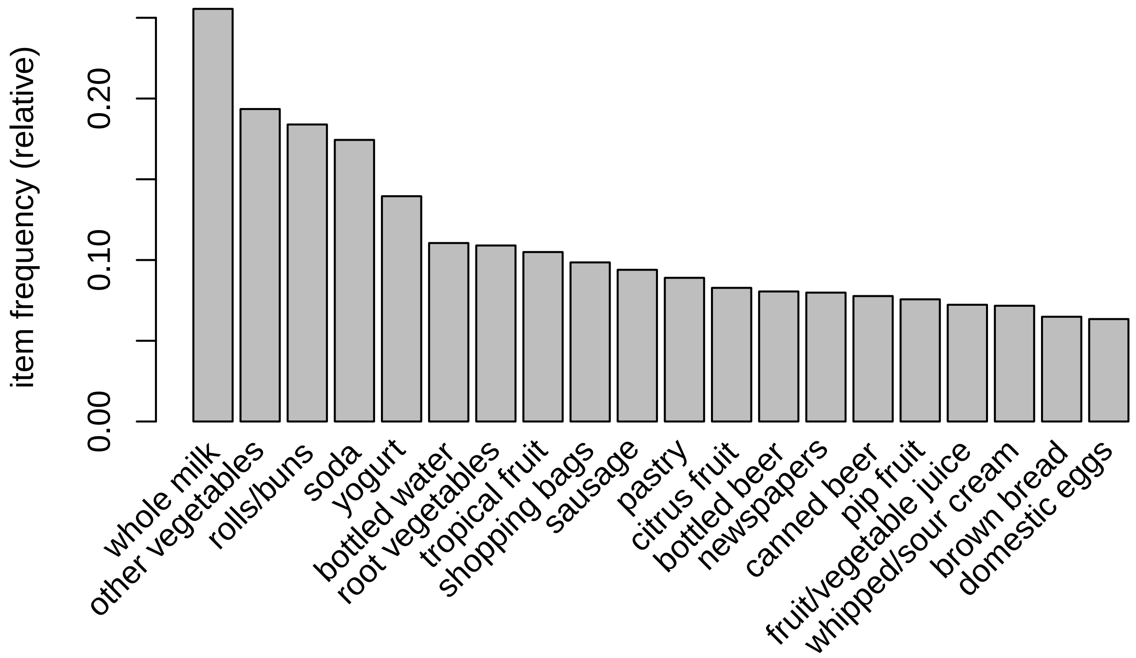
```r
# Identify top-20 most frequently bought products
col_sum <- colSums(df_trans)
View(col_sum)
```

```
top_20 <- col_sum %>%
  sort(decreasing = TRUE) %>%
  head(n=20)

# make dataframe
top_20 <- data.frame(item = names(top_20), frequency = top_20)

# bar plot
##ggplot(data=top_20, aes(x=reorder(item, frequency), y=frequency), fill = sample) + scale_x_discrete(n

# frequency plot for relative form
itemFrequencyPlot(trans, topN = 20)
```



```
# frequency plot for absolute form
##itemFrequencyPlot(trans, topN = 20, type='absolute')
```

Top-4 frequent bought items appear more than 15% among all transactions. The frequency decreases steeply after the four items.Yogurt and bottled water follow the items with 14% and 11% in the frequencies. However, a decrease in the frequency becomes stable after that. Even 20th frequent item maintains more than 5%.

