MIS 510 Portfolio Project Option 2

Jason Kehl

11/25/2019

For this analysis, I will utilize the first five steps of the CRISP-DM process to achieve the assignment goals:

1. Understand business
2. Understand data (explore data)
3. Prep data for modeling
4. Build model
5. Evaluate model
6. Deploy model

1. Understand business

The data set has 9 categories of catalogs, and shows which customers have purchased items from specific catalog categories. I need to evaluate the data to ascertain which customers are most likely to buy from which categories, based on their previous purchase history.

2. Understand data (explore data)

Data exploration functions (6 included)

```
library(DataExplorer) # Convenient data exploration functions
library(arulesViz)    # Association rule-specific visualizations

# Read in the data set
crossSell.df = read.csv("../datasets/CatalogCrossSell.csv")

# Explore different data summary views
dim(crossSell.df)

## [1] 60004    20

summary(crossSell.df)

##  Customer.Number     Clothing.Division Housewares.Division
##  Min.   :    11569   Min.   :0.00      Min.   :0.00
##  1st Qu.:122609958   1st Qu.:0.00      1st Qu.:0.00
##  Median :213445258   Median :0.00      Median :0.00
##  Mean   :189863280   Mean   :0.03      Mean   :0.39
##  3rd Qu.:271379011   3rd Qu.:0.00      3rd Qu.:1.00
##  Max.   :337534044   Max.   :1.00      Max.   :1.00
```

```
##  NA's   :55006       NA's   :55006     NA's   :55006
##  Health.Products.Division Automotive.Division
##  Min.   :1                Min.   :0.00
##  1st Qu.:1                1st Qu.:0.00
##  Median :1                Median :0.00
##  Mean   :1                Mean   :0.13
##  3rd Qu.:1                3rd Qu.:0.00
##  Max.   :1                Max.   :1.00
##  NA's   :55006            NA's   :55006
##  Personal.Electronics.Division Computers.Division Garden.Division
##  Min.   :0.00                  Min.   :0.00       Min.   :0.00
##  1st Qu.:0.00                  1st Qu.:0.00       1st Qu.:0.00
##  Median :0.00                  Median :0.00       Median :0.00
##  Mean   :0.47                  Mean   :0.05       Mean   :0.27
##  3rd Qu.:1.00                  3rd Qu.:0.00       3rd Qu.:1.00
##  Max.   :1.00                  Max.   :1.00       Max.   :1.00
##  NA's   :55006                 NA's   :55006      NA's   :55006
##  Novelty.Gift.Division Jewelry.Division    X             X.1
##  Min.   :0.00          Min.   :0.00     Mode:logical   Mode:logical
##  1st Qu.:0.00          1st Qu.:0.00     NA's:60004     NA's:60004
##  Median :0.00          Median :0.00
##  Mean   :0.23          Mean   :0.36
##  3rd Qu.:0.00          3rd Qu.:1.00
##  Max.   :1.00          Max.   :1.00
##  NA's   :55006         NA's   :55006
##    X.2            X.3            X.4            X.5
##  Mode:logical   Mode:logical   Mode:logical   Mode:logical
##  NA's:60004     NA's:60004     NA's:60004     NA's:60004
##
##
##
##
##
##    X.6            X.7            X.8            X.9
##  Mode:logical   Mode:logical   Mode:logical   Mode:logical
##  NA's:60004     NA's:60004     NA's:60004     NA's:60004
##
##
##
##
##
```

```
head(crossSell.df)
```

```
##   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
```

