ASSOCIATION_ANALYSIS

GRIFFIN BURET

2022-08-03

##RESEARCH QUESTION##

Carrefour Kenya and are currently undertaking a project that will inform the marketing department on the most relevant marketing strategies that will result in the highest no. of sales (total price including tax).This project is aimed at doing analysis on the dataset provided by carrefour and create insights on how to achieve highest sales.

##METRIC FOR SUCCESS##

Be able to come up with an association analysis for the products

##THE CONTEXT##

Carre Four is an International chain of retail supemarkets in the world, It was set up in Kenya in the year 2016 and has been performing well over the years.Carrefour ensures customer satisfaction and everyday convenience while offering unbeatable value for money with a vast array of more than 100,000 products, shoppers can purchase items for their every need, whether home electronics or fresh fruits from around the world, to locally produced items. This project is aimed at creating insights from existing and current trends to develop marketing strategies that will enable the marketing team achieve higher sales.

##EXPERIMENTAL DESIGN##

1. Loading libraries
2. Load data
3. Association analysis
4. Conclusion
5. Recommendation

loading libraries

```
#install.packages("arules")
library("arules")

## Loading required package: Matrix

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

loading dataset

```
#will use read.transactions fuction which will load data from comma-sep
arated files
# and convert them to the class transactions, which is the kind of data
 that
# we will require while working with models of association rules
Transactions<-read.transactions("C://moringa//GROUP WORK//Supermarket_S
ales_Dataset II.csv",sep=",")

## Warning in asMethod(object): removing duplicated items in transactio
ns

Transactions

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

previewing dataset

```
#looking at the items that make our data
items<-as.data.frame(itemLabels(Transactions))
colnames(items) <- "Item"
head(items, 10)

##                  Item
## 1             almonds
## 2   antioxydant juice
## 3           asparagus
## 4             avocado
## 5         babies food
## 6               bacon
## 7      barbecue sauce
## 8           black tea
## 9         blueberries
## 10         body spray
```

EDA

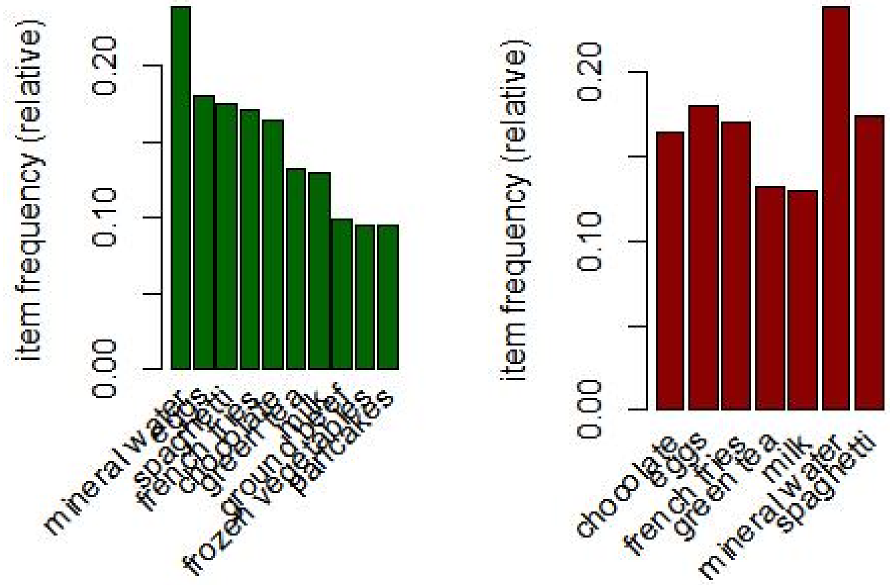
```
#generating summary that will give us info eg the most purchased
summary(Transactions)

## 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     chocolat
e
##          1788          1348          1306          1282          122
9
##       (Other)
##         22405
##
## element (itemset/transaction) length distribution:
## sizes
##    1    2    3    4    5    6    7    8    9   10   11   12   13   1
4   15   16
## 1754 1358 1044  816  667  493  391  324  259  139  102   67   40    2
2   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
```

we can see that the most purchased is mineral water followed by eggs

```
par(mfrow = c(1, 2))
# plot the frequency of items
itemFrequencyPlot(Transactions, topN = 10,col="darkgreen")
itemFrequencyPlot(Transactions, support = 0.1,col="darkred")
```



```
# Building a model based on association rules
# using the apriori function
# ---
# We use Min Support as 0.001 and confidence as 0.8
# ---
#
rules <- apriori (Transactions, parameter = list(supp = 0.001, conf = 0.
8))

## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support m
inlen
##         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 [0.01s].
## writing ... [74 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].

rules

## set of 74 rules

#we build the model  using 0.001 Min support
# and confidence as 0.8 we obtained 74 rules.
# to investigate the sensitivity of the parameters of the mode;
# we will see what happens if we increase the support or lower the conf
idence levee builtl
#
# Building a apriori model with Min Support as 0.002 and confidence as
0.8.
rules2 <- apriori (Transactions,parameter = list(supp = 0.002, conf = 0.
8))

## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support m
inlen
##         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 [0.01s].
## sorting and recoding items ... [115 item(s)] done [0.00s].
## creating transaction tree ... done [0.01s].
## checking subsets of size 1 2 3 4 5 done [0.01s].
## writing ... [2 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].

# Building apriori model with Min Support as 0.002 and confidence as 0.
6.
rules3 <- apriori (Transactions, parameter = list(supp = 0.001, conf =
0.6))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support m
inlen
##         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 [0.01s].
## writing ... [545 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```
rules2
```

