R Programming – Association Rules in R Markdown

Timothy McCombs

The catalog company Exeter, Inc. sells products to customers via its distribution of several catalogs that have a variety of subject categories. The company’s business goal is to optimize its profits by controlling printing and distribution costs while maximizing sales via improved marketing strategies. A potential opportunity to increase sales lies in cross-selling or enticing a customer to buy additional products given the combination of products already in their shopping cart. The company has provided a data set of customers with purchasing histories in a given set of catalog categories. The categories consist of Clothing, Housewares, Health, Automotive, Personal Electronics, Computers, Garden, Novelty Gifts, and Jewelry. The goal of the project is to identify strong associations between products already purchased by customers so that Exeter, Inc. may market these cross-selling opportunities.

## Program Code and Plots

**Figure 1**  
*Load Libraries and Import Data Set*

# Load appropriate libraries for the project.  
library(Matrix)  
library(arules)

##   
## Attaching package: 'arules'

## The following objects are masked from 'package:base':  
##   
## abbreviate, write

library(arulesViz)  
library(tidyverse)

## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5 v purrr 0.3.4  
## v tibble 3.1.2 v dplyr 1.0.7  
## v tidyr 1.1.3 v stringr 1.4.0  
## v readr 1.4.0 v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x tidyr::expand() masks Matrix::expand()  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()  
## x tidyr::pack() masks Matrix::pack()  
## x dplyr::recode() masks arules::recode()  
## x tidyr::unpack() masks Matrix::unpack()

# Create the pp.ccs data frame by reading the comma separated values file.  
pp.ccs <- read.csv("CatalogCrossSell.csv", header = TRUE)

**Figure 2**  
*Initial Exploration of Data*

# Display the first 6 rows of records and all columns in the data frame.  
head(pp.ccs)

## Customer.Number Clothing.Division Housewares.Division  
## 1 11569 0 1  
## 2 13714 0 1  
## 3 46391 0 1  
## 4 67264 0 0  
## 5 67363 0 0  
## 6 72553 0 1  
## Health.Products.Division Automotive.Division Personal.Electronics.Division  
## 1 1 1 1  
## 2 1 1 1  
## 3 1 1 1  
## 4 1 1 1  
## 5 1 0 1  
## 6 1 1 1  
## Computers.Division Garden.Division Novelty.Gift.Division Jewelry.Division  
## 1 0 0 1 0  
## 2 0 1 1 1  
## 3 0 1 1 1  
## 4 0 1 1 0  
## 5 0 1 1 0  
## 6 0 1 1 1

# Display the structure of the data frame which provides the number  
# of variables and rows as well as the variable types.  
str(pp.ccs)

## 'data.frame': 4998 obs. of 10 variables:  
## $ Customer.Number : int 11569 13714 46391 67264 67363 72553 79814 80903 91439 96701 ...  
## $ Clothing.Division : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ Housewares.Division : int 1 1 1 0 0 1 1 1 0 1 ...  
## $ Health.Products.Division : int 1 1 1 1 1 1 1 1 1 1 ...  
## $ Automotive.Division : int 1 1 1 1 0 1 0 0 1 1 ...  
## $ Personal.Electronics.Division: int 1 1 1 1 1 1 1 1 1 1 ...  
## $ Computers.Division : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ Garden.Division : int 0 1 1 1 1 1 1 0 1 1 ...  
## $ Novelty.Gift.Division : int 1 1 1 1 1 1 0 1 0 1 ...  
## $ Jewelry.Division : int 0 1 1 0 0 1 0 0 1 1 ...

**Figure 3**  
*Dimension Reduction and Conversion to Transaction Data*

# Convert data frame to a matrix format, and remove columns one and four.  
pp.ccs <- as.matrix(pp.ccs[,-c(1,4)])  
  
# Create a new pp.ccs.trans matrix and convert variable types to transaction data.  
pp.ccs.trans <- as(pp.ccs,"transactions")  
  
# List column (variable) names to verify integrity.  
colnames(pp.ccs.trans)

## [1] "Clothing.Division" "Housewares.Division"   
## [3] "Automotive.Division" "Personal.Electronics.Division"  
## [5] "Computers.Division" "Garden.Division"   
## [7] "Novelty.Gift.Division" "Jewelry.Division"

