Association Analysis HW

Dean D'souza

2016-07-08

## Solutions:

* **First, install the package arules**

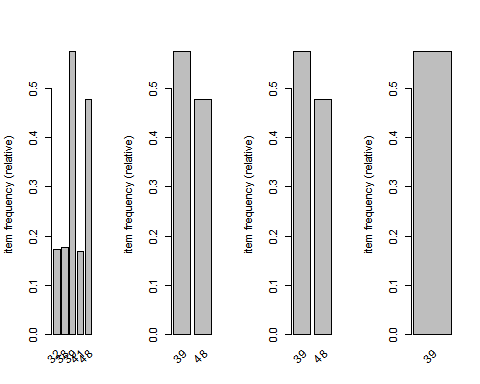
# Intalling the 'arules' package (commented out as already installed)  
# install.packages("arules")

* **Second, load the data set using the following command**

library(arules)  
grd <- read.transactions("http://fimi.ua.ac.be/data/retail.dat", format="basket")

* **Third, run the following commands and interpret the results**

par(mfrow=c(1,4)) # Setting the following graphs to appear in a single line  
itemFrequencyPlot(grd,support=.1)#run with support .2, .3, & .5  
itemFrequencyPlot(grd,support=.2)  
itemFrequencyPlot(grd,support=.3)  
itemFrequencyPlot(grd,support=.5)



summary(grd)

## 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   
## 3016 5516 6919 7210 6814 6163 5746 5143 4660 4086 3751 3285 2866 2620 2310   
## 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30   
## 2115 1874 1645 1469 1290 1205 981 887 819 684 586 582 472 480 355   
## 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45   
## 310 303 272 234 194 136 153 123 115 112 76 66 71 60 50   
## 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60   
## 44 37 37 33 22 24 21 21 10 11 10 9 11 4 9   
## 61 62 63 64 65 66 67 68 71 73 74 76   
## 7 4 5 2 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

#inspect(grd) #you will have to stop the listing manually (Commented out to make the document smaller)

* **Create the rules object using apriori**

grdar<-apriori(grd,parameter=list(supp=.05,conf=.5))

## Apriori  
##   
## Parameter specification:  
## confidence minval smax arem aval originalSupport support minlen maxlen  
## 0.5 0.1 1 none FALSE TRUE 0.05 1 10  
## target ext  
## rules FALSE  
##   
## Algorithmic control:  
## filter tree heap memopt load sort verbose  
## 0.1 TRUE TRUE FALSE TRUE 2 TRUE  
##   
## Absolute minimum support count: 4408   
##   
## set item appearances ...[0 item(s)] done [0.00s].  
## set transactions ...[16470 item(s), 88162 transaction(s)] done [0.16s].  
## sorting and recoding items ... [6 item(s)] done [0.01s].  
## creating transaction tree ... done [0.04s].  
## checking subsets of size 1 2 3 done [0.00s].  
## writing ... [15 rule(s)] done [0.00s].  
## creating S4 object ... done [0.01s].

inspect(grdar)

## lhs rhs support confidence lift   
## 1 {} => {39} 0.57479413 0.5747941 1.0000000  
## 2 {38} => {48} 0.09010685 0.5093614 1.0657723  
## 3 {38} => {39} 0.11734080 0.6633111 1.1539977  
## 4 {32} => {48} 0.09112770 0.5297026 1.1083338  
## 5 {32} => {39} 0.09590300 0.5574603 0.9698434  
## 6 {41} => {48} 0.10228897 0.6034125 1.2625621  
## 7 {41} => {39} 0.12946621 0.7637337 1.3287082  
## 8 {48} => {39} 0.33055058 0.6916340 1.2032726  
## 9 {39} => {48} 0.33055058 0.5750765 1.2032726  
## 10 {38,48} => {39} 0.06921349 0.7681269 1.3363513  
## 11 {38,39} => {48} 0.06921349 0.5898502 1.2341847  
## 12 {32,48} => {39} 0.06127356 0.6723923 1.1697968  
## 13 {32,39} => {48} 0.06127356 0.6389119 1.3368399  
## 14 {41,48} => {39} 0.08355074 0.8168108 1.4210493  
## 15 {39,41} => {48} 0.08355074 0.6453478 1.3503063