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

```
rules3
```

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

we first increased the minimum support of 0.001 to 0.002 and model rules went from 74 to only 2. This would lead us to understand that using a high level of support can make the model lose interesting rules. we then decreased the minimum confidence level to 0.6 and the number of model rules went from 74 to 545. This would mean that using a low confidence level increases the number of rules to quite an extent and many will not be useful

```
summary(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.   : 3.3
56
##  1st Qu.:0.001067   1st Qu.:0.8000   1st Qu.:0.001333   1st Qu.: 3.4
32
##  Median :0.001133   Median :0.8333   Median :0.001333   Median : 3.7
95
##  Mean   :0.001256   Mean   :0.8504   Mean   :0.001479   Mean   : 4.8
23
##  3rd Qu.:0.001333   3rd Qu.:0.8889   3rd Qu.:0.001600   3rd Qu.: 4.8
77
##  Max.   :0.002533   Max.   :1.0000   Max.   :0.002666   Max.   :12.7
22
##       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
##
 call
##  apriori(data = Transactions, parameter = list(supp = 0.001, conf =
0.8))

#, the function would give us information about the model :
# A.i.e. the size of rules, b.depending on the items that contain these
 rules.
# In our above case, most rules have 3 and 4 items though some rules do
 have upto 6.
# More statistical information such as support, lift and confidence is
also provided.
# ---

# Observing rules built in our model i.e. first 5 model rules
# ---
#
inspect(rules[1:5])

##     lhs                              rhs             support     con
fidence
## [1] {frozen smoothie, spinach}    => {mineral water} 0.001066524 0.8
888889
## [2] {bacon, pancakes}             => {spaghetti}     0.001733102 0.8
125000
```

```
## [3] {nonfat milk, turkey}         => {mineral water} 0.001199840 0.8
181818
## [4] {ground beef, nonfat milk}    => {mineral water} 0.001599787 0.8
571429
## [5] {mushroom cream sauce, pasta} => {escalope}      0.002532996 0.9
500000
##     coverage    lift      count
## [1] 0.001199840  3.729058  8
## [2] 0.002133049  4.666587 13
## [3] 0.001466471  3.432428  9
## [4] 0.001866418  3.595877 12
## [5] 0.002666311 11.976387 19
```

we can see that if a person frozen smoothie and spinach then there is 88% chance he will buy mineral water

```
# we order this rules by confidence though
# We can also use different criteria such as: (by = "lift" or by = "sup
port")
#
rules<-sort(rules, by="confidence", decreasing=TRUE)
inspect(rules[1:5])
```

```
##     lhs                        rhs                 support confidenc
e    coverage      lift count
## [1] {french fries,

##      mushroom cream sauce,

##      pasta}                 => {escalope}      0.001066524       1.0
0 0.001066524 12.606723     8
## [2] {ground beef,

##      light cream,

##      olive oil}             => {mineral water} 0.001199840       1.0
0 0.001199840  4.195190     9
## [3] {cake,

##      meatballs,

##      mineral water}         => {milk}          0.001066524       1.0
0 0.001066524  7.717078     8
## [4] {cake,

##      olive oil,

##      shrimp}                => {mineral water} 0.001199840       1.0
0 0.001199840  4.195190     9
## [5] {mushroom cream sauce,
##      pasta}                    => {escalope}      0.002532996       0.9
5 0.002666311 11.976387    19
```

the first four rules has 100% confidence

```
# the organisation has decided to make a promotion for chocolate thus w
e make a subset rule containg the product
# This would tell us the items that the customers bought before purchas
ing yogurt
# ---
#
chocolate <- subset(rules, subset = rhs %pin% "chocolate")

# Then order by confidence
chocolate<-sort(chocolate, by="confidence", decreasing=TRUE)
inspect(chocolate[1:2])
```

```
##     lhs                                rhs          support     conf
idence
## [1] {escalope, french fries, shrimp} => {chocolate} 0.001066524 0.88
88889
## [2] {red wine, tomato sauce}         => {chocolate} 0.001066524 0.80
00000
##     coverage    lift     count
## [1] 0.001199840 5.425188 8
## [2] 0.001333156 4.882669 8
```

we see that a person will buy escalope,french fries and shrimp has 88% chance of buying chocolate

```
# we are intrested by the items a person will buy after buying chocolat
e
#
chocolate <- subset(rules, subset = lhs %pin% "chocolate")

# Then order by confidence
chocolate<-sort(chocolate, by="confidence", decreasing=TRUE)
inspect(chocolate[1:2])
```

```
##     lhs                     rhs                 support confidence
  coverage     lift count
## [1] {chocolate,

##      frozen vegetables,

##      olive oil,

##      shrimp}             => {mineral water} 0.001199840  0.9000000 0.
001333156 3.775671     9
```

```
## [2] {chocolate,

##      soup,

##      turkey}             => {mineral water} 0.001066524  0.8888889 0.
001199840 3.729058     8
```

we see that a person who buys chocolate,soup ad turkey will 89% chance buy mineral water

Recomendations

The organisation should be using the association analysis in promoting it products

Conclusion

use of association analysis leads to increase sell as this will help them know what people purchase prior and after buying the product