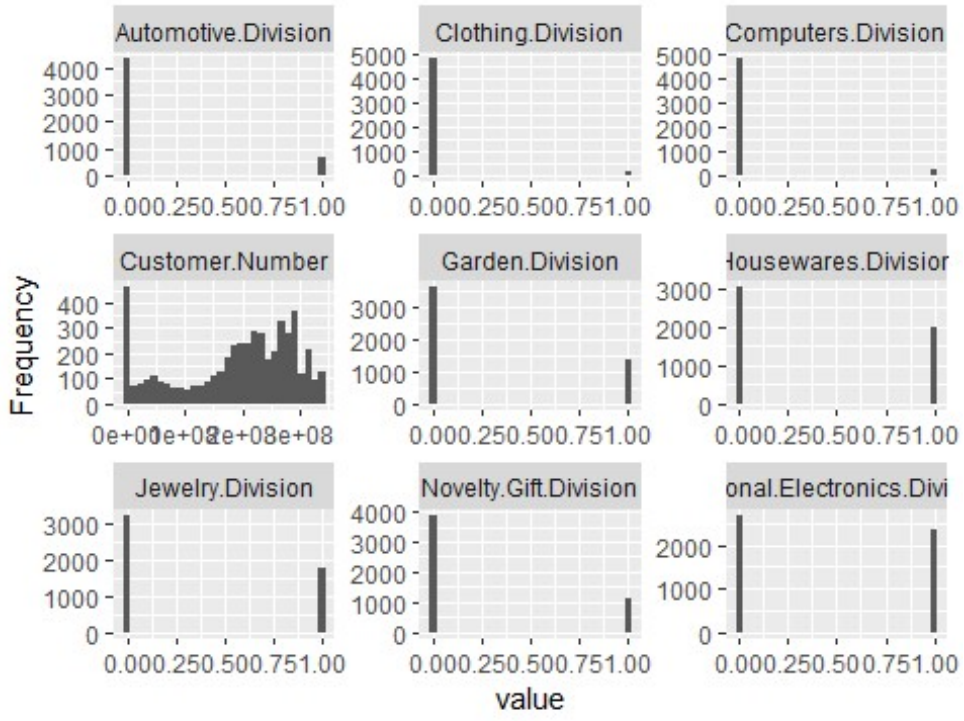
```
## 6           72553                 0                   1
##   Health.Products.Division Automotive.Division
## 1                        1                   1
## 2                        1                   1
## 3                        1                   1
## 4                        1                   1
## 5                        1                   0
## 6                        1                   1
##   Personal.Electronics.Division Computers.Division Garden.Division
## 1                             1                  0               0
## 2                             1                  0               1
## 3                             1                  0               1
## 4                             1                  0               1
## 5                             1                  0               1
## 6                             1                  0               1
##   Novelty.Gift.Division Jewelry.Division  X X.1 X.2 X.3 X.4 X.5 X.6 X.7
## 1                     1                0 NA  NA  NA  NA  NA  NA  NA  NA
## 2                     1                1 NA  NA  NA  NA  NA  NA  NA  NA
## 3                     1                1 NA  NA  NA  NA  NA  NA  NA  NA
## 4                     1                0 NA  NA  NA  NA  NA  NA  NA  NA
## 5                     1                0 NA  NA  NA  NA  NA  NA  NA  NA
## 6                     1                1 NA  NA  NA  NA  NA  NA  NA  NA
##   X.8 X.9
## 1  NA  NA
## 2  NA  NA
## 3  NA  NA
## 4  NA  NA
## 5  NA  NA
## 6  NA  NA
```

```
str(crossSell.df)
```

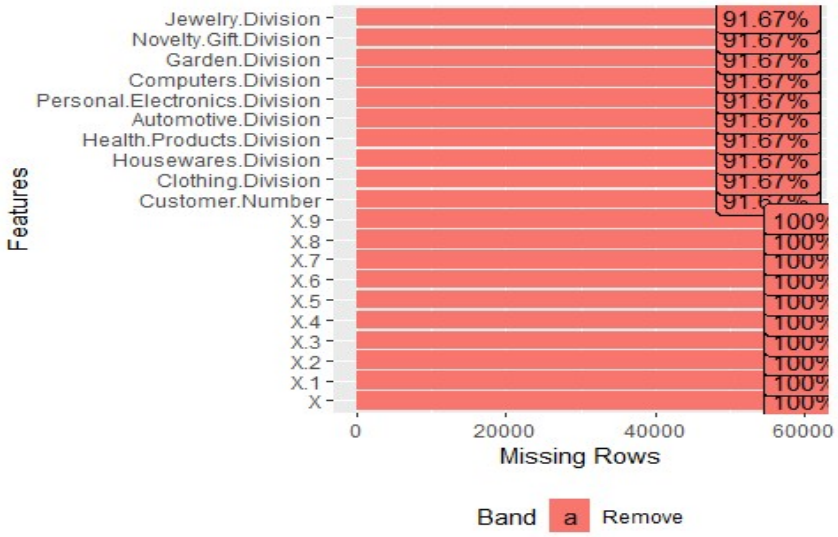
```
## 'data.frame':    60004 obs. of  20 variables:
##  $ Customer.Number              : int  11569 13714 46391 67264 67363 72553
79814 80903 91439 96701 ...
##  $ Clothing.Division            : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Housewares.Division          : int  1 1 1 0 0 1 1 1 0 1 ...
##  $ Health.Products.Division     : 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 ...
##  $ X                            : logi  NA NA NA NA NA NA ...
##  $ X.1                          : logi  NA NA NA NA NA NA ...
##  $ X.2                          : logi  NA NA NA NA NA NA ...
##  $ X.3                          : logi  NA NA NA NA NA NA ...
##  $ X.4                          : logi  NA NA NA NA NA NA ...
##  $ X.5                          : logi  NA NA NA NA NA NA ...
```

```
##  $ X.6                           : logi  NA NA NA NA NA NA ...
##  $ X.7                           : logi  NA NA NA NA NA NA ...
##  $ X.8                           : logi  NA NA NA NA NA NA ...
##  $ X.9                           : logi  NA NA NA NA NA NA ...
```

```
# Histograms using DataExporer
plot_histogram(crossSell.df, ncol=3)
```

