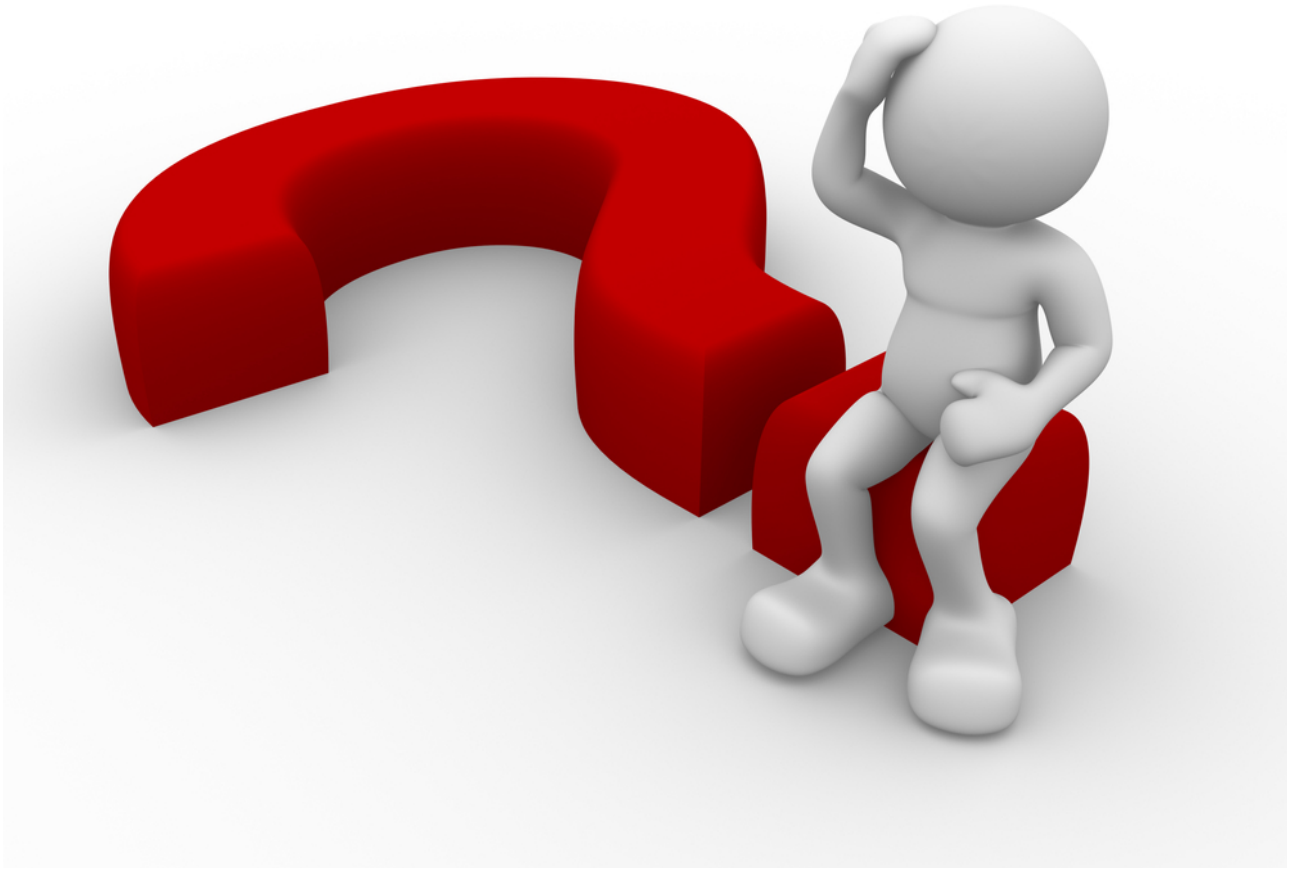
Title: Discovering Flourish and Blotts Best Sellers and Future Sales Opportunities
By: Dylan Clark Date: March 11th, 2017



According to Flourish and Blotts Sales data what is their best top 10 sellers? Additionally, what books would be good to display to book club members and others?

Abstract: The following excerpt identifies the top 10 sellers of Flourish and Blotts book shop. It also identifies books that should be displayed in the new extra display cases that Flourish and Blotts purchased specifically for Book Club books and Best Sellers.

The Area of Opportunity & Data Set

Flourish & Blotts provided us a data set that contained a list of items that customers puchased throughout their transactions. The data set that was provided contained 92,108 unique transactions with 22,047 different books that had been purchased. The problem that Flourish & Blotts ran into was that they did not have a clean way to read and analyze the data. Thus, they turned to us to be able to read the data and analyze it for answers on questions they had.

Flourish and Blotts had the following questions: 1. What were the top 15 sellling books from the data set? 2. What books would be good to display in a newly found book club section 3. What other books should be displayed in a display case at the front of the store?

After given the set of questions from Flourish & Blotts book store I started to brainstorm about possible data cleansing activities and data mining techniques that could be used to one clean the data and two to answer the questions. After brainstorming I knew that I could use R to help clean the data as well as use it to association rule mining techniques to answer the questions that Flourish & Blotts had requested to be answered. From there I moved on to begin my plan of attack!

