Association Rule Example: Online radio music recommendation

Online radio keeps track of everything you play. It uses this information for recommending music you are likely to enjoy and supports focused marketing that sends you advertisements for music you are likely to buy. For each user you may have a log of every artist he/she had downloaded to his/her computer.

The original data set we obtain is from last.fm online radio, with user ID as the first column, artistes preferred in the second column, gender in the third column and country in the fourth column. Each user (with the same ID) may have multiple records.

After some preprocessing, we got practice\_1\_dataset.csv is available for you. Only preferred artists were kept in the data. A row in the dataset shows preferred artists of a user. The data consist of 14593 rows, which are 14593 users music preference.

1. Load the music data into a sparse matrix. Since each user may mention one artist multiple times, we set “rm.duplicates=TRUE” when we load the data into a sparse matrix.

**Code:**

install.packages('arules')

library(arules)

music <- read.transactions(file.choose(), sep = ",", rm.duplicates=TRUE)

1. Use summary () function to inspect the sparse matrix. How many users and columns do this matrix have? What does density mean? Who is the most frequent artist/item? How many rows/users have 30 items/artists?

**Ans:**

**#users: 14593 & columns: 1004**

**#Density tells us percent of non-zero cells in sparse matrix**

**#Radiohead**

**#340**

**Code:**

**summary(music)**

1. Inspect the first 5 users’ artists. Does the records match the first five rows of the original csv file? Use itemFrequency to check the proportion of users liking certain artists. What are the proportions of users’ favorite artists in column from 4 to 7?

**Ans:**

**#Yes**

**#**

**3 doors down :: 3.17%**

**30 seconds to mars :: 3.37%**

**311 :: 0.84%**

**36 crazyfists :: 0.80%**

**Code:**

**inspect(music[1:5])**

**itemFrequency(music[, 1:5])**

**itemFrequency(music[, 4:7])**

1. Create a histogram plotting the artists have more than 10 percent support. How many artists in the matrix have at least 10 percent support? Who are these artists? Plot the first 20 artists with highest support. Which artist has the 15th highest support?

**Ans:**

**#10**

**#** **coldplay; linkin park, metallica, muse, nirvana, pink flyod, radiohead, red hot chili pepper, the beatles, the killers**

**#** **placebo**

Chart, bar chart

Description automatically generated

Chart, histogram

Description automatically generated

**Code:**

**itemFrequencyPlot(music, support = 0.1)**

**itemFrequencyPlot(music, topN = 20)**

1. Generate a visualization of the sparse matrix for the first 100 users' preference. Then generate a visualization of a random sample of 500 users' music favorite.

**Ans:**

A picture containing diagram

Description automatically generated

A picture containing qr code

Description automatically generated

**Code:**

**image(music[1:100])**

**image(sample(music, 500))**

1. Use apriori() function to train the association rules on the music data. How many rules do we have when we use the default setting? In order to learn more rules, we adjust support level to 0.01, minlen =2 and confidence level to 0.25. How many rules do we have then?

**Ans:**

**#0**

**#788 rules**

**Code:**

**apriori(music)**

**musicrules <- apriori(music, parameter = list(support =**

**0.01, confidence = 0.25, minlen = 2))**

**musicrules**

1. Summarize the rules generated from adjusted parameters. How many rules have size 3 among all the rules? Check the first ten rules. If a user likes james blunt, which artist else should online radio recommend to this user? Sort the rule list by lift and inspect the first five rules with highest lift. Which rule has the fourth highest lift?

**Ans:**

**#**224 rules

#If a user likes james blunt then the online radio should recommend {nas} to this user.

#{beyoncc} => {rihanna} has the fourth highest lift.

**Code:**

**summary(musicrules)**

**inspect(musicrules[1:10])**

**inspect(sort(musicrules, by = "lift")[1:5])**

1. Find subsets of rules containing any cold play. How many rules have cold play? Sort the rules by support and inspect the first five cold play rules with highest support. What rule has the 2nd highest support?

**Ans:**

**#** **172 rules**

**#** **{radiohead} => {coldplay} 0.05612280 0.3029967 0.1852258 1.860956 819**

**Code:**

coldrules <- subset(musicrules, items %in% "coldplay")

inspect(coldrules)

inspect(sort(coldrules, by = "support")[1:5])

1. You can write these rules to a file and save the rules into a data frame. How many variables are in this saved data frame?

**Ans:**

#788 obs. of 6 variables

**Code:**

write(musicrules, file = "musicrules.csv",

sep = ",", quote = TRUE, row.names = FALSE)

musicrules\_df <- as(musicrules, "data.frame")

str(musicrules\_df)