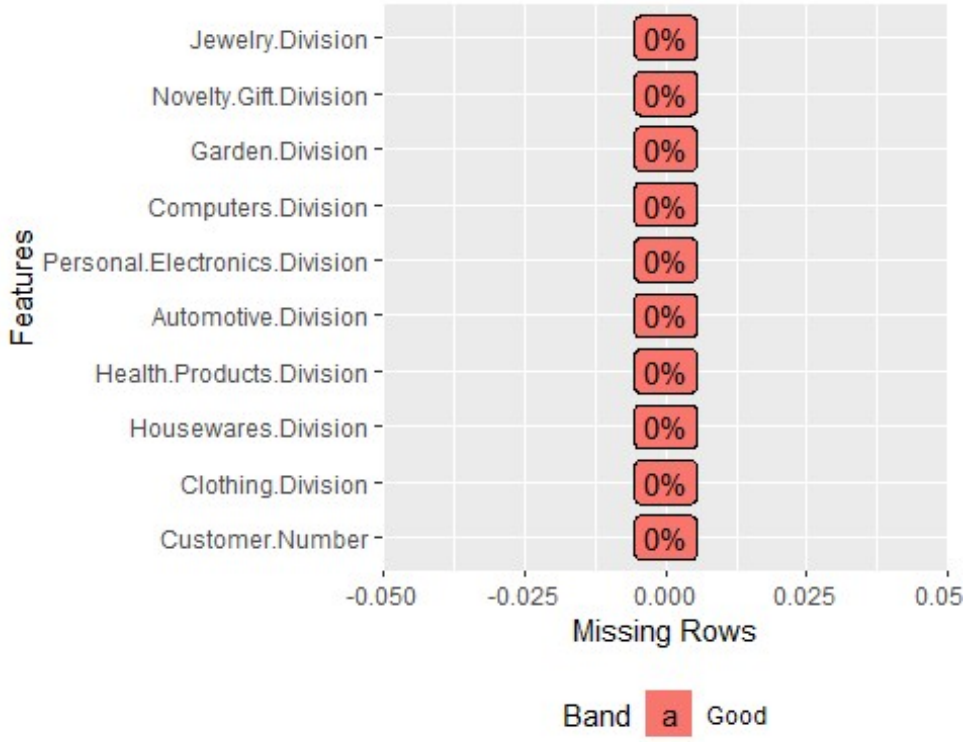



```
# Discover missing rows using DataExplorer
plot_missing(crossSell.df)
```



3. Prep data for modeling

```
# Eliminating blanks are not necessary for Apriori, but might be helpful for
other techniques, such as collaborative filtering

# Eliminate the blank columns in dataset
crossSell.df <- crossSell.df[,1:10]

# Eliminate the blank rows
crossSell.df <- crossSell.df[rowSums(is.na(crossSell.df)) != ncol(crossSell.d
f),]

# Remove first column and convert to matrix
crossSell.mat <- as.matrix( crossSell.df[, -1])

# Check our missing data again
plot_missing(crossSell.df)
```



4. Build model

```
# Convert into a transactions database
crossSell.trans <- as(crossSell.mat, "transactions")

# Get the rules using Apriori
rules <- apriori(crossSell.trans, parameter = list(supp = 0.01, conf = 0.5, t
arget = "rules"))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.5    0.1    1 none FALSE            TRUE       5    0.01      1
##  maxlen target   ext
##      10  rules FALSE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 49
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[9 item(s), 4998 transaction(s)] done [0.00s].
## sorting and recoding items ... [9 item(s)] done [0.00s].
## creating transaction tree ... done [0.02s].
## checking subsets of size 1 2 3 4 5 6 7 done [0.00s].
## writing ... [424 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

5. Evaluate model