# Display summary information of new matrix to verify itemMatrix in sparse format.  
# Also displays most frequent items used.  
summary(pp.ccs.trans)

## transactions as itemMatrix in sparse format with  
## 4998 rows (elements/itemsets/transactions) and  
## 8 columns (items) and a density of 0.2415216   
##   
## most frequent items:  
## Personal.Electronics.Division Housewares.Division   
## 2336 1967   
## Jewelry.Division Garden.Division   
## 1784 1360   
## Novelty.Gift.Division (Other)   
## 1137 1073   
##   
## element (itemset/transaction) length distribution:  
## sizes  
## 0 1 2 3 4 5 6 7 8   
## 1002 1358 1022 759 464 264 106 18 5   
##   
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.000 1.000 2.000 1.932 3.000 8.000   
##   
## includes extended item information - examples:  
## labels  
## 1 Clothing.Division  
## 2 Housewares.Division  
## 3 Automotive.Division

# Run structure command for pp.ccs.trans to verify data is in appropriate formats.  
str(pp.ccs.trans)

## Formal class 'transactions' [package "arules"] with 3 slots  
## ..@ data :Formal class 'ngCMatrix' [package "Matrix"] with 5 slots  
## .. .. ..@ i : int [1:9657] 1 2 3 6 1 2 3 5 6 7 ...  
## .. .. ..@ p : int [1:4999] 0 4 10 16 20 23 29 32 35 39 ...  
## .. .. ..@ Dim : int [1:2] 8 4998  
## .. .. ..@ Dimnames:List of 2  
## .. .. .. ..$ : NULL  
## .. .. .. ..$ : NULL  
## .. .. ..@ factors : list()  
## ..@ itemInfo :'data.frame': 8 obs. of 1 variable:  
## .. ..$ labels: chr [1:8] "Clothing.Division" "Housewares.Division" "Automotive.Division" "Personal.Electronics.Division" ...  
## ..@ itemsetInfo:'data.frame': 0 obs. of 0 variables

**Figure 4**  
*Apriori Model and Association Rules*

# Run association rules model Apriori  
pprules <- apriori(pp.ccs.trans,parameter = list(supp = 0.05, conf = 0.5,target = "rules"))

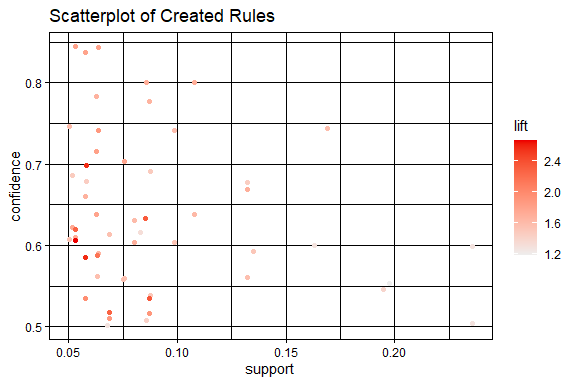
## Apriori  
##   
## Parameter specification:  
## confidence minval smax arem aval originalSupport maxtime support minlen  
## 0.5 0.1 1 none FALSE TRUE 5 0.05 1  
## maxlen target ext  
## 10 rules TRUE  
##   
## Algorithmic control:  
## filter tree heap memopt load sort verbose  
## 0.1 TRUE TRUE FALSE TRUE 2 TRUE  
##   
## Absolute minimum support count: 249   
##   
## set item appearances ...[0 item(s)] done [0.00s].  
## set transactions ...[8 item(s), 4998 transaction(s)] done [0.00s].  
## sorting and recoding items ... [6 item(s)] done [0.00s].  
## creating transaction tree ... done [0.00s].  
## checking subsets of size 1 2 3 4 done [0.00s].  
## writing ... [54 rule(s)] done [0.00s].  
## creating S4 object ... done [0.00s].

