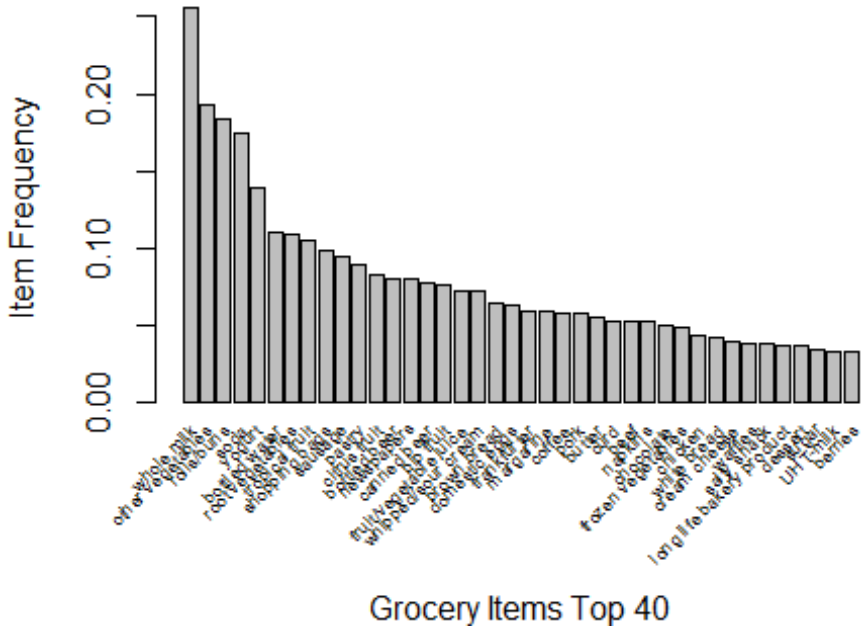
Groceries Dataset Analysis and Association Rule Mining

Author: Nandini Ethirajulu

```
#Considering the "groceries" dataset as source data.



###Installing arules (association rules) package
#install.packages("arules")
library("arules")

##Setting the directory for the source data
setwd("C:/Users/nandi/Documents/Personal/Academics/Projects/Groceries -
Association Rules")


##Analysing the transactions
groceries_dats <- read.transactions("groceries.csv", sep = ",")


summary(groceries_dats)

## transactions as itemMatrix in sparse format with
##  9835 rows (elements/itemsets/transactions) and
##  169 columns (items) and a density of 0.02609146
##
## most frequent items:
##       whole milk other vegetables       rolls/buns             soda
##             2513             1903             1809             1715
##           yogurt          (Other)
##             1372            34055
##
## element (itemset/transaction) length distribution:
## sizes
##    1    2    3    4    5    6    7    8    9   10   11   12   13   14   15
16
## 2159 1643 1299 1005  855  645  545  438  350  246  182  117   78   77   55
46
##   17   18   19   20   21   22   23   24   26   27   28   29   32
##   29   14   14    9   11    4    6    1    1    1    1    3    1
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   2.000   3.000   4.409   6.000  32.000
##
## includes extended item information - examples:
##             labels
```

```
## 1 abrasive cleaner
## 2 artif. sweetener
## 3   baby cosmetics
```

```
############ (A) Visualizing the item frequency plot for the top 40 grocery
items. ############

##using itemFrequencyPlot to fetch the frequency plot of groceries dataset
(only top 40 items)


itemFrequencyPlot(groceries_dats, topN = 40, xlab = "Grocery Items Top 40" ,
ylab =  "Item Frequency", cex.names=0.59)
```




```
jpeg(filename = "results/item_frequency_plot.jpg", width = 800, height = 600)

# Generate and save groceries_dats frequency plot for the top 40 items
itemFrequencyPlot(groceries_dats, topN = 40, xlab = "Grocery Items Top 40",
ylab = "Item Frequency", cex.names = 0.59)
```

```
############ (B) Ranking the top five association rules with the highest
"confidence". ############

# Generating association rules
#?apriori --help file

##Using apriori algorithm to fetch the top five rules based on confidence

rule_params <- list(support = .005, confidence = .01, minlen = 2, maxlen = 6)
groceries_arules <- apriori(groceries_dats, parameter = rule_params)

## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##        0.01    0.1    1 none FALSE            TRUE       5   0.005      2
##  maxlen target  ext
##       6  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 49
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[169 item(s), 9835 transaction(s)] done [0.00s].
