NEW

Emmanuel Titi

2022-06-14

```
mrt<- read.csv("http://bit.ly/SupermarketDatasetII")
head(mrt)
```

```
##                 shrimp      almonds    avocado   vegetables.mix green.grapes
## 1             burgers    meatballs       eggs
## 2             chutney
## 3              turkey      avocado
## 4       mineral water         milk energy bar whole wheat rice     green tea
## 5      low fat yogurt
## 6  whole wheat pasta french fries
##    whole.weat.flour yams cottage.cheese energy.drink tomato.juice low.fat.yogurt
## 1
## 2
## 3
## 4
## 5
## 6
##    green.tea honey salad mineral.water salmon antioxydant.juice frozen.smoothie
## 1
## 2
## 3
## 4
## 5
## 6
##    spinach olive.oil
## 1            NA
## 2            NA
## 3            NA
## 4            NA
## 5            NA
## 6            NA
```

We first load the data into an object of transaction class

```
library(arules)
```

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

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

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

```
tfrm<-as(mrt,"transactions")
```

```
## Warning: Column(s) 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16,
## 17, 18, 19 not logical or factor. Applying default discretization (see '?
## discretizeDF').
```

```
summary(tfrm)
```

```
## transactions as itemMatrix in sparse format with
##  7500 rows (elements/itemsets/transactions) and
##  1280 columns (items) and a density of 0.01484375
##
## most frequent items:
##          spinach= antioxydant.juice=   frozen.smoothie=        salmon=
##             7498              7497              7497           7493
##      mineral.water=          (Other)
##             7476            105039
##
## element (itemset/transaction) length distribution:
## sizes
##    19
## 7500
##
##     Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       19      19      19      19      19      19
##
## includes extended item information - examples:
##                     labels variables          levels
## ## 1          shrimp=almonds     shrimp          almonds
## ## 2 shrimp=antioxydant juice     shrimp antioxydant juice
## ## 3        shrimp=asparagus     shrimp         asparagus
##
## includes extended transaction information - examples:
##    transactionID
## ## 1             1
## ## 2             2
## ## 3             3
```

The summary gives us information on the shape of the data and the density tells us the percentage of non-zero cells .

Lets display the relative item frequency:

```
itemFrequencyPlot(tfrm, topN=20,cex.names=1)
```
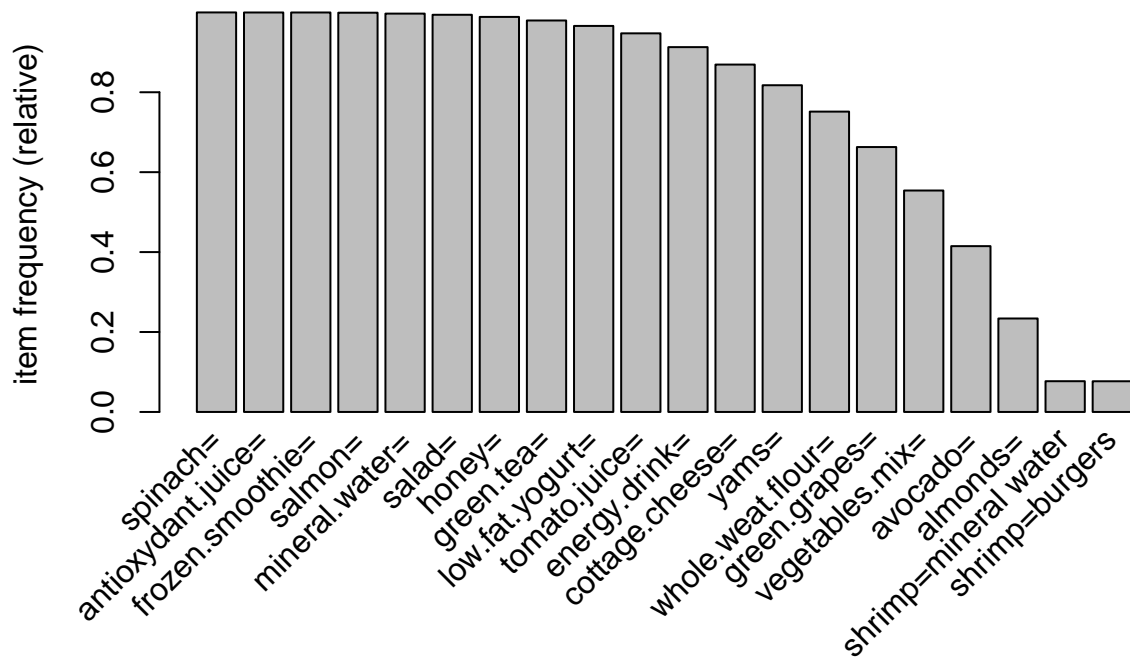
First five items have the same relative item frequency of 100% ,this can also be known as support.