```
# Inspect the first six rules, sorted by their lift
inspect(head(sort(rules, by = "lift"), n = 6))
```

```
##     lhs                                rhs                  support confid
ence     lift count
## [1] {Automotive.Division,
##      Personal.Electronics.Division,
##      Novelty.Gift.Division,
##      Jewelry.Division}              => {Garden.Division} 0.02400960  0.816
3265 3.000000   120
## [2] {Health.Products.Division,
##      Automotive.Division,
##      Personal.Electronics.Division,
##      Novelty.Gift.Division,
##      Jewelry.Division}              => {Garden.Division} 0.02400960  0.816
3265 3.000000   120
## [3] {Automotive.Division,
##      Novelty.Gift.Division,
##      Jewelry.Division}              => {Garden.Division} 0.02781112  0.812
8655 2.987281   139
## [4] {Health.Products.Division,
##      Automotive.Division,
##      Novelty.Gift.Division,
##      Jewelry.Division}              => {Garden.Division} 0.02781112  0.812
8655 2.987281   139
## [5] {Automotive.Division,
##      Personal.Electronics.Division,
```

```
##      Novelty.Gift.Division}         => {Garden.Division} 0.03541417  0.797
2973 2.930068   177
## [6] {Health.Products.Division,
##      Automotive.Division,
##      Personal.Electronics.Division,
##      Novelty.Gift.Division}         => {Garden.Division} 0.03541417  0.797
2973 2.930068   177
```

```r
# Prune the redundant rules for dense visualizations (Zhou, 2017).
rules.sorted <- sort(rules, by="lift")
rules.sorted
```

```
## set of 424 rules
```

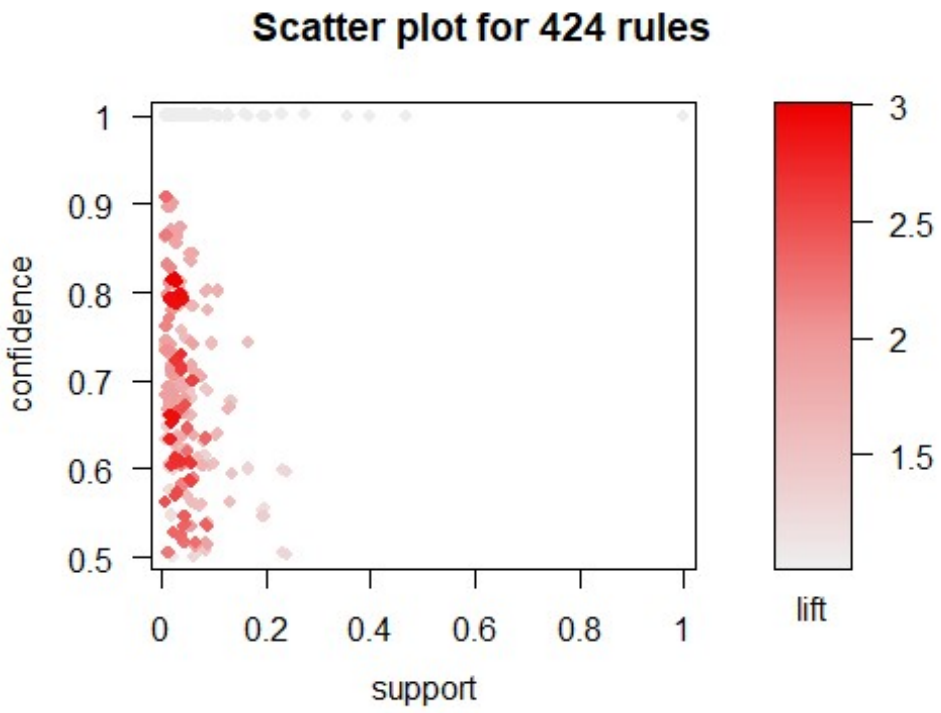
```r
subset.matrix <- is.subset(rules.sorted,rules.sorted)
subset.matrix[lower.tri(subset.matrix, diag=T)] <- F # keeps the lower triang
le (including diagonal)
redundant <- apply(subset.matrix, 2, any)
rules.pruned <- rules.sorted[!redundant]
rules.pruned
```