# Limit significant digits to three with options function.  
options(digits = 3)  
  
# List the top five rules identified from the model by descending lift.  
inspect(head(sort(pprules, by = "lift"), n = 5))

## lhs rhs support confidence coverage lift count  
## [1] {Personal.Electronics.Division,   
## Garden.Division,   
## Jewelry.Division} => {Novelty.Gift.Division} 0.0532 0.606 0.0878 2.66 266  
## [2] {Housewares.Division,   
## Personal.Electronics.Division,   
## Garden.Division} => {Novelty.Gift.Division} 0.0576 0.585 0.0984 2.57 288  
## [3] {Automotive.Division,   
## Personal.Electronics.Division} => {Garden.Division} 0.0580 0.699 0.0830 2.57 290  
## [4] {Personal.Electronics.Division,   
## Garden.Division} => {Novelty.Gift.Division} 0.0872 0.534 0.1633 2.35 436  
## [5] {Automotive.Division} => {Garden.Division} 0.0854 0.634 0.1349 2.33 427

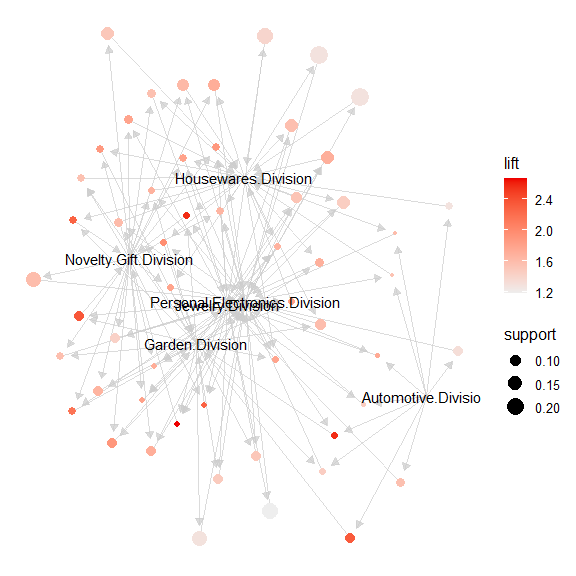
**Figure 5**  
*Scatter Plot of Created Rules*

# Scatter plot of created rules.  
plot(pprules, jitter = 0, main = "Scatterplot of Created Rules")



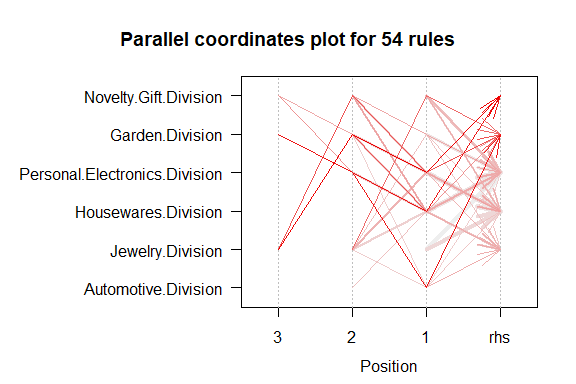
**Figure 6**  
*Graph of Created Rules*

# Graph of created rules.  
plot(pprules, method = "graph")



**Figure 7**  
*Parallel Coordinates of Created Rules*

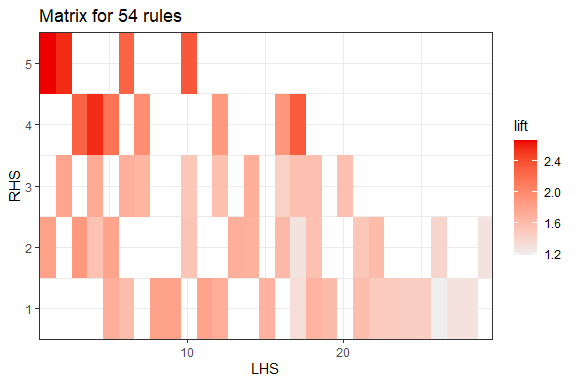
# Parallel coordinate plot of created rules.  
plot(pprules, method = "paracoord", control = list(reorder = TRUE))



