Part3 Association Rules

```
install.packages('arules')
```

```
## Installing package into '/home/greg/R/x86_64-pc-linux-gnu-library/3.6'
## (as 'lib' is unspecified)
```

```
library(arules)
```

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

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

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

```
df <- read.transactions('http://bit.ly/SupermarketDatasetII')
```

I'll load the dataset

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

```
df
```

```
## transactions in sparse format with
##  7501 transactions (rows) and
##  5729 items (columns)
```

```
class(df)
```

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

```
summary(df)
```

```
## transactions as itemMatrix in sparse format with
##  7501 rows (elements/itemsets/transactions) and
##  5729 columns (items) and a density of 0.0005421748
##
## most frequent items:
##     tea   wheat mineral     fat  yogurt (Other)
##     803     645     577     574     543   20157
##
## element (itemset/transaction) length distribution:
## sizes
##    1    2    3    4    5    6    7    8    9   10   11   12   13   15   16
## 1603 2007 1382  942  651  407  228  151   70   39   13    5    1    1    1
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   2.000   3.000   3.106   4.000  16.000
##
## includes extended item information - examples:
##                    labels
## 1                       &
## 2             accessories
## 3 accessories,antioxydant
```

```
# Previewing our first 5 rows
#
inspect(df[1:5])
```

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

Association Rules

```
items = as.data.frame(itemLabels(df))
colnames(items) <- "Item"
head(items, 5)
```

```
##                            Item
## 1                             &
## 2                   accessories
## 3       accessories,antioxydant
## 4   accessories,champagne,fresh
## 5 accessories,champagne,protein
```

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

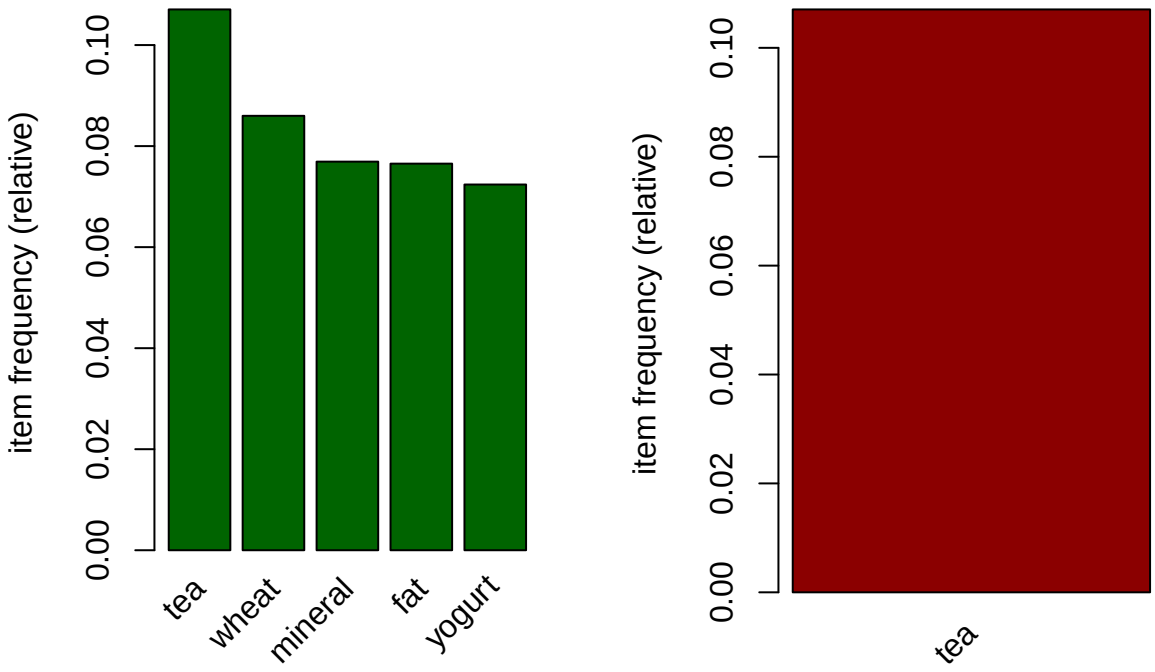
I'll then explore the frequency of the articles

```
##     accessories,chocolate,frozen        accessories,chocolate,low
##                                1                                1
## accessories,chocolate,pasta,salt
##                                1
```

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

```
##     accessories,chocolate,frozen        accessories,chocolate,low
##                             0.01                             0.01
## accessories,chocolate,pasta,salt
##                             0.01
```

```
par(mfrow = c(1, 2))
# plot the frequency of items
itemFrequencyPlot(df, topN = 5,col="darkgreen")
itemFrequencyPlot(df, support = 0.1,col="darkred")
```




```
# Building a model based on association
greg = apriori (df, 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 ...[5729 item(s), 7501 transaction(s)] done [0.01s].
## sorting and recoding items ... [354 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 3 4 done [0.00s].
## writing ... [271 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```
greg
```

```
## set of 271 rules
```

```
greg = sort(greg, by="lift", decreasing=TRUE)
inspect(greg[1:10])
```

I'll then order the rules by level of confidence

```
##      lhs                  rhs            support     confidence coverage
## [1]  {&,fresh}         => {tuna,herb}    0.001199840 0.90       0.001333156
## [2]  {parmesan,wheat} => {cheese,whole} 0.001333156 1.00       0.001333156
## [3]  {fat,tea}         => {yogurt,green} 0.004666045 1.00       0.004666045
## [4]  {&,grated}        => {cheese,herb}  0.004666045 1.00       0.004666045
## [5]  {bar,hand}        => {protein}      0.001199840 1.00       0.001199840
## [6]  {flour,green}     => {weat}         0.001199840 1.00       0.001199840
## [7]  {flour}           => {weat}         0.001466471 1.00       0.001466471
## [8]  {flour,french}    => {weat}         0.002133049 1.00       0.002133049
## [9]  {candy}           => {bars}         0.003066258 0.92       0.003332889
## [10] {extra}           => {dark}         0.001066524 1.00       0.001066524
##      lift      count
## [1]  613.71818  9
## [2]  258.65517 10
## [3]  197.39474 35
## [4]  153.08163 35
## [5]  144.25000  9
## [6]  107.15714  9
## [7]  107.15714 11
## [8]  107.15714 16
## [9]  100.01333 23
## [10]  83.34444  8
```

Conclusion

1. Tea and wheat were the most frequent items bought.