## sorting and recoding items ... [120 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 3 4 done [0.00s].
## writing ... [2050 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].

##top five rules with the highest "confidence" is given below
inspect(sort(groceries_arules, by = "confidence")[1:5])

##     lhs                     rhs              support confidence
coverage     lift count
## [1] {root vegetables,
##      tropical fruit,
##      yogurt}             => {whole milk} 0.005693950  0.7000000
0.008134215 2.739554    56
## [2] {other vegetables,
##      pip fruit,
##      root vegetables}    => {whole milk} 0.005490595  0.6750000
0.008134215 2.641713    54
## [3] {butter,
##      whipped/sour cream} => {whole milk} 0.006710727  0.6600000
0.010167768 2.583008    66
## [4] {pip fruit,
```

```
##      whipped/sour cream} => {whole milk} 0.005998983  0.6483516
0.009252669 2.537421    59
## [5] {butter,
##      yogurt}             => {whole milk} 0.009354347  0.6388889
0.014641586 2.500387    92
```

```
############ (C) Ranking the top ten association rules with the highest
"lift". ############
```

```
##Using apriori algorithm to fetch the top ten rules based on lift
```

```
inspect(sort(groceries_arules, by = "lift")[1:10])
```

```
##      lhs                     rhs                      support confidence
coverage     lift count
## [1]  {ham}                => {white bread}        0.005083884 0.19531250
0.02602949 4.639851    50
## [2]  {white bread}        => {ham}                0.005083884 0.12077295
0.04209456 4.639851    50
## [3]  {citrus fruit,
##       other vegetables,
##       whole milk}         => {root vegetables}    0.005795628 0.44531250
0.01301474 4.085493    57
## [4]  {butter,
##       other vegetables}   => {whipped/sour cream} 0.005795628 0.28934010
0.02003050 4.036397    57
## [5]  {root vegetables}    => {herbs}              0.007015760 0.06436567
0.10899847 3.956477    69
## [6]  {herbs}              => {root vegetables}    0.007015760 0.43125000
0.01626843 3.956477    69
## [7]  {other vegetables,
##       root vegetables}    => {onions}             0.005693950 0.12017167
0.04738180 3.875044    56
## [8]  {citrus fruit,
##       pip fruit}          => {tropical fruit}     0.005592272 0.40441176
0.01382816 3.854060    55
## [9]  {berries}            => {whipped/sour cream} 0.009049314 0.27217125
0.03324860 3.796886    89
## [10] {whipped/sour cream} => {berries}            0.009049314 0.12624113
0.07168277 3.796886    89
```

########### (D) consistency and differences between top rules based on highest “confidence” and highest “lift”. ###########

Anaslysing the output ranks of B (rules based on confidence sorting), we understand that different combinational items that are associated with whole milk have the highest confidence values. All the top five rules have whole milk in the RHS. More specifically, there is 70% chance of purchasing whole milk whenever(root vegetables, tropical fruit, yogurt) items are bought together.

Analysing the output ranks of C (top rules based on lift sorting)-- we understand that likelihood of purchasing white bread is about 4.64 times increased when ham is purchased. Vice versa, the second rule indicates association/likelihood of purchasing ham is higher whenever white bread is purchased

#Consistently both the rankings indicates in common that, whole milk, white bread, root vegetables are regularly associated with other vegetables, ham and Whipped/Sour Cream.

#Differences - We see that Citrus fruits and Berries are repeated in the top 10 rankings based on lift sorting. However, it has not been picked up in the top 5 ranks of confidence.