* **Now that you have rules:**
* **Find a few interesting rules**
* **Tell me something you learned from interpreting the rules**
* **Show all your steps (especially in data conversion) using knitR**

According to me, the interesting rules that we obtain by running apriori on the data set are: + if we consider that item 39 is present in the list then 48 will also be present (roughly 57.5% and we are 1.2 times better at determining this). + Similarly, if 48 is present in the list then 39 will also be present (roughly 69.1% and we are 1.2 times better at determining this). If we consider pairs then the interesting pairs would be: + {38,48} which helps us determine that 39 will also be present (roughly 76.8% of the time and we are 1.33 times better at determining this) + {41,48} which helps us determine that 39 will also be present (roughly 81.7% of the time and we are 1.4 times better at determining this) + {39,41} which helps us determine that 48 will also be present (roughly 64.5% of the time and we are 1.3 times better at determining this)

We can also see that the rules for the individual items is in a way reflected in the pair rules as the items we predict to also occur are either 39 or 48. We can obtain the important rules in R as follows:

subrules<-grdar[quality(grdar)$confidence > 0.6]  
inspect(subrules)

## lhs rhs support confidence lift   
## 3 {38} => {39} 0.11734080 0.6633111 1.153998  
## 6 {41} => {48} 0.10228897 0.6034125 1.262562  
## 7 {41} => {39} 0.12946621 0.7637337 1.328708  
## 8 {48} => {39} 0.33055058 0.6916340 1.203273  
## 10 {38,48} => {39} 0.06921349 0.7681269 1.336351  
## 12 {32,48} => {39} 0.06127356 0.6723923 1.169797  
## 13 {32,39} => {48} 0.06127356 0.6389119 1.336840  
## 14 {41,48} => {39} 0.08355074 0.8168108 1.421049  
## 15 {39,41} => {48} 0.08355074 0.6453478 1.350306

The above results in eliminating one rule from the assumption (namely if 39 then 48).

Interpreting the rules requires that we have a good idea of what value of Confidence we choose for determining the important rules. We must also be mindful of the Lift so as to increase the success of guessing the other items.

* **Next, tell what you would like to do next with the retail data**
* **Is there a hypothesis you would like to test?**
* **Is there data from another source you would like to add?**
* **Is there a predictive model you would like to build?**
* **Anything else?**
* **Tell me what would be interesting to you to do next**

With regards to the retail data we pulled up, it would be interesting to add data such as the cost of the items and if possible the names of the item as well. Other data that could be included would be the age group of the purchaser (if possible), the gender of the purchaser, the quantity of the items purchased and maybe even the average income of the household.  
 If such data could be included then it would be interesting to test hypothesis along the lines of a purchaser of a particular age group purchasing a particular item (or set of items). We might then be able to use association analysis to roughly predict what other items the purchaser would most likely be interested in or would buy along with a given item.  
 We could build a predictive model which could predict the products a person of a particular age group would purchase or a predictive model which could predict the average expenditure of a person on groceries over a week given the household income.  
 The next step (according to me), would be to test out these rules or to perform regression analysis to see if there is any actual relation between the various items. This might help us better understand the rules we obtained and make better assumptions on how items are sold.

* **Tell me about the project you would like to do with Association Analysis**
* **It can be a project at work**
* **Or, suppose you could download data from data.gov on healthcare, or education, or whatever**
* **What would you like to do Assocation Analysis on if you could**

A data set I would like to perform association analysis on would be the Commodity flow survey data set which includes information about the commodities shipped, their values, weight and mode of transportation, etc.  
 I would like to see if given a product we can predict the mode of transport and possibly other products that would be shipped along with it to a particular place or if we are given a mode of transport we could predict the other products that were shipped along with it. Other questions could include given a temperature control value, can we predict the products shipped or the transport mode used.