```
## set of 68 rules
```

```r
inspect(head(sort(rules.pruned, by = "lift"), n = 10))
```

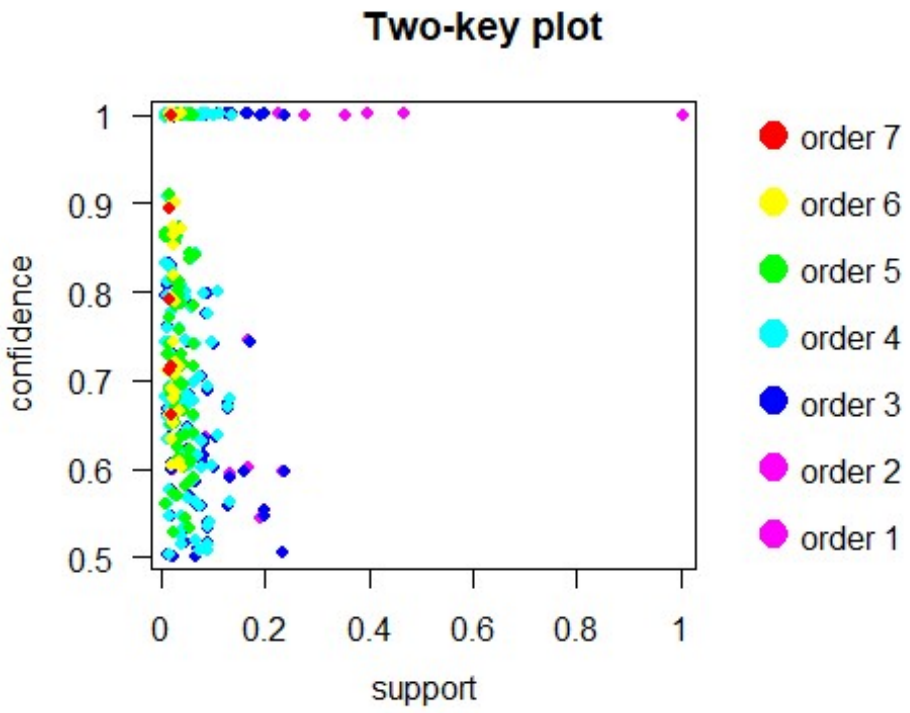
```
##      lhs                                   rhs                         support
confidence     lift count
## [1]  {Automotive.Division,
##       Personal.Electronics.Division,
##       Novelty.Gift.Division,
##       Jewelry.Division}              => {Garden.Division}       0.02400960
0.8163265 3.000000   120
## [2]  {Automotive.Division,
##       Novelty.Gift.Division,
##       Jewelry.Division}              => {Garden.Division}       0.02781112
0.8128655 2.987281   139
## [3]  {Automotive.Division,
##       Personal.Electronics.Division,
##       Novelty.Gift.Division}         => {Garden.Division}       0.03541417
0.7972973 2.930068   177
## [4]  {Automotive.Division,
##       Novelty.Gift.Division}         => {Garden.Division}       0.04401761
0.7913669 2.908273   220
## [5]  {Housewares.Division,
##       Automotive.Division,
##       Personal.Electronics.Division} => {Garden.Division}       0.03681473
0.7301587 2.683333   184
## [6]  {Housewares.Division,
##       Personal.Electronics.Division,
##       Garden.Division,
##       Jewelry.Division}              => {Novelty.Gift.Division} 0.03821529
```

```
0.6082803 2.673865   191
## [7]  {Housewares.Division,
##       Automotive.Division,
##       Personal.Electronics.Division} => {Novelty.Gift.Division} 0.03061224
0.6071429 2.668865   153
## [8]  {Personal.Electronics.Division,
##       Garden.Division,
##       Jewelry.Division}              => {Novelty.Gift.Division} 0.05322129
0.6059226 2.663501   266
## [9]  {Automotive.Division,
##       Personal.Electronics.Division,
##       Jewelry.Division}              => {Garden.Division}       0.03681473
0.7131783 2.620930   184
## [10] {Housewares.Division,
##       Personal.Electronics.Division,
##       Garden.Division}               => {Novelty.Gift.Division} 0.05762305
0.5853659 2.573139   288

