Association Rules Analysis IP 14

Naomi Chebet

9/18/2020

1. Business Understanding

1.1 Define the question:

working for Carrefour Kenya, you've been tasked with creating association rules to identify relationship between variables. Thereafter, provide insights to the marketing department based on your analysis.

1.2 Metric for success

Our project will be successful if we are able to create apriori model with confidence level of at least 80%.

1.3 Experimental Design

Our project will follow following path:

1. Business Understanding
2. Data Understanding
3. Loading and Checking the Data
4. Implementation of the Solution by creating Apriori Model
5. Conclusion
6. Findings and Recommendations

2. Importing Required Libraries

```r
library(arules)
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'arules'
```

```
## The following objects are masked from 'package:base':
##
##     abbreviate, write
```

3. Loading and Checking the Data

```r
path <- "http://bit.ly/SupermarketDatasetII"
transactions <- read.transactions(path, sep = ",")
```

```
## Warning in asMethod(object): removing duplicated items in transactions
```

```
transactions
```

```
## transactions in sparse format with
##  7501 transactions (rows) and
##  119 items (columns)
```

Our dataset has 7501 rows and 119 columns.

```
# checking the class of our transaction dataset

class(transactions)
```

```
## [1] "transactions"
## attr(,"package")
## [1] "arules"
```

The data is in the right format.

```
# previewing the first five transactions

inspect(transactions[1:5])
```

```
##     items
## [1] {almonds,
##      antioxydant juice,
##      avocado,
##      cottage cheese,
##      energy drink,
##      frozen smoothie,
##      green grapes,
##      green tea,
##      honey,
##      low fat yogurt,
##      mineral water,
##      olive oil,
##      salad,
##      salmon,
##      shrimp,
##      spinach,
##      tomato juice,
##      vegetables mix,
##      whole weat flour,
##      yams}
## [2] {burgers,
##      eggs,
##      meatballs}
## [3] {chutney}
## [4] {avocado,
##      turkey}
## [5] {energy bar,
##      green tea,
##      milk,
##      mineral water,
##      whole wheat rice}
```

```r
# generating a summary of the transaction dataset
summary(transactions)
```

```
## transactions as itemMatrix in sparse format with
##  7501 rows (elements/itemsets/transactions) and
##  119 columns (items) and a density of 0.03288973
##
## most frequent items:
## mineral water          eggs     spaghetti  french fries     chocolate
##          1788          1348          1306          1282          1229
##       (Other)
##         22405
##
## element (itemset/transaction) length distribution:
## sizes
##    1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16
## 1754 1358 1044  816  667  493  391  324  259  139  102   67   40   22   17    4
##   18   19   20
##    1    2    1
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   2.000   3.000   3.914   5.000  20.000
##
## includes extended item information - examples:
##             labels
## 1          almonds
## 2 antioxydant juice
## 3        asparagus
```

The summary generates a list of the most frequent items which include: mineral water, eggs, spaghetti, french fries, and chocolate. It also shows the total number of items purchased in each transaction (length distribution).

```r
# exploring the frequency of some items. For example,
# transactions ranging from 8 to 10 and
# performing some operation in percentage terms of the total transactions.

itemFrequency(transactions[, 8:10], type = "absolute")
```

```
##   black tea blueberries  body spray
##         107          69          86
```

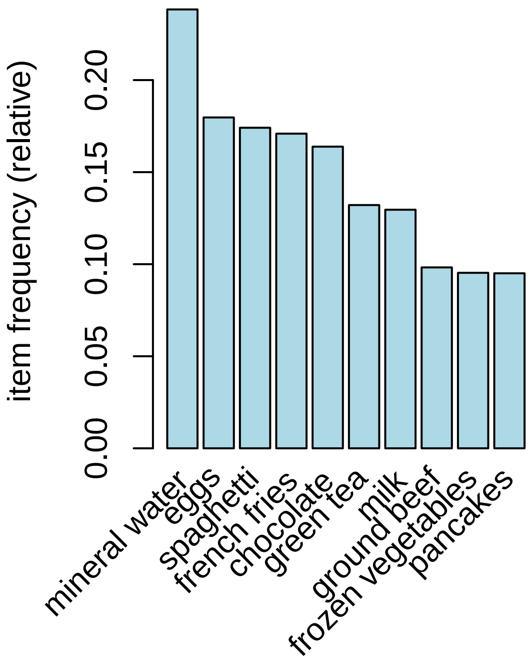
```r
round(itemFrequency(transactions[,8:10], type = "relative") *100,2)
```

