MO433 – Aprendizado de Máquina Não Supervisionado

Trabalho 1

Thiago Bruschi Martins RA 120212

Obs: Este relatório foi desenvolvido utilizando a linguagem R no RStudio.

```r
# Importando as bibliotecas necessárias para gerar regras e visualiza-las
library(arules)
```

```
## Carregando pacotes exigidos: Matrix
```

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

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

```r
library(arulesViz)

# Setando o workspace
setwd("C:\\Users\\thiag\\OneDrive\\Documents\\Unicamp\\Master\\Unsupervised-Learning\\Trabalho 1")
```

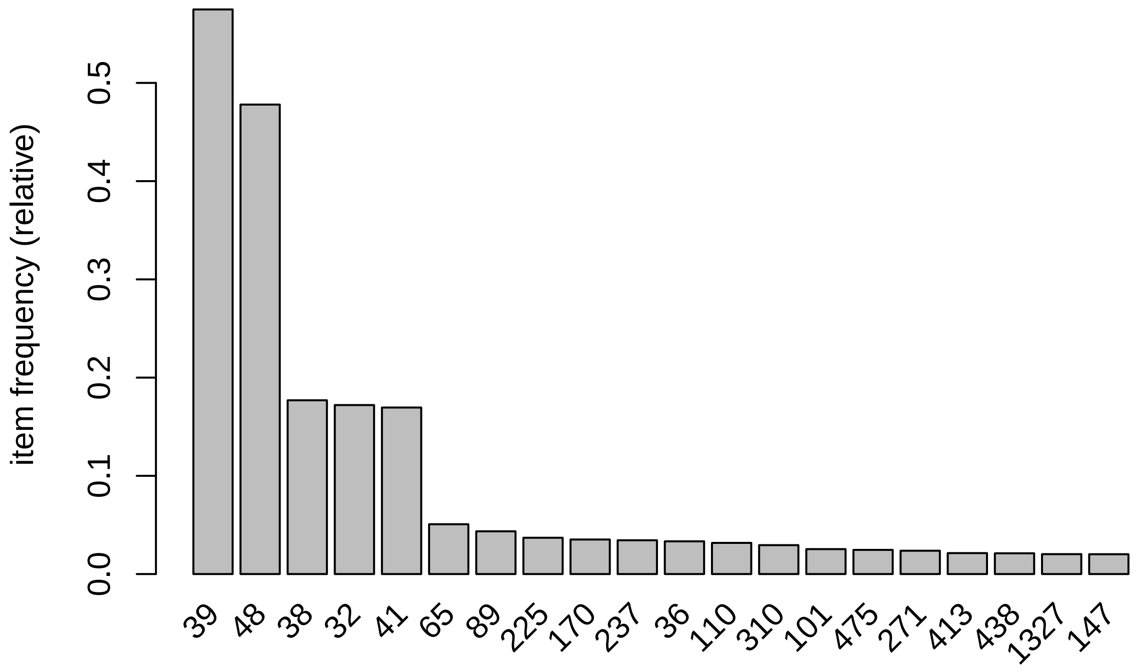
Lendo o arquivo e visualizando o resumo das transações. Neste resumo podemos ver os itens mais frequentes e também a distribuição do tamanho das compras

```r
transacoes <- read.transactions('retail.txt', format = "basket", sep=" ")
summary(transacoes)
```

```
## transactions as itemMatrix in sparse format with
##  88162 rows (elements/itemsets/transactions) and
##  16470 columns (items) and a density of 0.0006257289
##
## most frequent items:
##      39      48      38      32      41 (Other)
##   50675   42135   15596   15167   14945  770058
##
## element (itemset/transaction) length distribution:
## sizes
##    1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16
## 3016 5516 6919 7210 6814 6163 5746 5143 4660 4086 3751 3285 2866 2620 2310 2115
##   17   18   19   20   21   22   23   24   25   26   27   28   29   30   31   32
## 1874 1645 1469 1290 1205  981  887  819  684  586  582  472  480  355  310  303
##   33   34   35   36   37   38   39   40   41   42   43   44   45   46   47   48
##  272  234  194  136  153  123  115  112   76   66   71   60   50   44   37   37
##   49   50   51   52   53   54   55   56   57   58   59   60   61   62   63   64
##   33   22   24   21   21   10   11   10    9   11    4    9    7    4    5    2
##   65   66   67   68   71   73   74   76
##    2    5    3    3    1    1    1    1
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00    4.00    8.00   10.31   14.00   76.00
##
## includes extended item information - examples:
##   labels
## 1      0
## 2      1
## 3     10
```

Plotando um gráfico para melhor visualizar os itens mais frequentes do conjunto de dados. Vemos neste gráfico que os items 39 e 48 são muito mais frequentes que os demais, tendo suporte superior a 0.4. Depois deles, apenas três items possuem suporte superior a 0.1.

```
itemFrequencyPlot(transacoes, topN=20, support=0.005)
```