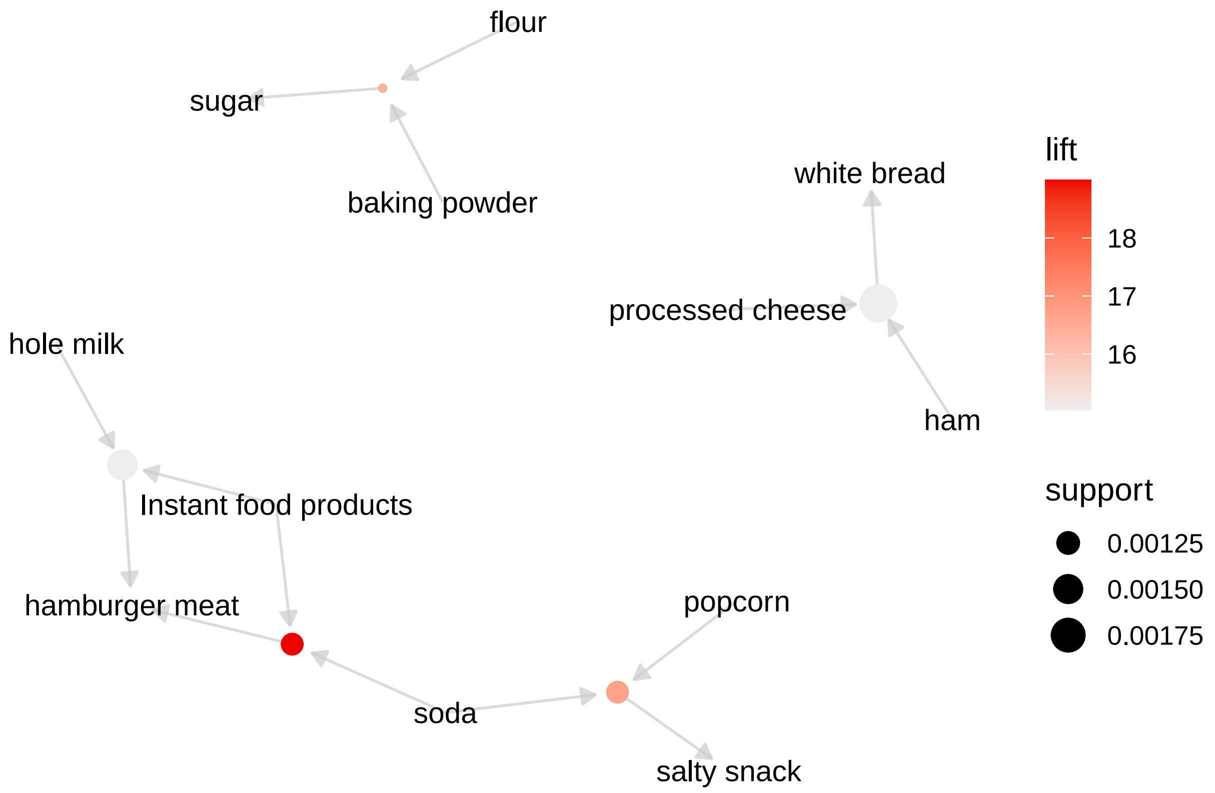
```
# rules exceeding support and confidence thresholds
rules <- apriori(trans, parameter = list(support=0.001, confidence=0.5))
```

```
## Apriori
```

```
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.5    0.1    1 none FALSE            TRUE       5   0.001      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 9
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[169 item(s), 9835 transaction(s)] done [0.01s].
## sorting and recoding items ... [157 item(s)] done [0.00s].
## creating transaction tree ... done [0.01s].
## checking subsets of size 1 2 3 4 5 6 done [0.03s].
## writing ... [5668 rule(s)] done [0.00s].
## creating S4 object  ... done [0.01s].
```

```
# most promising rules by lift (confidence-support scatterplot)
rules_sel <- head(sort(rules, by="lift"), 5)

# network representation
plot(rules_sel, method = "graph",  engine = "ggplot2")
```



The network diagram shows relationships among items.

2.My solution

Since 8th most frequent bought product is tropical fruit, my solution is the following based on the most interesting product association rules in which support is at least 0.001, confidence is above 0.5 (50%), and lift values are as high as possible.

```r
# rules for a selected product
rules_selprod <- subset(rules, subset = rhs %pin% top_20$item[8])
inspect(head(sort(rules_selprod, by="lift"), 5))
```

```
##     lhs                          rhs                  support confidence    coverage     lift count
## [1] {citrus fruit,
##      grapes,
##      fruit/vegetable juice} => {tropical fruit} 0.001118454  0.8461538 0.001321810 8.063879    11
## [2] {ham,
##      pip fruit,
##      other vegetables,
##      yogurt}                => {tropical fruit} 0.001016777  0.8333333 0.001220132 7.941699    10
## [3] {grapes,
##      other vegetables,
##      fruit/vegetable juice} => {tropical fruit} 0.001118454  0.7857143 0.001423488 7.487888    11
## [4] {root vegetables,
##      other vegetables,
##      whole milk,
##      yogurt,
##      bottled water}         => {tropical fruit} 0.001118454  0.7857143 0.001423488 7.487888    11
## [5] {other vegetables,
##      whole milk,
##      butter,
##      yogurt,
##      domestic eggs}         => {tropical fruit} 0.001016777  0.7692308 0.001321810 7.330799    10
```

3.My approach and methodology I chose to reach a solution

I dealt with this task in R. This is because R applies to large-size datasets. There are three steps in reaching my solution.

Step 1. Load required libraries 'arules' package provides us with functions of representing, manipulating, and analyzing transaction data and patterns. 'arulesViz' package is for visualizing Association Rules and Frequent Itemsets. 'RColorBrewer 'is a ColorBrewer Palette that provides color schemes for maps and other graphics.

Step 2. Read the transaction dataset This step is our always task. Use the read_csv function to load the data, then transform the data to make the data easy to handle. As.matrix with the transaction is helpful to apply the following algorithm.

Step 3. Visualize the data, and understand it In order for readers to understand the data, visualization techniques are helpful. Among them, a frequency plot gives us the top n-th frequent bought items. The result enables us to have a basic understanding of the data.

Step 4. Apply apriori algorithm for the target dataset Apriori algorithm is a logic based on Association Rule. There are three different metrics in it: these are (1) Support, (2) Confidence, and (3) Lift. When applying apriori algorithms by using the functions provided in R, the results show up instantly. Before doing this, I did consider a policy applying for the Apriori algorithm to determine what rule is a good rule.

Based on what we learned in this course, the most exciting product association rules are that support is at least 0.001, confidence is above 0.5 (50%), and lift values are as high as possible. I adopted this to reach a solution.