Market Basket Analysis

Sally

7/22/2017

Load exported clean data set

```
setwd("~/Desktop")
Top_item<-read.csv("~/Desktop/MyData.csv")
```

or call from global environment

```
mydata <- Top_item
```

Loading arules package

```
library(arules)
library(arulesViz)
library(datasets)
```

```
mydata$Cat<-as.factor(mydata$Cat)
mydata<-mydata[mydata$Cat!=c("OTHER"),]
```

See first 10 observations

```
head(mydata, n=10)
```

```
##     X InvoiceNo StockCode                         Description Quantity
## 1   1    536365     71053                WHITE METAL LANTERN        6
## 4   4    536365     21730 GLASS STAR FROSTED T-LIGHT HOLDER        6
## 13 13    536367     22623      BOX OF VINTAGE JIGSAW BLOCKS        3
## 14 14    536367     22622    BOX OF VINTAGE ALPHABET BLOCKS        2
## 19 19    536368     22960           JAM MAKING SET WITH JARS        6
## 30 30    536370     21791 VINTAGE HEADS AND TAILS CARD GAME       24
## 33 33    536370     22629                SPACEBOY LUNCH BOX       24
## 34 34    536370     22659           LUNCH BOX I LOVE LONDON       24
## 35 35    536370     22631           CIRCUS PARADE LUNCH BOX       24
## 36 36    536370     22661   CHARLOTTE BAG DOLLY GIRL DESIGN       20
##            InvoiceDate UnitPrice CustomerID        Country Sales month
## 1  2010-12-01 08:26:00      3.39      17850 United Kingdom 20.34    12
## 4  2010-12-01 08:26:00      4.25      17850 United Kingdom 25.50    12
## 13 2010-12-01 08:34:00      4.95      13047 United Kingdom 14.85    12
## 14 2010-12-01 08:34:00      9.95      13047 United Kingdom 19.90    12
## 19 2010-12-01 08:34:00      4.25      13047 United Kingdom 25.50    12
## 30 2010-12-01 08:45:00      1.25      12583         France 30.00    12
## 33 2010-12-01 08:45:00      1.95      12583         France 46.80    12
## 34 2010-12-01 08:45:00      1.95      12583         France 46.80    12
## 35 2010-12-01 08:45:00      1.95      12583         France 46.80    12
## 36 2010-12-01 08:45:00      0.85      12583         France 17.00    12
```

```
##              Cat
## 1        LANTERN
## 4         HOLDER
## 13       VINTAGE
## 14       VINTAGE
## 19           JAR
## 30          CARD
## 33 LUNCH BAG/BOX
## 34 LUNCH BAG/BOX
## 35 LUNCH BAG/BOX
## 36           BAG
```

Split data

```
dt <- split(mydata$Cat, mydata$InvoiceNo)
```

Convert data to transaction level

```
dt2 <- as(dt,"transactions")
```

```
## Warning in asMethod(object): removing duplicated items in transactions
```

```
summary(dt2)
```

```
## transactions as itemMatrix in sparse format with
##  16218 rows (elements/itemsets/transactions) and
##  19 columns (items) and a density of 0.2014169
##
## most frequent items:
##    SIGN     BAG VINTAGE  HOLDER    CARD (Other)
##    8151    6975    5926    5320    4606   31087
##
## element (itemset/transaction) length distribution:
## sizes
##    1    2    3    4    5    6    7    8    9   10   11   12   13   14   15
## 3155 2839 2627 2182 1740 1293  883  614  382  227  120   76   43   15    9
##   17   18
##    6    7
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   2.000   3.000   3.827   5.000  18.000
##
## includes extended item information - examples:
##   labels
## 1    BAG
## 2 BOTTLE
## 3   CARD
##
## includes extended transaction information - examples:
##   transactionID
## 1        536365
```

```
## 2        536367
## 3        536368
```

```
inspect(head(dt2,5))
```