```
########### (E) Recommendations to increase their pastry sales ###########


groceries_pastry_rules_e = subset(groceries_arules, rhs %in% "pastry")
#groceries_pastry_rules_e

groceries_pastry_rules_e_sort = sort(groceries_pastry_rules_e, by = "lift")
inspect(groceries_pastry_rules_e_sort[1:10])

##      lhs                              rhs      support     confidence
coverage
## [1]  {soda, whole milk}           => {pastry} 0.008235892 0.2055838
0.04006101
## [2]  {sausage, whole milk}        => {pastry} 0.005693950 0.1904762
0.02989324
## [3]  {waffles}                    => {pastry} 0.007015760 0.1825397
0.03843416
## [4]  {pip fruit, whole milk}      => {pastry} 0.005083884 0.1689189
0.03009659
## [5]  {rolls/buns, yogurt}         => {pastry} 0.005795628 0.1686391
0.03436706
## [6]  {other vegetables, soda}     => {pastry} 0.005490595 0.1677019
0.03274021
## [7]  {whole milk, yogurt}         => {pastry} 0.009150991 0.1633394
0.05602440
## [8]  {chocolate}                  => {pastry} 0.008032537 0.1618852
0.04961871
## [9]  {tropical fruit, whole milk} => {pastry} 0.006710727 0.1586538
0.04229792
## [10] {long life bakery product}   => {pastry} 0.005897306 0.1576087
0.03741739
##      lift     count
## [1]  2.310761 81
## [2]  2.140952 56
## [3]  2.051746 69
## [4]  1.898649 50
## [5]  1.895503 57
## [6]  1.884969 54
## [7]  1.835935 90
## [8]  1.819590 79
## [9]  1.783269 66
## [10] 1.771522 58

#Rules are filtered specifically for pastries, and sorted based on the lift
values

### Recommendations to increase pastry sales --

## From the top ranked rules, we see that purchase of soda and whole milk has
a chance of increasing the pastry sales (lift value is greater than 1
```

```
indicating the high association of these items )

##Sausage, pip fruit, rolls/buns, yogurt are some of the items that increase
the pastry sales whenever they are bought together with whole milk.

##In addition,  waffles whenever they are bought alone by the customers have
also promoted the sales of pastries.

##In conclusion, in order to increase the pastry sales, it is recommended to
promote or bundle or organize the above suggested products together in the
store to increase the sales of the pastries


##Printing the complete Pastry sales association rules
inspect(groceries_pastry_rules_e_sort)
```

```
##      lhs                                   rhs      support
confidence
## [1]  {soda, whole milk}                 => {pastry} 0.008235892
0.20558376
## [2]  {sausage, whole milk}              => {pastry} 0.005693950
0.19047619
## [3]  {waffles}                          => {pastry} 0.007015760
0.18253968
## [4]  {pip fruit, whole milk}            => {pastry} 0.005083884
0.16891892
## [5]  {rolls/buns, yogurt}               => {pastry} 0.005795628
0.16863905
## [6]  {other vegetables, soda}           => {pastry} 0.005490595
0.16770186
## [7]  {whole milk, yogurt}               => {pastry} 0.009150991
0.16333938
## [8]  {chocolate}                        => {pastry} 0.008032537
0.16188525
## [9]  {tropical fruit, whole milk}       => {pastry} 0.006710727
0.15865385
## [10] {long life bakery product}         => {pastry} 0.005897306
0.15760870
## [11] {sugar}                            => {pastry} 0.005185562
0.15315315
## [12] {other vegetables, yogurt}         => {pastry} 0.006609049
0.15222482
## [13] {rolls/buns, whole milk}           => {pastry} 0.008540925
0.15080790
## [14] {brown bread}                      => {pastry} 0.009659380
0.14890282
## [15] {dessert}                          => {pastry} 0.005388917
0.14520548
## [16] {other vegetables, rolls/buns}     => {pastry} 0.006100661
0.14319809
```

```
## [17] {domestic eggs}                       => {pastry} 0.009049314
0.14262821
## [18] {other vegetables, tropical fruit}  => {pastry} 0.005083884
0.14164306