**Figure 8**  
*Heat Matrix of Created Rules*

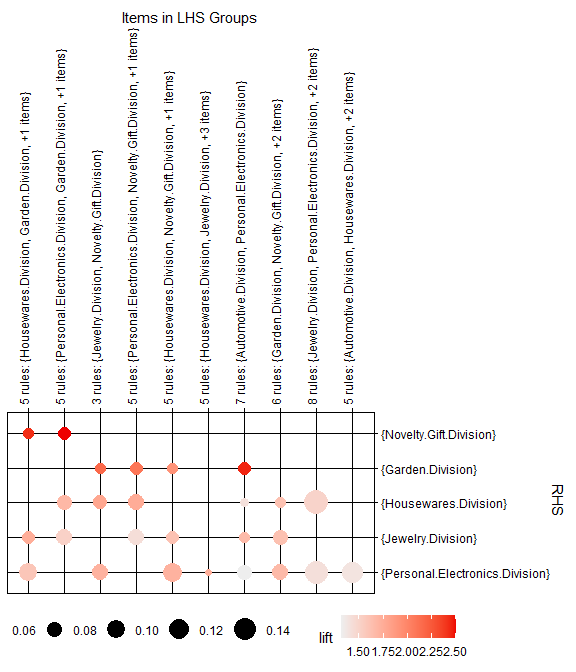
# Heat matrix of created rules.  
plot(pprules, method = "matrix")

## Itemsets in Antecedent (LHS)  
## [1] "{Personal.Electronics.Division,Garden.Division,Jewelry.Division}"   
## [2] "{Housewares.Division,Personal.Electronics.Division,Garden.Division}"   
## [3] "{Personal.Electronics.Division,Novelty.Gift.Division,Jewelry.Division}"   
## [4] "{Automotive.Division,Personal.Electronics.Division}"   
## [5] "{Novelty.Gift.Division,Jewelry.Division}"   
## [6] "{Housewares.Division,Garden.Division}"   
## [7] "{Housewares.Division,Personal.Electronics.Division,Novelty.Gift.Division}"  
## [8] "{Garden.Division,Novelty.Gift.Division,Jewelry.Division}"   
## [9] "{Housewares.Division,Novelty.Gift.Division,Jewelry.Division}"   
## [10] "{Personal.Electronics.Division,Garden.Division}"   
## [11] "{Housewares.Division,Garden.Division,Novelty.Gift.Division}"   
## [12] "{Housewares.Division,Novelty.Gift.Division}"   
## [13] "{Personal.Electronics.Division,Jewelry.Division}"   
## [14] "{Personal.Electronics.Division,Garden.Division,Novelty.Gift.Division}"   
## [15] "{Housewares.Division,Garden.Division,Jewelry.Division}"   
## [16] "{Personal.Electronics.Division,Novelty.Gift.Division}"   
## [17] "{Automotive.Division}"   
## [18] "{Garden.Division,Novelty.Gift.Division}"   
## [19] "{Housewares.Division,Automotive.Division}"   
## [20] "{Housewares.Division,Personal.Electronics.Division}"   
## [21] "{Novelty.Gift.Division}"   
## [22] "{Garden.Division,Jewelry.Division}"   
## [23] "{Automotive.Division,Jewelry.Division}"   
## [24] "{Automotive.Division,Garden.Division}"   
## [25] "{Housewares.Division,Jewelry.Division}"   
## [26] "{Jewelry.Division}"   
## [27] "{Garden.Division}"   
## [28] "{Housewares.Division}"   
## [29] "{Personal.Electronics.Division}"   
## Itemsets in Consequent (RHS)  
## [1] "{Personal.Electronics.Division}" "{Housewares.Division}"   
## [3] "{Jewelry.Division}" "{Garden.Division}"   
## [5] "{Novelty.Gift.Division}"



**Figure 9**  
*Grouped Matrix of Top Ten Antecedents by Lift*

# Grouped matrix plot.  
# Use k parameter to limit number of antecedent groups.  
plot(pprules, method = "grouped", k = 10)



## Description, Visualizations, and Interpretation

To start, the appropriate libraries are loaded to R Studio for the project (Figure 1). *Arules* and *arulesViz* are the key libraries providing association rule processes and visualizations. The comma-separated values data set provided by the company is read and made into a data frame. Some initial data exploration (Figure 2) using the head and str functions reveal a data set of ten variables (attributes) and 4,998 rows or observations. The initial column or variable is the customer number that will not be needed for market basket analysis, therefore it will need to be removed. The rest of the columns are integers of one or zero. The categories are further described as the following: *Customer.Number, Clothing.Division, Housewares.Division, Health.Products.Division, Automotive.Division, Personal.Electronics.Division, Computers.Division, Garden.Division, Novelty.Gift.Division*, and *Jewelry.Division*. These columns can be changed to transaction data. To verify there is varied results in each column, the original *.csv* file was explored in Microsoft Excel using column filtering. The *Health.Products.Division* category column has a value of one for all rows. This variable (column four) will not provide any insight and will also be removed before analysis.

