hw2_q3

Lufang Liu

8/18/2017

Practice with association rule mining

Pick your own thresholds for lift and confidence; just be clear what these thresholds are and how you picked them. Do your discovered item sets make sense? Present your discoveries in an interesting and concise way.

Dataset Loading and Initializing

```
library(arules)
```

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

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

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

```
groceries <- read.transactions ("~/Documents/STA380/groceries.txt", format = 'basket',sep=',',rm.duplica

inspect(groceries[1:10])
```

```
##      items
## [1]  {citrus fruit,
##       margarine,
##       ready soups,
##       semi-finished bread}
## [2]  {coffee,
##       tropical fruit,
##       yogurt}
## [3]  {whole milk}
## [4]  {cream cheese,
##       meat spreads,
##       pip fruit,
##       yogurt}
## [5]  {condensed milk,
##       long life bakery product,
##       other vegetables,
##       whole milk}
## [6]  {abrasive cleaner,
##       butter,
##       rice,
##       whole milk,
##       yogurt}
## [7]  {rolls/buns}
## [8]  {bottled beer,
```

```
##       liquor (appetizer),
##       other vegetables,
##       rolls/buns,
##       UHT-milk}
## [9]  {pot plants}
## [10] {cereals,
##       whole milk}
```

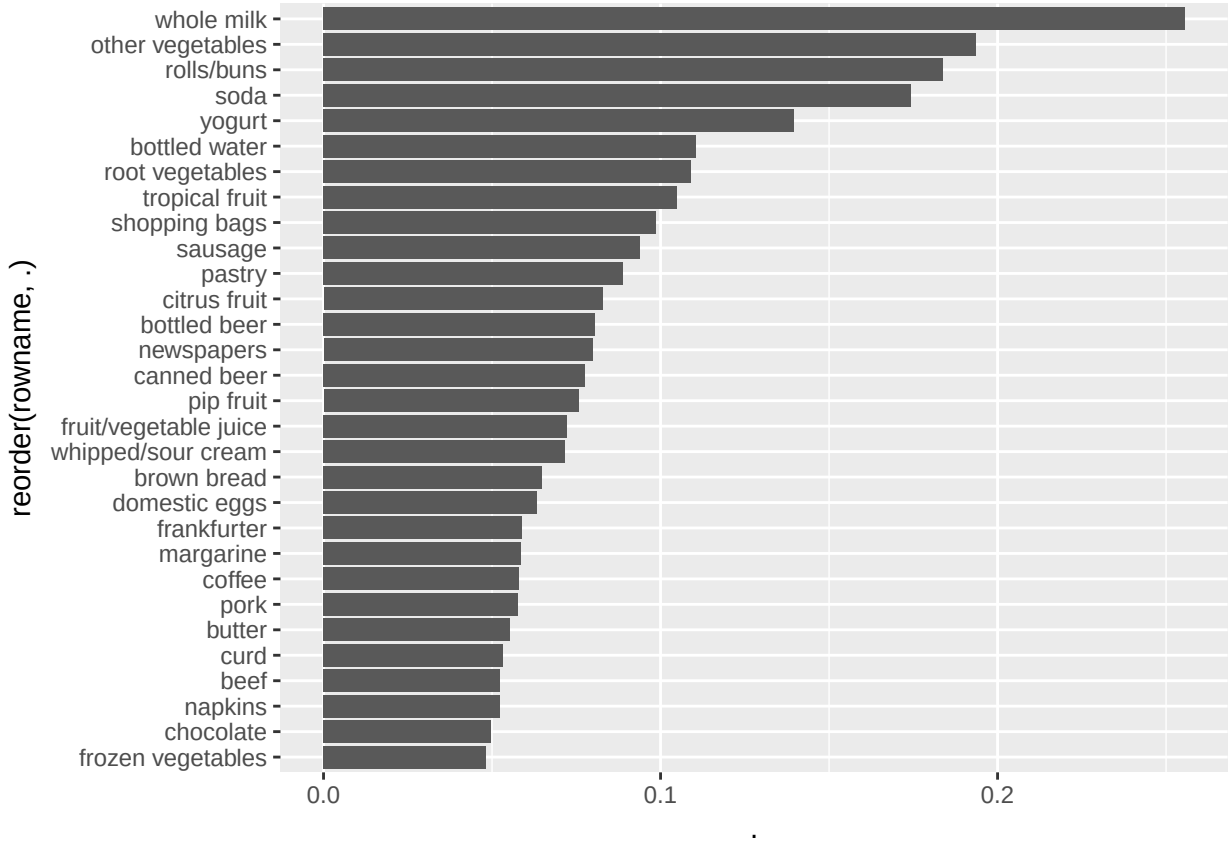
```
summary(groceries)
```

```
## 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
## 2159 1643 1299 1005  855  645  545  438  350  246  182  117   78   77   55
##   16   17   18   19   20   21   22   23   24   26   27   28   29   32
##   46   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
```

```
itemFrequencyGGPlot <- function(x, topN) {
  library(tidyverse)
  x %>%
    itemFrequency %>%
    sort %>%
    tail(topN) %>%
    as.data.frame %>%
    tibble::rownames_to_column() %>%
    ggplot(aes(reorder(rowname, `.`),`.`)) +
    geom_col() +
    coord_flip()
}
itemFrequencyGGPlot(groceries, 30)
```

```
## Loading tidyverse: ggplot2
## Loading tidyverse: tibble
## Loading tidyverse: tidyr
## Loading tidyverse: readr
## Loading tidyverse: purrr
```

```
## Loading tidyverse: dplyr
## Conflicts with tidy packages ----------------------------------------------
## expand(): tidyr, Matrix
## filter(): dplyr, stats
## lag():    dplyr, stats
## recode(): dplyr, arules
```