Apriori Rules

```
#min support 0.2, confidence as 0.5
rules<-apriori(tfrm,parameter = list(supp=0.3,conf=0.5,
                                     maxlen=1,
                                     target="rules"))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.5    0.1    1 none FALSE            TRUE       5     0.3      1
##  maxlen target  ext
##       1  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 2250
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[1280 item(s), 7500 transaction(s)] done [0.09s].
## sorting and recoding items ... [17 item(s)] done [0.00s].
```

```
## creating transaction tree ... done [0.00s].
## checking subsets of size 1


## Warning in apriori(tfrm, parameter = list(supp = 0.3, conf = 0.5, maxlen = 1, :
## Mining stopped (maxlen reached). Only patterns up to a length of 1 returned!


##  done [0.00s].
## writing ... [16 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

summary(rules)

```
## set of 16 rules
##
## rule length distribution (lhs + rhs):sizes
##  1
## 16
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       1       1       1       1       1       1
##
## summary of quality measures:
##     support          confidence         coverage       lift         count
##  Min.   :0.5541   Min.   :0.5541   Min.   :1   Min.   :1   Min.   :4156
##  1st Qu.:0.8564   1st Qu.:0.8564   1st Qu.:1   1st Qu.:1   1st Qu.:6423
##  Median :0.9728   Median :0.9728   Median :1   Median :1   Median :7296
##  Mean   :0.9024   Mean   :0.9024   Mean   :1   Mean   :1   Mean   :6768
##  3rd Qu.:0.9974   3rd Qu.:0.9974   3rd Qu.:1   3rd Qu.:1   3rd Qu.:7480
##  Max.   :0.9997   Max.   :0.9997   Max.   :1   Max.   :1   Max.   :7498
##
## mining info:
##  data ntransactions support confidence
##  tfrm          7500     0.3        0.5
##                                                                              call
##  apriori(data = tfrm, parameter = list(supp = 0.3, conf = 0.5, maxlen = 1, target = "rules"))
```

inspect(rules)

```
##       lhs    rhs                 support   confidence coverage lift count
## [1]  {}  => {vegetables.mix=}   0.5541333 0.5541333  1        1    4156
## [2]  {}  => {green.grapes=}     0.6629333 0.6629333  1        1    4972
## [3]  {}  => {whole.weat.flour=} 0.7516000 0.7516000  1        1    5637
## [4]  {}  => {yams=}             0.8176000 0.8176000  1        1    6132
## [5]  {}  => {cottage.cheese=}   0.8693333 0.8693333  1        1    6520
## [6]  {}  => {energy.drink=}     0.9129333 0.9129333  1        1    6847
## [7]  {}  => {tomato.juice=}     0.9474667 0.9474667  1        1    7106
## [8]  {}  => {low.fat.yogurt=}   0.9660000 0.9660000  1        1    7245
## [9]  {}  => {green.tea=}        0.9796000 0.9796000  1        1    7347
## [10] {}  => {honey=}            0.9885333 0.9885333  1        1    7414
## [11] {}  => {salad=}            0.9938667 0.9938667  1        1    7454
## [12] {}  => {mineral.water=}    0.9968000 0.9968000  1        1    7476
## [13] {}  => {salmon=}           0.9990667 0.9990667  1        1    7493
```

4

```
## [14] {}  => {antioxydant.juice=} 0.9996000 0.9996000  1         1    7497
## [15] {}  => {frozen.smoothie=}   0.9996000 0.9996000  1         1    7497
## [16] {}  => {spinach=}           0.9997333 0.9997333  1         1    7498
```

the rules above have empty LHS meaning thet no mater what other items are involved, the item in the RHS
will appear with the probability given in the confidence which equals the support.