Grocery Analysis

Harshad Kumar Elangovan

26/12/2020

Groceries Dataset from ‘arules’ package

```
library(arules)
```

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

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

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

```
data("Groceries")

#?Groceries

class(Groceries)
```

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

```
print(Groceries)
```

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

```
inspect(Groceries[1:3])
```

```
##     items
## [1] {citrus fruit,
##      semi-finished bread,
##      margarine,
##      ready soups}
## [2] {tropical fruit,
##      yogurt,
##      coffee}
## [3] {whole milk}
```

We will set the support and confidence for the provided dataset.

```
rules <-apriori(Groceries,parameter =list(support =0.01,confidence =0.5))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.5    0.1    1 none FALSE            TRUE       5    0.01      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: 98
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[169 item(s), 9835 transaction(s)] done [0.00s].
## sorting and recoding items ... [88 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 3 4 done [0.00s].
## writing ... [15 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```
# extract quality measures
qual <-quality(rules)
```

We will compute the probabilities and standardized lift

```
# compute p(A) and p(B)
pA <-qual$support/qual$confidence
pB <-qual$confidence/qual$lift

# compute lift upper and lower bounds
U <-apply(cbind(1/pA,1/pB),1, min)
L <-apply(cbind(1/pA+1/pB-1/(pA*pB),0.01/(pA*pB),0.5/pB,0),1, max)

sLift <-(qual$lift-L)/(U-L)# standardized lift
data.frame(rule =labels(rules), sLift)# print rules and sLift
```

```
##                                                  rule       sLift
## 1                           {curd,yogurt} => {whole milk} 0.009071877
## 2               {other vegetables,butter} => {whole milk} 0.147208122
## 3         {other vegetables,domestic eggs} => {whole milk} 0.105022831
## 4              {yogurt,whipped/sour cream} => {whole milk} 0.049019608
## 5   {other vegetables,whipped/sour cream} => {whole milk} 0.014084507
## 6             {pip fruit,other vegetables} => {whole milk} 0.035019455
## 7    {citrus fruit,root vegetables} => {other vegetables} 0.048248513
## 8  {tropical fruit,root vegetables} => {other vegetables} 0.169082126
## 9         {tropical fruit,root vegetables} => {whole milk} 0.140096618
## 10                {tropical fruit,yogurt} => {whole milk} 0.034722222
## 11          {root vegetables,yogurt} => {other vegetables} 0.000000000
## 12               {root vegetables,yogurt} => {whole milk} 0.125984252
## 13     {root vegetables,rolls/buns} => {other vegetables} 0.004184100
## 14            {root vegetables,rolls/buns} => {whole milk} 0.046025105
## 15               {other vegetables,yogurt} => {whole milk} 0.025761124
```

Association Rule Visualization

```
rules <-apriori(Groceries,parameter =list(support =0.01,confidence =0.2))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.2    0.1    1 none FALSE            TRUE       5    0.01      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: 98
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[169 item(s), 9835 transaction(s)] done [0.00s].
## sorting and recoding items ... [88 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 3 4 done [0.00s].
## writing ... [232 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```
sub_rules <-head(rules,n =10,by ="lift")# extract top 10 rules with largest lift
inspect(sub_rules)# inspect
```