# Evaluate the rules
# Additional rule context, e.g. chiSquare, cosine (Zhou, 2017)
head(interestMeasure(rules.pruned, c("support", "chiSquare", "confidence", "c
osine", "coverage", "leverage", "lift", "oddsRatio"), crossSell.df))

##      support chiSquared confidence    cosine   coverage   leverage
## 1 0.02400960  0.4531294  0.8163265 0.2683818 0.02941176 0.01600640
## 2 0.02781112  0.5230142  0.8128655 0.2882354 0.03421369 0.01850128
## 3 0.03541417  0.6473064  0.7972973 0.3221271 0.04441777 0.02332770
## 4 0.04401761  0.8017892  0.7913669 0.3577922 0.05562225 0.02888230
## 5 0.03681473  0.5624569  0.7301587 0.3143027 0.05042017 0.02309495
## 6 0.03821529  0.5531137  0.6082803 0.3196600 0.06282513 0.02392313
##       lift oddsRatio
## 1 3.000000 12.942652
## 2 2.987281 12.828471
## 3 2.930068 11.946295
## 4 2.908273 11.911676
## 5 2.683333  8.214286
## 6 2.673865  6.135874

# Visualize the rules (Hahsler & Chelluboina, n.d.)
plot(rules.sorted)

## To reduce overplotting, jitter is added! Use jitter = 0 to prevent jitter.
```




```
plot(rules.sorted, method = "two-key plot")
```

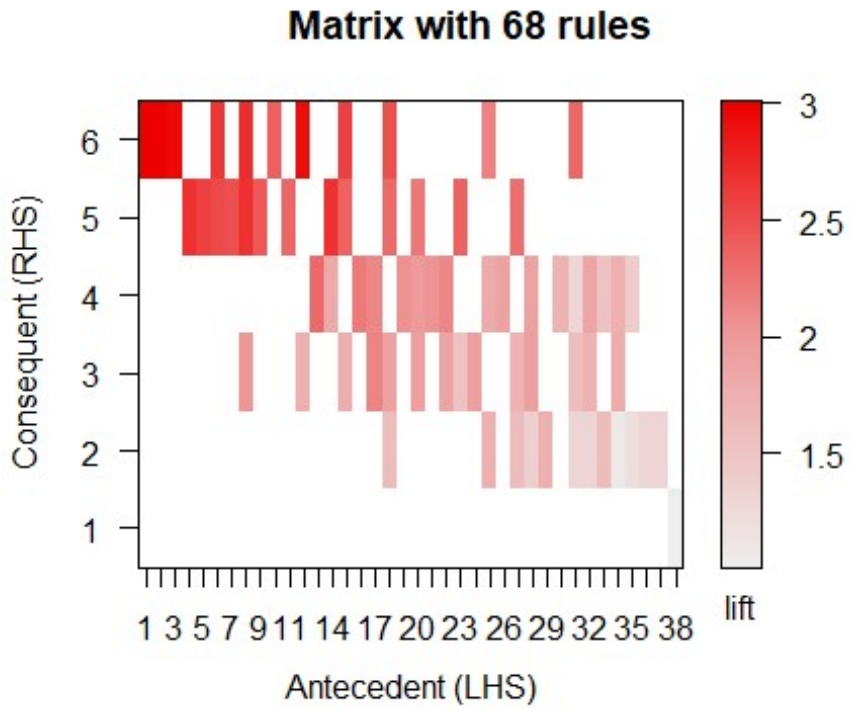
```
## To reduce overplotting, jitter is added! Use jitter = 0 to prevent jitter.
```



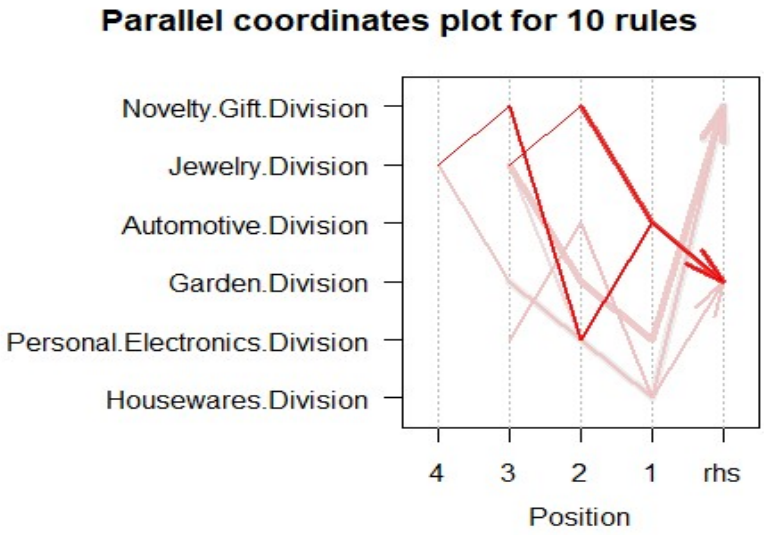
```r
# (Hahsler & Chelluboina, n.d.)
plot(rules.pruned, method = "matrix", measure = "lift")
```