## [19] {frankfurter}                         => {pastry} 0.008337570
0.14137931
## [20] {other vegetables, whole milk}      => {pastry} 0.010574479
0.14130435
## [21] {curd}                                => {pastry} 0.007524148
0.14122137
## [22] {pip fruit}                           => {pastry} 0.010676157
0.14112903
## [23] {rolls/buns, soda}                    => {pastry} 0.005388917
0.14058355
## [24] {butter}                              => {pastry} 0.007625826
0.13761468
## [25] {salty snack}                         => {pastry} 0.005185562
0.13709677
## [26] {napkins}                             => {pastry} 0.007015760
0.13398058
## [27] {sausage}                             => {pastry} 0.012506355
0.13311688
## [28] {white bread}                         => {pastry} 0.005592272
0.13285024
## [29] {whole milk}                          => {pastry} 0.033248602
0.13012336
## [30] {yogurt}                              => {pastry} 0.017691917
0.12682216
## [31] {tropical fruit}                      => {pastry} 0.013218099
0.12596899
## [32] {other vegetables, root vegetables} => {pastry} 0.005897306
0.12446352
## [33] {shopping bags}                       => {pastry} 0.011896289
0.12074303
## [34] {soda}                                => {pastry} 0.021047280
0.12069971
## [35] {beef}                                => {pastry} 0.006304016
0.12015504
## [36] {coffee}                              => {pastry} 0.006914082
0.11908932
## [37] {fruit/vegetable juice}               => {pastry} 0.008540925
0.11814346
## [38] {citrus fruit}                        => {pastry} 0.009761057
0.11793612
## [39] {other vegetables}                    => {pastry} 0.022572445
0.11665791
## [40] {root vegetables, whole milk}       => {pastry} 0.005693950
0.11642412
## [41] {margarine}                           => {pastry} 0.006812405
0.11631944
```

```
## [42] {rolls/buns}                          => {pastry} 0.020945602
0.11387507
## [43] {pork}                                => {pastry} 0.006304016
0.10934744
## [44] {newspapers}                          => {pastry} 0.008439248
0.10573248
## [45] {whipped/sour cream}                  => {pastry} 0.007524148
0.10496454
## [46] {root vegetables}                     => {pastry} 0.010981190
0.10074627
## [47] {bottled water}                       => {pastry} 0.008947636
0.08095676
##      coverage   lift     count
## [1]  0.04006101 2.310761  81
## [2]  0.02989324 2.140952  56
## [3]  0.03843416 2.051746  69
## [4]  0.03009659 1.898649  50
## [5]  0.03436706 1.895503  57
## [6]  0.03274021 1.884969  54
## [7]  0.05602440 1.835935  90
## [8]  0.04961871 1.819590  79
## [9]  0.04229792 1.783269  66
## [10] 0.03741739 1.771522  58
## [11] 0.03385867 1.721441  51
## [12] 0.04341637 1.711007  65
## [13] 0.05663447 1.695081  84
## [14] 0.06487036 1.673668  95
## [15] 0.03711235 1.632110  53
## [16] 0.04260295 1.609547  60
## [17] 0.06344687 1.603141  89
## [18] 0.03589222 1.592068  50
## [19] 0.05897306 1.589103  82
## [20] 0.07483477 1.588261 104
## [21] 0.05327911 1.587328  74
## [22] 0.07564820 1.586290 105
## [23] 0.03833249 1.580159  53
## [24] 0.05541434 1.546789  75
## [25] 0.03782410 1.540968  51
## [26] 0.05236401 1.505942  69
## [27] 0.09395018 1.496234 123
## [28] 0.04209456 1.493237  55
## [29] 0.25551601 1.462587 327
## [30] 0.13950178 1.425481 174
## [31] 0.10493137 1.415891 130
## [32] 0.04738180 1.398970  58
## [33] 0.09852567 1.357152 117
## [34] 0.17437722 1.356665 207
## [35] 0.05246568 1.350543  62
## [36] 0.05805796 1.338564  68
## [37] 0.07229283 1.327932  84
```

```
## [38] 0.08276563 1.325602  96
## [39] 0.19349263 1.311235 222
## [40] 0.04890696 1.308607  56
## [41] 0.05856634 1.307431  67
## [42] 0.18393493 1.279956 206
## [43] 0.05765125 1.229065  62
## [44] 0.07981698 1.188433  83
## [45] 0.07168277 1.179801  74
## [46] 0.10899847 1.132388 108
## [47] 0.11052364 0.909954  88
```