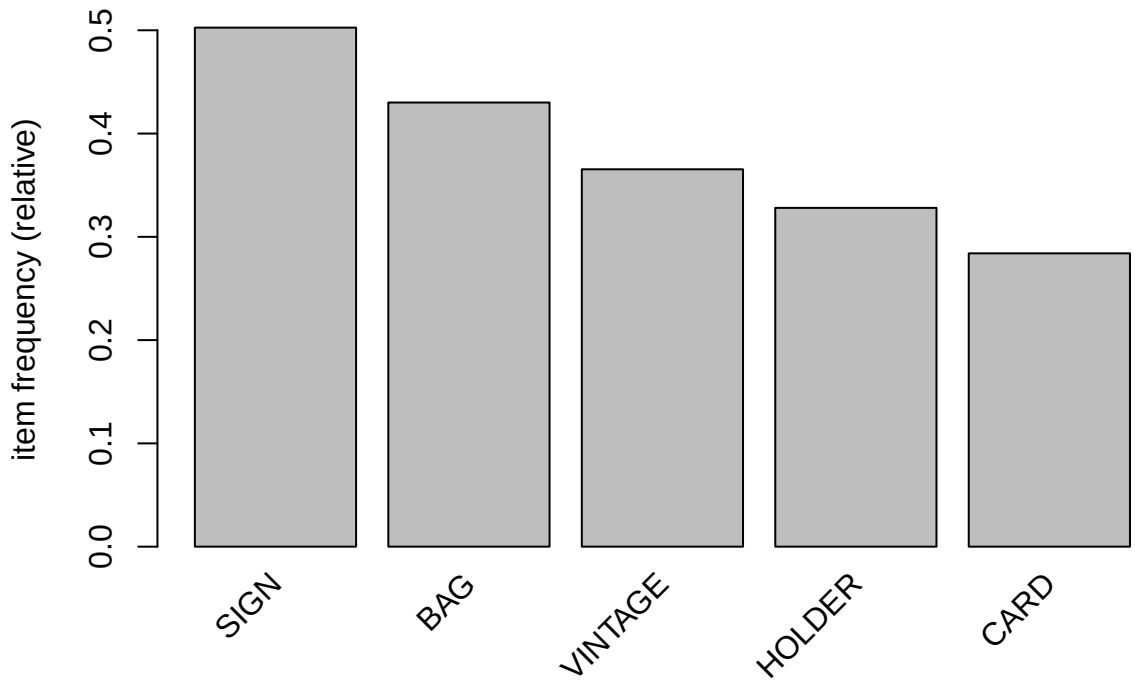
```
##     items                                              transactionID
## [1] {HOLDER,LANTERN}                                   536365
## [2] {VINTAGE}                                          536367
## [3] {JAR}                                              536368
## [4] {BAG,CARD,LIGHT,LUNCH BAG/BOX,PAINT,VINTAGE}       536370
## [5] {CHRISTMAS}                                        536371
```

Most Frequent Items

```
itemFrequency(dt2, type = "relative")
```

```
##           BAG        BOTTLE          CARD         CASES     CHRISTMAS
##  4.300777e-01  2.093970e-01  2.840054e-01  2.095203e-01  2.704403e-01
##       GLIDERS        HOLDER           JAR       LANTERN         LIGHT
##  2.910347e-02  3.280306e-01  1.601307e-01  8.706376e-02  2.806758e-01
## LUNCH BAG/BOX       NAPKINS         PAINT   PAPER CRAFT      PLASTERS
##  2.685288e-01  9.834752e-02  1.110495e-01  6.165988e-05  8.706376e-02
##          SIGN      STICKERS       TISSUES       VINTAGE
##  5.025897e-01  2.608213e-02  7.935627e-02  3.653965e-01
```

```
itemFrequencyPlot(dt2,topN = 5)
```