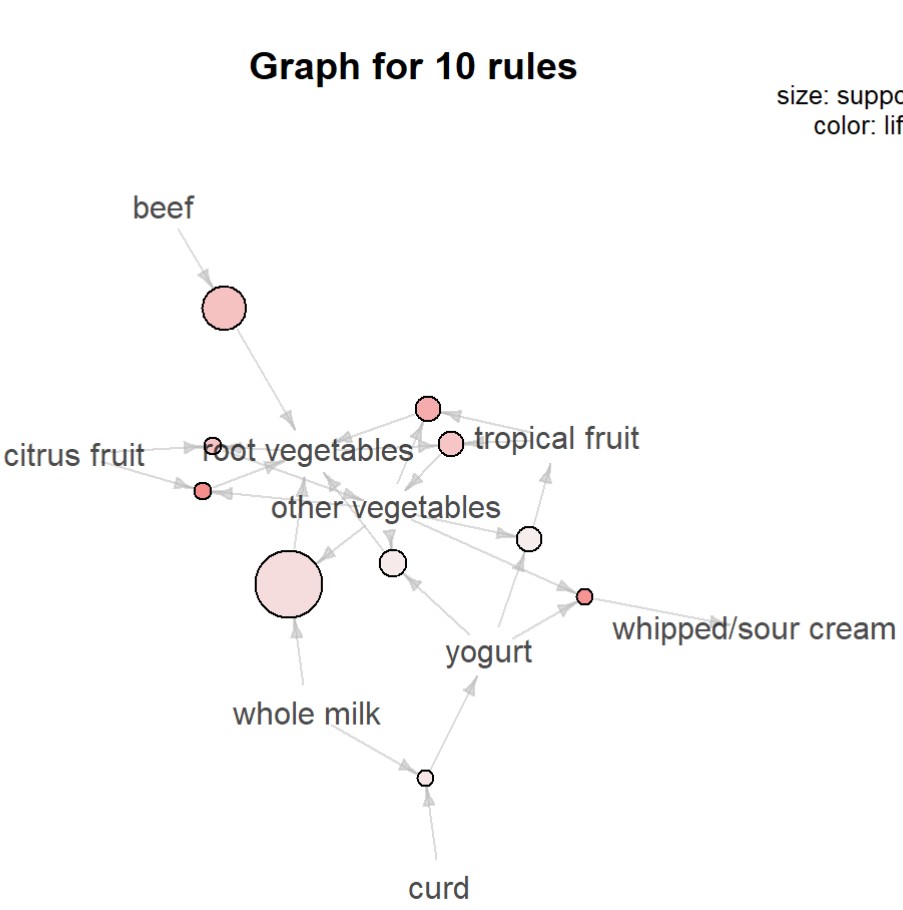
```
##      lhs                                rhs                  support
## [1]  {citrus fruit,other vegetables}   => {root vegetables}    0.01037112
## [2]  {other vegetables,yogurt}         => {whipped/sour cream} 0.01016777
## [3]  {tropical fruit,other vegetables} => {root vegetables}    0.01230300
## [4]  {beef}                            => {root vegetables}    0.01738688
## [5]  {citrus fruit,root vegetables}    => {other vegetables}   0.01037112
## [6]  {tropical fruit,root vegetables}  => {other vegetables}   0.01230300
## [7]  {other vegetables,whole milk}     => {root vegetables}    0.02318251
## [8]  {whole milk,curd}                 => {yogurt}             0.01006609
## [9]  {other vegetables,yogurt}         => {root vegetables}    0.01291307
## [10] {other vegetables,yogurt}         => {tropical fruit}     0.01230300
##      confidence coverage   lift     count
## [1]  0.3591549  0.02887646 3.295045 102
## [2]  0.2341920  0.04341637 3.267062 100
## [3]  0.3427762  0.03589222 3.144780 121
## [4]  0.3313953  0.05246568 3.040367 171
## [5]  0.5862069  0.01769192 3.029608 102
## [6]  0.5845411  0.02104728 3.020999 121
## [7]  0.3097826  0.07483477 2.842082 228
## [8]  0.3852140  0.02613116 2.761356  99
## [9]  0.2974239  0.04341637 2.728698 127
## [10] 0.2833724  0.04341637 2.700550 121
```

```
#install.packages('arulesViz')

library(arulesViz)# load package
```

```
## Loading required package: grid
```

```
plot(sub_rules,method ="graph")
```


Graph for 10 rules
size: support (0.01 - 0.023)
color: lift (2.701 - 3.295)
beef
citrus fruit
root vegetables
tropical fruit
other vegetables
whipped/sour cream
yogurt
whole milk
curd