Market Basket Analysis

Ranjit Mishra

November 08, 2018

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see [http://rmarkdown.rstudio.com](http://rmarkdown.rstudio.com/).

I am using LastFM dataset to demostrate music recommendation based on association rules of Market Basket Analysis. LastFM dataset is widely available on various websites, hence not including the actual data. One needs to download and read the data as 'lastfm' before executing the below piece of code:

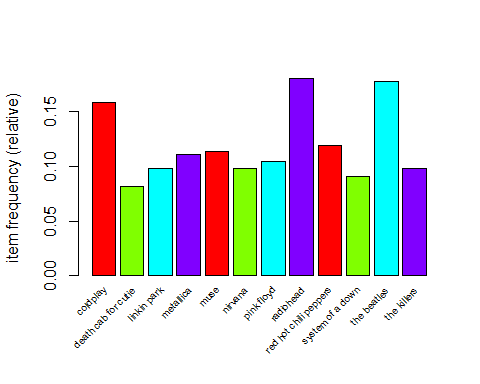
head(lastfm)

## user artist sex country  
## 1 1 red hot chili peppers f Germany  
## 2 1 the black dahlia murder f Germany  
## 3 1 goldfrapp f Germany  
## 4 1 dropkick murphys f Germany  
## 5 1 le tigre f Germany  
## 6 1 schandmaul f Germany

## Turn the "user" column into a factor  
lastfm$user <- factor(lastfm$user)  
  
## Load library arules for executing market basket analysis. The below piece of code does the followings -   
# 1. Splits values containing in "artists" column into groups by the "user"  
# 2. Removes duplicates from playlists  
# 3. Use the "transactions" data class to create a dataset of transactions.   
  
library(arules)

## Loading required package: Matrix  
##   
## Attaching package: 'arules'  
##   
## The following objects are masked from 'package:base':  
##   
## %in%, write

playlist <- split(x=lastfm[,"artist"],f=lastfm$user)  
  
playlist <- lapply(playlist,unique)  
  
playlist <- as(playlist,"transactions")  
  
# Make a frequency plot of the transactions with a support of 0.08 or greater. We can find out the the most popular artists. The below plot shows that the 3 most popular artists are coldplay, radiohead and the beatles.   
  
itemFrequencyPlot(playlist, support = .08, cex.names = .6, col = rainbow(4))



# Let's now create the association rules by using the apriori function with a support of 0.01 and confidence of 0.45.   
  
musicrules <- apriori(playlist,parameter=list(support=.01, confidence=.45))

##   
## Parameter specification:  
## confidence minval smax arem aval originalSupport support minlen maxlen  
## 0.45 0.1 1 none FALSE TRUE 0.01 1 10  
## target ext  
## rules FALSE  
##   
## Algorithmic control:  
## filter tree heap memopt load sort verbose  
## 0.1 TRUE TRUE FALSE TRUE 2 TRUE  
##   
## apriori - find association rules with the apriori algorithm  
## version 4.21 (2004.05.09) (c) 1996-2004 Christian Borgelt  
## set item appearances ...[0 item(s)] done [0.00s].  
## set transactions ...[1004 item(s), 15000 transaction(s)] done [0.03s].  
## sorting and recoding items ... [655 item(s)] done [0.01s].  
## creating transaction tree ... done [0.00s].  
## checking subsets of size 1 2 3 4 done [0.05s].  
## writing ... [120 rule(s)] done [0.00s].  
## creating S4 object ... done [0.00s].

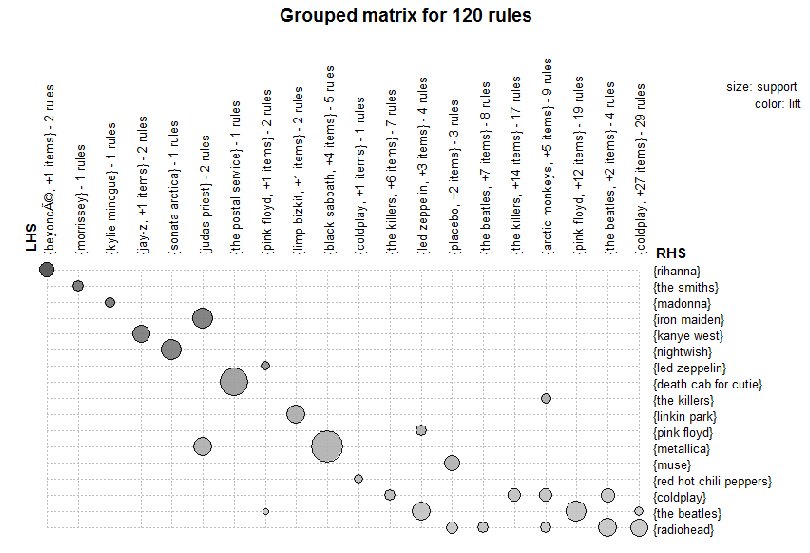
# We can use 'inspect' function to output the subset with a lift of 6 or more. We can also see Which rules have the highest confidence  
  
inpect6sort <- inspect(sort(subset(musicrules, subset=lift > 6), by="confidence"))

## lhs rhs support confidence lift  
## 1 {the pussycat dolls} => {rihanna} 0.01040000 0.5777778 13.415893  
## 2 {t.i.} => {kanye west} 0.01040000 0.5672727 8.854413  
## 3 {pink floyd,   
## the doors} => {led zeppelin} 0.01066667 0.5387205 6.802027  
## 4 {sonata arctica} => {nightwish} 0.01346667 0.5101010 8.236292  
## 5 {judas priest} => {iron maiden} 0.01353333 0.5075000 8.562992  
## 6 {jay-z} => {kanye west} 0.01506667 0.4967033 7.752913  
## 7 {kylie minogue} => {madonna} 0.01093333 0.4781341 8.757035  
## 8 {beyoncÃ©} => {rihanna} 0.01393333 0.4686099 10.881034  
## 9 {morrissey} => {the smiths} 0.01126667 0.4655647 8.896141

# Let's now load arulesViz package to visualize the associations of all the rules  
  
library(arulesViz)

## Loading required package: grid  
##   
## Attaching package: 'arulesViz'  
##   
## The following object is masked from 'package:base':  
##   
## abbreviate

plot(musicrules, method = "grouped", control = list(k= 20))

# By default number of groups is 20, hence the above plot is showing only 20 groups. It can be increased or decreased based on what needs to be analyzed. The group of most important rules according to lift are shown above. For example, there are 2 rules containing 'beyonce' and 1 more item in the LHS (or the antecedent) and the RHS (or the consequent) is 'rihanna'. It seems the lift is higher for the consequent – 'rihanna' and antecedent - 'beyonce' (because the circle is the darkest in color). Similarly, rest of the rules should be analyzed to exttract insights.   
  
# Scatter plot can also be created to visualize association rules and itemsets. Try using "T" for interactive parameter below :)

plot(musicrules, method = NULL, measure = "support", shading = "lift", interactive = F)