aggregated data

```
rules = apriori(dt2, parameter=list(support=0.01, confidence=0.8))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.8    0.1    1 none FALSE            TRUE       5    0.01      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: 162
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[19 item(s), 16218 transaction(s)] done [0.00s].
## sorting and recoding items ... [18 item(s)] done [0.00s].
## creating transaction tree ... done [0.01s].
## checking subsets of size 1 2 3 4 5 6 7 done [0.01s].
## writing ... [1033 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```
rules = apriori(dt2, parameter=list(support=0.01, confidence=0.8, minlen = 3))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.8    0.1    1 none FALSE            TRUE       5    0.01      3
##  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: 162
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[19 item(s), 16218 transaction(s)] done [0.00s].
## sorting and recoding items ... [18 item(s)] done [0.00s].
## creating transaction tree ... done [0.01s].
## checking subsets of size 1 2 3 4 5 6 7 done [0.01s].
## writing ... [1033 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```
rules = apriori(dt2, parameter=list(support=0.01, confidence=0.8, maxlen = 4))
```

```
## Apriori
##
## Parameter specification:
```

```
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.8    0.1    1 none FALSE            TRUE       5    0.01      1
##  maxlen target   ext
##       4  rules FALSE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 162
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[19 item(s), 16218 transaction(s)] done [0.00s].
## sorting and recoding items ... [18 item(s)] done [0.00s].
## creating transaction tree ... done [0.01s].
## checking subsets of size 1 2 3 4
## Warning in apriori(dt2, parameter = list(support = 0.01, confidence =
## 0.8, : Mining stopped (maxlen reached). Only patterns up to a length of 4
## returned!
##  done [0.01s].
## writing ... [234 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

Convert rules into data frame

```
rules3 = as(rules, "data.frame")
```

and

```
write(rules, "~/Desktop/ARule2.csv", sep=",")
```

Show only particular product rules

```
inspect(head(sort(subset(rules, subset = rhs %pin% "SIGN" ),by="lift"),10))
```

```
##      lhs                                rhs    support    confidence
## [1]  {BOTTLE,LUNCH BAG/BOX,NAPKINS} => {SIGN} 0.01029720 0.9027027
## [2]  {CARD,JAR,TISSUES}             => {SIGN} 0.01079048 0.8928571
## [3]  {CARD,CASES,NAPKINS}           => {SIGN} 0.01695647 0.8928571
## [4]  {BOTTLE,NAPKINS,VINTAGE}       => {SIGN} 0.01276360 0.8884120
## [5]  {BOTTLE,HOLDER,NAPKINS}        => {SIGN} 0.01171538 0.8878505
## [6]  {BOTTLE,CASES,NAPKINS}         => {SIGN} 0.01122210 0.8878049
## [7]  {CASES,JAR,NAPKINS}            => {SIGN} 0.01202368 0.8863636
## [8]  {BOTTLE,CASES,PLASTERS}        => {SIGN} 0.01165372 0.8831776
## [9]  {BOTTLE,HOLDER,TISSUES}        => {SIGN} 0.01116044 0.8829268
## [10] {BAG,JAR,PAINT}                => {SIGN} 0.01381181 0.8818898
##      lift
## [1]  1.796103
## [2]  1.776513
## [3]  1.776513
```

```
## [4]  1.767669
## [5]  1.766551
## [6]  1.766460
## [7]  1.763593
## [8]  1.757254
## [9]  1.756755
## [10] 1.754691
```

Show the top 10 rules

```
options(digits=2)
inspect(head(sort(rules[1:40],by="lift"),5))
```

```
##     lhs                                  rhs       support confidence lift
## [1] {CHRISTMAS,LUNCH BAG/BOX,TISSUES} => {VINTAGE} 0.010   0.81       2.2
## [2] {CHRISTMAS,LUNCH BAG/BOX,TISSUES} => {BAG}     0.011   0.86       2.0
## [3] {LIGHT,LUNCH BAG/BOX,TISSUES}     => {BAG}     0.012   0.85       2.0
## [4] {LANTERN,LUNCH BAG/BOX,VINTAGE}   => {BAG}     0.011   0.85       2.0
## [5] {BOTTLE,LUNCH BAG/BOX,TISSUES}    => {BAG}     0.010   0.82       1.9
```

Get Summary Information

```
summary(rules)
```

```
## set of 234 rules
##
## rule length distribution (lhs + rhs):sizes
##   3   4
##   8 226
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       3       4       4       4       4       4
##
## summary of quality measures:
##     support        confidence        lift
##  Min.   :0.010   Min.   :0.80   Min.   :1.59
##  1st Qu.:0.013   1st Qu.:0.81   1st Qu.:1.63
##  Median :0.016   Median :0.83   Median :1.67
##  Mean   :0.019   Mean   :0.83   Mean   :1.71
##  3rd Qu.:0.022   3rd Qu.:0.85   3rd Qu.:1.74
##  Max.   :0.046   Max.   :0.90   Max.   :2.22
##
## mining info:
##  data ntransactions support confidence
##   dt2         16218    0.01        0.8
```