```
## Itemsets in Antecedent (LHS)
##  [1] "{Automotive.Division,Personal.Electronics.Division,Novelty.Gift.Division,Jewelry.Division}"
##  [2] "{Automotive.Division,Novelty.Gift.Division,Jewelry.Division}"
##  [3] "{Automotive.Division,Personal.Electronics.Division,Novelty.Gift.Division}"
##  [4] "{Housewares.Division,Personal.Electronics.Division,Garden.Division,Jewelry.Division}"
##  [5] "{Housewares.Division,Personal.Electronics.Division,Garden.Division}"
##  [6] "{Automotive.Division,Personal.Electronics.Division,Jewelry.Division}"
##  [7] "{Housewares.Division,Personal.Electronics.Division,Computers.Division}"
##  [8] "{Housewares.Division,Automotive.Division,Personal.Electronics.Division}"
##  [9] "{Housewares.Division,Garden.Division,Jewelry.Division}"
## [10] "{Automotive.Division,Jewelry.Division}"
## [11] "{Housewares.Division,Automotive.Division,Jewelry.Division}"
## [12] "{Automotive.Division,Novelty.Gift.Division}"
## [13] "{Clothing.Division,Personal.Electronics.Division,Jewelry.Division}"
## [14] "{Personal.Electronics.Division,Garden.Division,Jewelry.Division}"
## [15] "{Automotive.Division,Personal.Electronics.Division}"
## [16] "{Personal.Electronics.Division,Computers.Division,Jewelry.Division}"
## [17] "{Computers.Division,Novelty.Gift.Division}"
## [18] "{Housewares.Division,Automotive.Division}"
## [19] "{Clothing.Division,Jewelry.Division}"
## [20] "{Personal.Electronics.Division,Computers.Division}"
## [21] "{Clothing.Division,Novelty.Gift.Division}"
## [22] "{Clothing.Division,Personal.Electronics.Division}"
## [23] "{Personal.Electronics.Division,Garden.Division}"
## [24] "{Housewares.Division,Computers.Division}"
## [25] "{Novelty.Gift.Division,Jewelry.Division}"
## [26] "{Personal.Electronics.Division,Novelty.Gift.Division,Jewelry.Division}"
## [27] "{Housewares.Division,Garden.Division}"
## [28] "{Computers.Division,Garden.Division}"
## [29] "{Housewares.Division,Novelty.Gift.Division}"
## [30] "{Personal.Electronics.Division,Jewelry.Division}"
## [31] "{Automotive.Division}"
## [32] "{Clothing.Division}"
## [33] "{Novelty.Gift.Division}"
## [34] "{Computers.Division}"
## [35] "{Jewelry.Division}"
## [36] "{Garden.Division}"
## [37] "{Housewares.Division}"
## [38] "{}"
## Itemsets in Consequent (RHS)
```

```
## [1] "{Health.Products.Division}"      "{Personal.Electronics.Division}"
## [3] "{Jewelry.Division}"              "{Housewares.Division}"
## [5] "{Novelty.Gift.Division}"         "{Garden.Division}"
```