The data frame is converted to a matrix format and columns one and four (*Customer.Number* and *Health.Products.Division*) are removed (Figure 3). Also, the data are transformed into transaction data and named *pp.ccs.trans*. As a check, Colnames and summary are used to verify the correct columns and rows remain and str verifies data is now transaction data. The transaction data is now ready for the model.

The commonly used Apriori algorithm will be selected to identify associations (Figure 4). A minimum confidence value of 0.5 (50%) is used to ensure the associations identified have significant weight. After testing a range of minimum support values, a 0.05 setting is used to create a manageable list of 54 rules. Model results are rounded to three decimal places. The results are sorted by decreasing lift with the top five results listed at the bottom of Figure 4. These rules can be interpreted as follows:

**1. If items are purchased in the *Personal.Electronics.Division, Garden.Division*, and *Jewelry.Division*, then with confidence 60.6% *Novelty.Gift.Division* items will also be purchased. This rule has a lift ratio of 2.66.**  
**2. If items are purchased in the *Housewares.Division, Personal.Electronics.Division*, and *Garden.Division*, then with confidence 58.5% *Novelty.Gift.Division* items will also be purchased. This rule has a lift ratio of 2.57.**  
**3. If items are purchased in the *Automotive.Division* and *Personal.Electronics.Division*, then with confidence 69.9% *Garden.Division* items will also be purchased. This rule has a lift ratio of 2.57.**  
**4. If items are purchased in the *Personal.Electronics.Division* and *Garden.Division*, then with confidence 53.4% *Novelty.Gift.Division* items will also be purchased. This rule has a lift ratio of 2.35.**  
**5. If items are purchased in the *Automotive.Division*, then with confidence 63.4% *Garden.Division* items will also be purchased. This rule has a lift ratio of 2.33.**

### Visualizations

Figure 5 is a scatter plot graph plotting the created rules in relation to support and confidence with lift ratio intensity. It is noted that rules with support greater than 0.10 do not have a significant lift. This would indicate there is no single association that will yield a significant marketing windfall. It will be the combination of smaller support, higher lift marketing associations that will add up. A mapped graph of the created association rules in Figure 6 is a little coagulated, however, it can be seen that the *Garden.Division, Personal.Electronics.Division, Jewelry.Division*, and *Housewares.Division* categories are involved with the majority of created rules. This is supported in the parallel coordinates plot displayed in Figure 7, with many association lines converging on these divisions. Figure 8 is a heat matrix of the created rules with left-hand side antecedents (LHS) on the x-axis and right hand-side consequents (RHS) on the y-axis. The heat map demonstrates cross-marketing novelty gifts and gardening items (darker red shades for RHS 4 and 5) with the other divisions will provide the highest lift ratios. Lastly, Figure 9 lists the top ten grouped antecedents by lift (LHS groups) paired with respective consequents listed to the right (RHS) on the y-axis. The RHS rules are sorted by descending lift values so that the graph displays the strongest rules by lift in the upper left. Again, novelty gifts and garden division items show the strong lift associations.

### Interpretation Summary

The analysis model suggests many cross-purchasing associations among the *Garden.Division, Personal.Electronics.Division, Housewares.Division*, and *Jewelry.Division* catalogs. These may be good targets to focus much of the marketing budget. As for what items to cross-market, novelty gift items and gardening items indicate the best lift or probability of subsequent purchasing over mere random occurrence. Conversely, the *Computer.Division* appears to be a silo-ed, niche market that does not warrant cross-marketing investment.

## References:

Colorado State University Global (CSUG). (2021). *Module 6: Relationships in unsupervised learning* [Interactive lecture]. Canvas. <https://portal.csuglobal.edu>.

RStudio Team. (2021). *RStudio: Integrated development environment for R. RStudio*. PBC, Boston, MA. <http://www.rstudio.com/>.

Shmueli, G., Bruce, P.C., Yahav, I., Patel, N.R., and Lichtendahl, K.C. (2018). *Data mining for business analytics: Concepts, techniques, and application sin R.* Wiley Publishing. ISBN: 9781118879337.