We first read in the given groceries dataset and created a transactions object using the "read.transactions" function in R. The object format satisfied the format expected by the "arules" package. We then inspected and verified the first 10 items in this object. Next, we did a summary statistics on the object created and plotted the frequency of each food item.

Apriori Algorithm Applying and Parameters Selecting

```
groceryrules <- apriori(groceries,parameter=list(support=0.001, confidence=0.4, maxlen=10))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.4    0.1    1 none FALSE            TRUE       5   0.001      1
##  maxlen target   ext
##      10  rules FALSE
##
```

```
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 9
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[169 item(s), 9835 transaction(s)] done [0.00s].
## sorting and recoding items ... [157 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 3 4 5 6 done [0.01s].
## writing ... [8955 rule(s)] done [0.00s].
## creating S4 object  ... done [0.01s].
```

```
groceryrules_confidence <- sort(groceryrules, by="confidence", decreasing=TRUE)
inspect(groceryrules_confidence[1:10])
```

```
##      lhs                     rhs                    support confidence     lift
## [1]  {rice,
##       sugar}              => {whole milk}       0.001220132          1 3.913649
## [2]  {canned fish,
##       hygiene articles}   => {whole milk}       0.001118454          1 3.913649
## [3]  {butter,
##       rice,
##       root vegetables}    => {whole milk}       0.001016777          1 3.913649
## [4]  {flour,
##       root vegetables,
##       whipped/sour cream} => {whole milk}       0.001728521          1 3.913649
## [5]  {butter,
##       domestic eggs,
##       soft cheese}        => {whole milk}       0.001016777          1 3.913649
## [6]  {citrus fruit,
##       root vegetables,
##       soft cheese}        => {other vegetables} 0.001016777          1 5.168156
## [7]  {butter,
##       hygiene articles,
##       pip fruit}          => {whole milk}       0.001016777          1 3.913649
## [8]  {hygiene articles,
##       root vegetables,
##       whipped/sour cream} => {whole milk}       0.001016777          1 3.913649
## [9]  {hygiene articles,
##       pip fruit,
##       root vegetables}    => {whole milk}       0.001016777          1 3.913649
## [10] {cream cheese,
##       domestic eggs,
##       sugar}              => {whole milk}       0.001118454          1 3.913649
```

```
summary (groceryrules)
```

```
## set of 8955 rules
##
## rule length distribution (lhs + rhs):sizes
##    2    3    4    5    6
##   81 2771 4804 1245   54
##
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.000   3.000   4.000   3.824   4.000   6.000
##
## summary of quality measures:
##     support           confidence          lift
##  Min.   :0.001017   Min.   :0.4000   Min.   : 1.565
##  1st Qu.:0.001118   1st Qu.:0.4583   1st Qu.: 2.316
##  Median :0.001322   Median :0.5319   Median : 2.870
##  Mean   :0.001811   Mean   :0.5579   Mean   : 3.191
##  3rd Qu.:0.001830   3rd Qu.:0.6296   3rd Qu.: 3.733
##  Max.   :0.056024   Max.   :1.0000   Max.   :21.494
##
## mining info:
##       data ntransactions support confidence
##  groceries          9835   0.001        0.4
```

```
groceryrules_lift <- sort(groceryrules, by="lift", decreasing=TRUE)
inspect(groceryrules_lift[1:10])
```

```
##      lhs                          rhs                  support confidence     lift
## [1]  {bottled beer,
##       liquor}                  => {red/blush wine} 0.001931876  0.4130435 21.49356
## [2]  {Instant food products,
##       soda}                    => {hamburger meat} 0.001220132  0.6315789 18.99565
## [3]  {processed cheese,
##       white bread}             => {ham}            0.001931876  0.4634146 17.80345
## [4]  {popcorn,
##       soda}                    => {salty snack}    0.001220132  0.6315789 16.69779
## [5]  {baking powder,
##       flour}                   => {sugar}          0.001016777  0.5555556 16.40807
## [6]  {ham,
##       processed cheese}        => {white bread}    0.001931876  0.6333333 15.04549
## [7]  {Instant food products,
##       whole milk}              => {hamburger meat} 0.001525165  0.5000000 15.03823
## [8]  {curd,
##       other vegetables,
##       whipped/sour cream,
##       yogurt}                  => {cream cheese}   0.001016777  0.5882353 14.83409
## [9]  {Instant food products,
##       rolls/buns}              => {hamburger meat} 0.001016777  0.4347826 13.07672
## [10] {flour,
##       margarine}               => {sugar}          0.001626843  0.4324324 12.77169
```

```
summary (groceryrules)
```

```
## set of 8955 rules
##
## rule length distribution (lhs + rhs):sizes
##    2    3    4    5    6
##   81 2771 4804 1245   54
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.000   3.000   4.000   3.824   4.000   6.000
##
## summary of quality measures:
```

```
##     support           confidence          lift
##  Min.   :0.001017   Min.   :0.4000   Min.   : 1.565
##  1st Qu.:0.001118   1st Qu.:0.4583   1st Qu.: 2.316
##  Median :0.001322   Median :0.5319   Median : 2.870
##  Mean   :0.001811   Mean   :0.5579   Mean   : 3.191
##  3rd Qu.:0.001830   3rd Qu.:0.6296   3rd Qu.: 3.733
##  Max.   :0.056024   Max.   :1.0000   Max.   :21.494
##
## mining info:
##       data ntransactions support confidence
##  groceries          9835   0.001        0.4
```