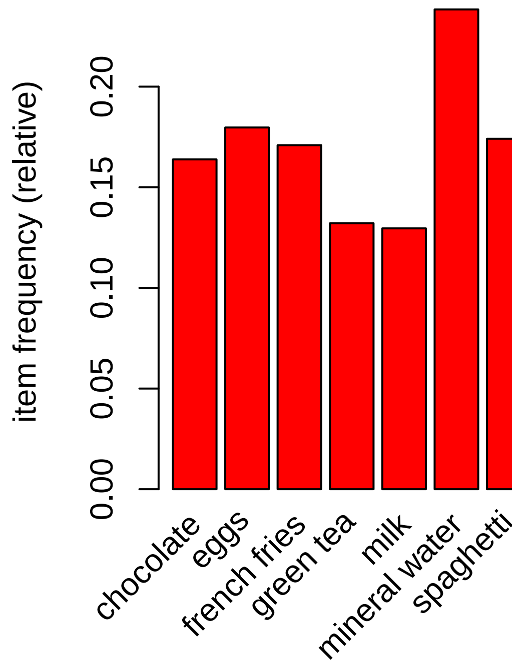
```
##   black tea blueberries  body spray
##        1.43        0.92        1.15
```

We see black tea was the most popular which amounts to 1.43 percent of total number of items purchased.

```r
# plotting a chart of frequency of items and filtering to consider only items
# with a minimum percentage of support considering a top x of items.
# here we plot top 10 most common items whose relative importance is at least 10%

par(mfrow = c(1,2))
itemFrequencyPlot(transactions, topN = 10, col = "light blue")
itemFrequencyPlot(transactions, support = 0.1, col = "red")
```





From the first graph we have a plot of top ten most common items. In plot two we have a plot of top items from plot 1 with support of at least 10%.

The items with support of at least 10% include: chocolate, eggs, french fries, green tea, milk, mineral water and spaghetti.

4. Implementing the Solution

```r
# next, we move on to build a model based on association rules using the apriori function.
# Apriori through Apriori Property assumes that all subsets of a frequent itemset much be
# frequent, this helps to improve the efficiency of level-wise generation of frequent items.
# If an itemset is infrequent, all it's supersets will be infrequent and dropped from
# evaluation.

# we will use min support of 0.001 and confidence of 0.8

association.rules <- apriori(transactions, parameter = list(supp = 0.001, conf = 0.8))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.8    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: 7
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[119 item(s), 7501 transaction(s)] done [0.00s].
## sorting and recoding items ... [116 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 3 4 5 6 done [0.00s].
## writing ... [74 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```
association.rules
```

```
## set of 74 rules
```

After building the model with 0.001 min support and confidence of 0.8, we obtain a set of 74 rules. Which is pretty good given our dataset, but how sensitive is the model? Let's vary min support and confidence and see how that compares.

```
# to test the sensitivity of the model, we will change the min support to 0.002
rules.002 <- apriori(transactions, parameter = list(supp = 0.002, conf = 0.8))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.8    0.1    1 none FALSE            TRUE       5   0.002      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: 15
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[119 item(s), 7501 transaction(s)] done [0.00s].
## sorting and recoding items ... [115 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 3 4 5 done [0.00s].
## writing ... [2 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```
rules.002
```

```
## set of 2 rules
```

```
# then compare with changing the confidence level to 0.6

rules.6 <- apriori(transactions, parameter = list(supp = 0.001, conf = 0.6))
```

```
## Apriori
##
## Parameter specification:
```

```
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.6    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: 7
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[119 item(s), 7501 transaction(s)] done [0.00s].
## sorting and recoding items ... [116 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 3 4 5 6 done [0.00s].
## writing ... [545 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```r
rules.6
```

```
## set of 545 rules
```

When we change the min support to 0.002, the set of rules drops to 2 which is a big drop and it means that we won't be able to obtain interesting rules. When confidence level is reduced to 0.6, the set of rules increases drastically to 545 which might be a bit too much and not very useful. From the test, we conclude that a min support of 0.001 and confidence level of 0.8 is optimal to use for this problem.

```r
# checking the summary of our optimal model

summary(association.rules)
```

```
## set of 74 rules
##
## rule length distribution (lhs + rhs):sizes
##  3  4  5  6
## 15 42 16  1
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   3.000   4.000   4.000   4.041   4.000   6.000
##
## summary of quality measures:
##     support           confidence        coverage             lift
##  Min.   :0.001067   Min.   :0.8000   Min.   :0.001067   Min.   : 3.356
##  1st Qu.:0.001067   1st Qu.:0.8000   1st Qu.:0.001333   1st Qu.: 3.432
##  Median :0.001133   Median :0.8333   Median :0.001333   Median : 3.795
##  Mean   :0.001256   Mean   :0.8504   Mean   :0.001479   Mean   : 4.823
##  3rd Qu.:0.001333   3rd Qu.:0.8889   3rd Qu.:0.001600   3rd Qu.: 4.877
##  Max.   :0.002533   Max.   :1.0000   Max.   :0.002666   Max.   :12.722
##      count
##  Min.   : 8.000
##  1st Qu.: 8.000
##  Median : 8.500
##  Mean   : 9.419
##  3rd Qu.:10.000
##  Max.   :19.000
```

```
##
## mining info:
##          data ntransactions support confidence
##  transactions          7501   0.001        0.8
```

5. Conclusion

From the above analysis we have successfully created apriori model with rules to identify the relationships among items in our dataset.