Agora vamos gerar regras utilizando o algoritmo Apriori. Através de uma lista de parâmetros nós definimos o suporte e a confiança mínima que desejamos para as regras. Na saída do comando podemos ver que foram geradas 37 regras.

```r
rules <- apriori(transacoes, parameter = list(supp = 0.005, conf = 0.9, target = "rules"))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.9    0.1    1 none FALSE            TRUE       5   0.005      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: 440
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[16470 item(s), 88162 transaction(s)] done [0.99s].
## sorting and recoding items ... [221 item(s)] done [0.02s].
## creating transaction tree ... done [0.09s].
## checking subsets of size 1 2 3 4 5 done [0.05s].
## writing ... [37 rule(s)] done [0.00s].
## creating S4 object  ... done [0.01s].
```

Utilizamos o comando inspect para visualizar as regras geradas e o comando head para ordená-las por lift. É notável que praticamente todas as regras, com exceção da primeira, possuem o item 38. O curioso é que o item 38 é apenas o terceiro item mais frequênte do conjunto de dados.

```r
inspect(head(rules, n = 37, by = "lift"))
```

```
##      lhs             rhs     support     confidence coverage    lift      count
## [1]  {16011}      => {16010} 0.007384134 0.9730942  0.007588303 65.189915  651
## [2]  {110,39,48} => {38}    0.011694381 0.9942141  0.011762437  5.620153 1031
## [3]  {110,39,41} => {38}    0.005796148 0.9922330  0.005841519  5.608954  511
## [4]  {170,39,48} => {38}    0.013531907 0.9892206  0.013679363  5.591925 1193
## [5]  {110,39}    => {38}    0.019736394 0.9891984  0.019951907  5.591800 1740
## [6]  {371,39}    => {38}    0.005966289 0.9887218  0.006034346  5.589106  526
## [7]  {170,48}    => {38}    0.017445158 0.9877970  0.017660670  5.583878 1538
## [8]  {286,39,48} => {38}    0.005194982 0.9870690  0.005263038  5.579762  458
## [9]  {105,39}    => {38}    0.005092897 0.9868132  0.005160954  5.578317  449
## [10] {110,32}    => {38}    0.005024841 0.9866370  0.005092897  5.577320  443
## [11] {170,41}    => {38}    0.009006148 0.9863354  0.009130918  5.575616  794
## [12] {110,48}    => {38}    0.015437490 0.9862319  0.015653002  5.575030 1361
## [13] {37,48}     => {38}    0.006317915 0.9858407  0.006408657  5.572819  557
## [14] {170,39,41} => {38}    0.006975795 0.9855769  0.007077879  5.571328  615
## [15] {170,32}    => {38}    0.006034346 0.9851852  0.006125088  5.569114  532
## [16] {110,41}    => {38}    0.007554275 0.9837518  0.007679045  5.561011  666
## [17] {170,41,48} => {38}    0.005489894 0.9837398  0.005580636  5.560943  484
## [18] {286,48}    => {38}    0.006590141 0.9830795  0.006703568  5.557211  581
## [19] {371}       => {38}    0.008699893 0.9808184  0.008870035  5.544429  767
## [20] {170,39}    => {38}    0.022901023 0.9805731  0.023354733  5.543042 2019
## [21] {105}       => {38}    0.007293392 0.9786910  0.007452190  5.532403  643
## [22] {170}       => {38}    0.034379892 0.9780574  0.035151199  5.528821 3031
## [23] {110}       => {38}    0.030909008 0.9753042  0.031691659  5.513258 2725
## [24] {37}        => {38}    0.011864522 0.9739292  0.012182119  5.505485 1046
## [25] {790}       => {38}    0.005762120 0.9713193  0.005932261  5.490732  508
## [26] {286,39}    => {38}    0.008257526 0.9706667  0.008507067  5.487042  728
## [27] {36,39,48}  => {38}    0.012250176 0.9677419  0.012658515  5.470509 1080
## [28] {37,39}     => {38}    0.007758445 0.9674682  0.008019328  5.468962  684
## [29] {36,39,41}  => {38}    0.006272544 0.9667832  0.006488056  5.465090  553
## [30] {56}        => {38}    0.005830176 0.9607477  0.006068374  5.430972  514
## [31] {36,48}     => {38}    0.015426147 0.9604520  0.016061342  5.429300 1360
## [32] {36,41}     => {38}    0.007610989 0.9585714  0.007939929  5.418670  671
## [33] {32,36}     => {38}    0.005353781 0.9554656  0.005603321  5.401113  472
## [34] {36,39}     => {38}    0.022061659 0.9548355  0.023105193  5.397551 1945
## [35] {36}        => {38}    0.031646288 0.9502725  0.033302330  5.371757 2790
## [36] {286}       => {38}    0.012658515 0.9433643  0.013418480  5.332706 1116
## [37] {55}        => {38}    0.007452190 0.9332386  0.007985300  5.275467  657
```