Support is the fraction of which our item set occurs in our dataset. Therefore, we chose a relatively small support ratio to have more rules included for inspection.

Confidence is the probability that a rule is correct for a new transaction with items on the left. We set the minimum confidence to be 0.4 which we believe is moderate. Then we sorted rules by confidence and found the top10-ranked rules are mostly predicting "whole milk" with 100% confidence ratio, which makes sense as whole milk is the most common item for all shoppers.

Lift is the ratio by which by the confidence of a rule exceeds the expected confidence. Based on lift ratio, we sorted the rules again. We found top10-ranked rules all make common sense. For example, first rule says: with bottled beer and liquor in lhs, you will likely see red/blush wine in rhs.

Items Targeting

What are customers likely to buy before buying soda?

```
groceryrules <-apriori(data=groceries, parameter=list(support=0.001, confidence=0.4),
               appearance = list(default="lhs",rhs="soda"),
               control = list(verbose=F))
groceryrules <-sort(groceryrules, decreasing=TRUE,by="confidence")
inspect(groceryrules[1:5])
```

```
##     lhs                   rhs        support confidence     lift
## [1] {coffee,
##      misc. beverages} => {soda} 0.001016777  0.7692308 4.411303
## [2] {bottled water,
##      newspapers,
##      rolls/buns,
##      yogurt}          => {soda} 0.001016777  0.7692308 4.411303
## [3] {bottled beer,
##      bottled water,
##      sausage}         => {soda} 0.001118454  0.7333333 4.205442
## [4] {sausage,
##      shopping bags,
##      white bread}     => {soda} 0.001016777  0.6666667 3.823129
## [5] {bottled water,
##      chocolate,
##      rolls/buns}      => {soda} 0.001321810  0.6500000 3.727551
```

What are customers likely to buy if they purchase soda?

```
groceryrules_2<-apriori(data=groceries, parameter=list(supp=0.001,conf = 0.15,minlen=2),
               appearance = list(default="rhs",lhs="soda"),
               control = list(verbose=F))
groceryrules_2<-sort(groceryrules_2, decreasing=TRUE,by="confidence")
inspect(groceryrules_2[1:5])
```

```
##     lhs       rhs                support    confidence lift
## [1] {soda} => {whole milk}       0.04006101 0.2297376  0.8991124
## [2] {soda} => {rolls/buns}       0.03833249 0.2198251  1.1951242
## [3] {soda} => {other vegetables} 0.03274021 0.1877551  0.9703476
## [4] {soda} => {bottled water}    0.02897814 0.1661808  1.5035766
## [5] {soda} => {yogurt}           0.02735130 0.1568513  1.1243678
```

These two examples show that these association rules can help store managers to promote the sales of certain goods by placing them closer to other goods that are associated with them.