Kiriakos\_Week 5 ARules for Association Analysis

Cathy Kiriakos

April 6, 2020

## ARules for Association Analysis

For this analysis I’ve downloaded the bxBooks data set from Git Hub at <https://github.com/WinVector/zmPDSwR/blob/master/Bookdata/create_bookdata.R>.

According to details at <http://www2.informatik.uni-freiburg.de/~cziegler/BX/> the books are identified by their respective ISBN. Invalid ISBNs have already been removed from the data set. Moreover, some content-based information is given (Book-Title, Book-Author, Year-Of-Publication, Publisher), obtained from Amazon Web Services. Note that in case of several authors, only the first is provided. URLs linking to cover images are also given, appearing in three different flavours (Image-URL-S, Image-URL-M, Image-URL-L), i.e.,

## 1) Exploratory data analysis:

We will first start with gaining some insight into the structure of the data set:

str(bxBooks)

So we can see that we have a data frame with 8 variables, ISBN, Book Title, Book Author, Publisher, Imagage URL.S, Image URL.M, Imager URL.L. For the purposes of this analysis, we know that the images are irrelevant and can be dropped.

bxBooks2 <-subset(bxBooks, select = -c(6:8))  
str(bxBooks2)

Now we can load the appropriate libraries for our analysis:

Now we can leverage dplyr to get an understanding of our most popular books grouping by ISBN and the number of our observations n:

topBooks <- bxBookRatings %>% group\_by(ISBN) %>% summarise(n=n()) %>%  
 top\_n(n=20) %>% arrange(n)  
topBooks

We will start by pulling together the needed data points, merging our topBooks dataframe with bxBooks; ISBN and Book ratings providing us with a complete picture so that we can build our histogram covering the Top 20 rated books below:

topBooks<-merge(topBooks,bxBooks, by = "ISBN")  
head(topBooks)

topBooks <-subset(topBooks, select = -c(6:8))

Quick view of cleaned dataframe:

head(topBooks)

Now we can get a visual of our Top 20 rated books leveraging ggplot as shown below:

ggplot(topBooks,aes(x=reorder(Book.Title,n),n))+ geom\_bar(stat='identity')+ theme(axis.text.x=element\_text(angle=90, hjust=1))+ coord\_flip() + labs(x = "Book Title",y="Number of Ratings")

## 2) Use apriori() algorithm to display the top ten rules (sort by confidence), and summary information:

Set the minimum support to 0.005, and minimum confidence of 0.70.

For the sake of time, I will reduce the data set down to 50,000 (135k was causing R to run our of allocated memory) records before transforming the data set into transaction format, as shown below:

subsetBooks<-bxBookRatings[sample(nrow(bxBookRatings), 50000), ]

Ok so now we can bring in the needed additional information back into our subsetted dataframe:

subsetBooks <- merge(subsetBooks,bxBooks, by = "ISBN")

Now I will transform the required columns into transaction format as shown below:

subsetBooks$User.ID <- as.factor(subsetBooks$User.ID)  
subsetBooks$ISBN <- as.factor(subsetBooks$ISBN)  
subsetBooks$Book.Rating <- as.factor(subsetBooks$Book.Rating)  
subsetBooks$Book.Title <- as.factor(subsetBooks$Book.Title)  
subsetBooks$Book.Author <- as.factor(subsetBooks$Book.Author)  
subsetBooks$Year.Of.Publication <- as.factor(subsetBooks$Year.Of.Publication)  
subsetBooks$Publisher <- as.factor(subsetBooks$Publisher)

So we can create the association rules on our subset as shown below:

rules1 <- apriori(subsetBooks, parameter = list(support = 0.005, confidence = 0.70))  
summary(rules1)

Now we can get a look at our rules by calling inspect:

inspect(rules1)

inspect(sort(rules1, by="confidence", decreasing=TRUE))

## 3) Adjusting results:

Due to our default confidence intervals we didn’t get very exciting results, so we can start by adjusting our support and confidence intervals. We will also incorporate a max length of rules into this algorithm.

rules2 <- apriori(subsetBooks, parameter = list(support = 0.01, confidence = 0.80, maxlen = 3))  
summary(rules2)

inspect(sort(rules2, by="confidence", decreasing=TRUE))

So we can get a look at this data using a minimum length, on a subset of 20

rules.sorted <- sort(rules2,by = "lift")  
inspect(rules.sorted[1:20])

So from our updated sorted results on ratings, we can see that Harlequin, Silhouette, and Avon has the highest number of ratings, and that 1988 was a popular year.

So this makes me wonder what made 1988 popular, we will investigate this relationship:

rules1988 <- apriori(subsetBooks,parameter = list(supp =0.001, conf = 0.08), appearance = list(default = "rhs", lhs = "Year.Of.Publication=1988"), control = list(verbose=F))  
rules\_conf <- sort (rules1988, by="confidence", decreasing=TRUE)  
inspect(head(rules\_conf))

## 4) Use library(aruleViz) to display rules:

So now we can get a visualization on our final set of 1988 rules:

plot(rules1988)

For a comparison against our first set of generated rules, we can see the variance below, showing a drastic difference of relationship between the confidence and lift comparing our first analysis, and subsetted 1988 results.

plot(rules1)

Lastly, lets compare against our 2nd set of sorted rules below:

plot(rules.sorted)

## 5) Discuss any findings from your experiment results. Are there any rules that surprise you?

From our analysis we can see that we have some favorite publishers associated with ratings including Harlequin, Silhouette, and Avon has the highest number of ratings, and that 1988 was a popular year. When investigating further we found that ratings for books written in 1988 were also associated with the year 2002.

## 6) Recommended actions:

Recommended actions for books in this subset of data would be to group the publishers Harlequin, Silhouette, and Avon together. An additional suggestion would associate the book titles from 1988 and 2002 so that you could find the similarities between ratings and the qualities or types of books from those years to advertise more effectively to the masses.