1. From earlier analysis we saw that the most frequent items in our dataset include: mineral water, eggs, spaghetti, french fries, and chocolate.
2. The items with support of at least 10% include: chocolate, eggs, french fries, green tea, milk, mineral water and spaghetti.
3. The optimal apriori model for this problem, based on our findings, has a min support of 0.001 and confidence of 0.8.

Moving forward, what insights can we draw from apriori model? What recommendations could we give to the marketing team? We will explore on insights and recommendations below.

6. Findings and Recommendations

```r
# What are the first 10 model rules and what do they tell us?
# association rules ordered by confidence.
association.rules <- sort(association.rules, by = "confidence", decreasing = TRUE)
inspect(association.rules[1:10])
```

```
##      lhs                        rhs                 support confidence    coverage      lift count
## [1]  {french fries,
##       mushroom cream sauce,
##       pasta}                 => {escalope}      0.001066524  1.0000000 0.001066524 12.606723     8
## [2]  {ground beef,
##       light cream,
##       olive oil}             => {mineral water} 0.001199840  1.0000000 0.001199840  4.195190     9
## [3]  {cake,
##       meatballs,
##       mineral water}         => {milk}          0.001066524  1.0000000 0.001066524  7.717078     8
## [4]  {cake,
##       olive oil,
##       shrimp}                => {mineral water} 0.001199840  1.0000000 0.001199840  4.195190     9
## [5]  {mushroom cream sauce,
##       pasta}                 => {escalope}      0.002532996  0.9500000 0.002666311 11.976387    19
## [6]  {red wine,
##       soup}                  => {mineral water} 0.001866418  0.9333333 0.001999733  3.915511    14
## [7]  {eggs,
##       mineral water,
##       pasta}                 => {shrimp}        0.001333156  0.9090909 0.001466471 12.722185    10
## [8]  {herb & pepper,
##       mineral water,
##       rice}                  => {ground beef}   0.001333156  0.9090909 0.001466471  9.252498    10
## [9]  {ground beef,
##       pancakes,
##       whole wheat rice}      => {mineral water} 0.001333156  0.9090909 0.001466471  3.813809    10
```

```
## [10] {frozen vegetables,
##       milk,
##       spaghetti,
##       turkey}              => {mineral water} 0.001199840  0.9000000 0.001333156  3.775671     9
```

FINDINGS:

The first four rules have 100 confidence level. This means that if someone buys the items in Ihs column (first column), they are 100% likely to buy the items in rhs column (second column).

As for rule 5 to 10:

1. If someone buys mushroom cream sauce and pasta, they are 95% likely to buy escalope too.
2. If someone buys red wine and soup, they are 93% likely to buy mineral water as well.
3. If someone buys eggs,mineral water and pasta, they are 90.9% likely to buy shrimp as well.
4. If someone buys herb & pepper,mineral water and rice, they are 90.9% likely to buy ground beef too.
5. If someone buys ground beef,pancakes and whole wheat rice, they are 90.9% likely to buy mineral water too.
6. If someone buys frozen vegetables, milk, spaghetti and turkey, they are 90% likely to buy mineral water as well.

RECOMMENDATION: based on these findings, we recommend that the marketing team organize these items such that they are easy to find. Ideally, the items be located in the same area/section to maximize on the customer's purchase, which will in turn increase the supermarket's number of sales.

```r
# How do you make a promotion relating to a particular product?
# For example if we wanted to do a promotion around milk,
# we can find out what the customers bought before purchasing milk as shown below:

#selecting the items that were bought with milk
milk <- subset(association.rules, subset = rhs %pin% "milk")

#sorting in descending order by confidence.
milk <- sort(milk, by = "confidence", decreasing = TRUE)

#checking the first 5 records
inspect(milk[1:5])
```

```
##     lhs                                    rhs    support     confidence
## [1] {cake,meatballs,mineral water}    => {milk} 0.001066524 1.0000000
## [2] {escalope,hot dogs,mineral water} => {milk} 0.001066524 0.8888889
## [3] {meatballs,whole wheat pasta}     => {milk} 0.001333156 0.8333333
## [4] {black tea,frozen smoothie}       => {milk} 0.001199840 0.8181818
## [5] {burgers,ground beef,olive oil}   => {milk} 0.001066524 0.8000000
##     coverage    lift     count
## [1] 0.001066524 7.717078  8
## [2] 0.001199840 6.859625  8
## [3] 0.001599787 6.430898 10
## [4] 0.001466471 6.313973  9
## [5] 0.001333156 6.173663  8
```

FINDINGS: the records show items that were bought together with milk at different confidence level of 80% - 100%. Generally the above method would allow the marketing team to find out all the rhs products and what accompanied it's purchase.

RECOMMENDATION: from the above finding we get good insight in understanding which items (rhs items) are bought with what accompaniments (lhs). The marketing team can leverage this knowledge in creating promotions around such items. A small promotion around these items could be organized at the end of the month when people have earned their monthly salaries, this could heavily increase the number of sales for the supermarket.