```
# (Hahsler & Chelluboina, n.d.)
# Limit to top 10 rules by lift
subrules <- head(rules.pruned, n = 10, by = "lift")
plot(subrules, method = "paracoord")
```



Assignment Description

Association rules are a common unsupervised learning technique for determining what items are commonly purchased together (Shmueli, Bruce, Yahav, Patel, & Lichtendahl, 2018). Following the first five steps of CRISP-DM, I completed an association rules analysis to decide which magazine categories should be marketed to which customers based on previous purchase behavior.

After studying the business problem, I used several data exploration functions to understand the data. This activity exposed blank data that was cleaned up in the "prep data" phase. The data was then coerced into a transaction binary incidence matrix, and the rules were generated using the Apriori function from the arules R package. I chose .01 support and .5 confidence for the algorithm because the resulting data set size worked well for several evaluative visualizations.

Next, I began model evaluation by generating a pruned version of the data to eliminate redundancy. This abbreviated set helps for some of the dense visualizations in the arulesViz R package, and may be a reasonable step regardless (Zhou, 2017). I then generated a set of measures using the interestMeasure function of the arules R package including support, chiSquare, confidence, cosine, coverage, leverage, lift, and oddsRatio.

To facilitate data visualization, I employed the arulesViz R package for concise and powerful visuals specifically designed for association rules. The first two charts are scatterplots utilizing different measures for the data point color. These charts show clearly that the support for the top lift rules are relatively low, and as support increases, lift decreases. The two-key plot is interesting to view the general support and confidence for the number of items in each rule.

Next, I plotted a matrix grid where the consequent and antecedent values are ordered along the y and x-axes such that the top left cell holds the greatest lift. To save space, this chart is labeled with numbers, which are decoded in the preceding output text. This chart shows that the top two categories (consequent) by lift ratio are "Garden" and "Novelty.Gift". The final chart is a parallel coordinate plot that shows the top 10 rules by lift. This chart also shows that the top 10 rules lead to either "Novelty.Gift" or "Garden". The weakness of this chart is that it can be difficult to pinpoint individual rule paths from beginning to end.

Table 1

Association Rule Interpretation (Shmueli et al., 2018)

If items were purchased in these categories	**Then an item will be purchased in this category**	**With this confidence**	**The rule was observed in this many out of 4998 transactions**	**The lift ratio for the rule is**
Automotive Personal Electronics Novelty Gift Jewelry	Garden	81%	120	3.0
Automotive Novelty Gift Jewelry	Garden	81%	139	2.9
Automotive Personal Electronics Novelty Gift	Garden	80%	177	2.9
Automotive Novelty Gift	Garden	79%	220	2.9
Housewares Automotive Personal Electronics	Garden	73%	184	2.7
Housewares Personal Electronics Garden Jewelry	Novelty Gift	61%	191	2.7
Housewares Automotive Personal Electronics	Novelty Gift	61%	153	2.7
Personal Electronics Garden Jewelry	Novelty Gift	61%	266	2.7

Automotive Personal Electronics Jewelry	Garden	71%	184	2.6
Housewares Personal Electronics Garden	Novelty Gift	59%	288	2.6

Rule Interpretation

Table 1 shows an interpretation of the top 10 pruned rules in table format. One fact jumps out in table 1 and several of the charts: Although the confidence in the top rules are high, the support for these rules are low (288/4998 is the highest support of these rules). According to Shmueli et al. (2018), this may be an indication that confidence should be sacrificed for a list of higher-supported rules. For instance, if the minimum support used for the Apriori function is raised from .01 to .1, several different rules boil to the top where the confidence is around 45% and the support is around 10%. The lift ratio is greater for the higher confidence, lower support rule sets, indicating a possible better financial return per customer, although there are fewer customers to solicit. For catalog marketing, the higher support threshold may be desirable since a maximum return per catalog may be less important than over-all sales numbers. Another good option is that both lists are maintained: a highly targeted (high confidence) list, and a larger but lower-confidence list.

Lessons Learned

This was a valuable exercise in creating, evaluating, visualizing, and configuring association rules. I learned that specialized visualization libraries exist for visualizing specific techniques, such as arulesViz for association rules. I also learned how one may interpret lift and confidence in a business-centric way, which may lead the final implementation away from the highest lift. Finally, I learned several good charts and tables for association rules that truly help to evaluate and interpret the results of the analysis.

References

Hahsler, M., & Chelluboina, S. (n.d.). Visualizing Association Rules: Introduction to the R-extension Package arulesViz. Retrieved November 21, 2019, from https://cran.r-project.org/web/packages/arulesViz/vignettes/arulesViz.pdf

Shmueli, G., Bruce, P., Yahav, I., Patel, N., & Lichtendahl, K. (2018) Data Mining for Business Analytics: Concepts, Techniques, and Applications in R, Hoboken, New Jersey: John Wiley & Sons, Inc.

Zhou, L. (2017). Association rules. Retrieved from https://rpubs.com/lingyanzhou/examples-association-rules