My Plan of Attack

I started with using R to begin my data cleaning. I loaded the appropriate information that I needed into R. Using R I started using some data cleansing technique to eliminate duplicate entries of data from the data set as well as mock the data into a more readable format.

DATA PREPARATION

```
getwd()
```

```
## [1] "C:/Users/dylan/Desktop/MBA/ADMSpring/Week_4/Data"
```

```
setwd("C:/Users/dylan/Desktop/MBA/ADMSpring/Week_4/Data")

library(Matrix)
```

```
## Warning: package 'Matrix' was built under R version 3.3.3
```

```
library(arules)
```

```
## Warning: package 'arules' was built under R version 3.3.3
```

```
##
## Attaching package: 'arules'
```

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

```
library(sqldf)
```

```
## Warning: package 'sqldf' was built under R version 3.3.3
```

```
## Loading required package: gsubfn
```

```
## Warning: package 'gsubfn' was built under R version 3.3.3
```

```
## Loading required package: proto
```

```
## Warning: package 'proto' was built under R version 3.3.3
```

```
## Loading required package: RSQLite
```

```
## Warning: package 'RSQLite' was built under R version 3.3.3
```

```
library(arulesViz)
```

```
## Warning: package 'arulesViz' was built under R version 3.3.3
```

```
## Loading required package: grid
```

```
BOOKBASKETS <- read.transactions("bookdata.tsv.gz", format="single",sep="\t", c
ols=c("userid", "title"), rm.duplicates=T)
```

```
## distribution of transactions with duplicates:
## items
##   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18
## 701 222 106  68  43  39  23  24  18  18  16  10   7   7  13   7   8   5
##  19  20  21  22  23  25  26  27  28  29  30  31  33  34  35  38  39  42
##   3   9   4   4   3   2   2   5   4   5   4   4   1   2   1   1   1   2
##  44  45  47  48  49  52  56  57  59  61  63  71  73  80  84  86  91  93
##   1   1   1   1   1   1   2   1   2   1   2   1   1   1   1   1   1   1
##  95  96  99 103 158 206 260 891
##   1   1   1   1   1   2   1   1
```

After cleaning the data I was able to begin my basic exploration to find out some key elements that I needed to utlimately start getting to the meat of some of the major questions that were asked by Flourish and Blotts. Thus, I started to analyze the data set with basic exploratory data techniques. I was able to identify the total amount of purchases, what transactions looked like, how often each book was purchased and what books were purchased on more than one transaction.

Exploring the Data Set

```
summary(BOOKBASKETS)
```

```
## transactions as itemMatrix in sparse format with
##  92108 rows (elements/itemsets/transactions) and
##  220447 columns (items) and a density of 5.034811e-05
##
## most frequent items:
##                           Wild Animus
##                                  2502
##             The Lovely Bones: A Novel
##                                  1295
##                     She's Come Undone
##                                   934
##                     The Da Vinci Code
##                                   905
## Harry Potter and the Sorcerer's Stone
##                                   832
##                                (Other)
##                               1015847
##
## element (itemset/transaction) length distribution:
## sizes
##     1     2     3     4     5     6     7     8     9    10    11    12
## 51286 10804  5760  3850  2700  2044  1609  1241  1075   901   755   643
##    13    14    15    16    17    18    19    20    21    22    23    24
##   555   460   464   393   342   332   268   258   237   222   195   179
##    25    26    27    28    29    30    31    32    33    34    35    36
##   182   170   156   154   129   114   120   103   128   101    88    94
##    37    38    39    40    41    42    43    44    45    46    47    48
##    98    82    71    80    60    81    70    65    73    77    79    54
##    49    50    51    52    53    54    55    56    57    58    59    60
##    54    47    43    50    52    43    38    47    43    39    46    44
##    61    62    63    64    65    66    67    68    69    70    71    72
##    36    30    31    24    46    25    27    35    29    28    32    22
##    73    74    75    76    77    78    79    80    81    82    83    84
##    26    24    25    19    22    26    27    21    29    13    29    21
##    85    86    87    88    89    90    91    92    93    94    95    96
##    20    13    15    10    21    26    14    16    23    13    15    19
##    97    98    99   100   101   102   103   104   105   106   107   108
##    14    20    21    12    12    14    17    16    15    12    13    18
##   109   110   111   112   113   114   115   116   117   118   119   120
##    11    11    15    16    14    10    12    14     5     9    17     5
##   121   122   123   124   125   126   127   128   129   130   131   132
##    10    11    11     9    17    16    14    12     6    10    10     8
##   133   134   135   136   137   138   139   140   141   142   143   144
##     8     9    10    11     8     5     7    12     9     6     6     7
##   145   146   147   148   149   150   151   152   153   154   155   156
##     7     9     7     6     6     5     8    10     7    11     6     9
##   157   158   159   160   161   162   163   164   165   166   167   168
##     7     3     8     7     8     6     7     9     4     9     6     6
```

```
##   169   170   171   172   173   174   175   176   177   178   179   180
##     1    10     6     6     6     2     7     8     3     9     5     5
##   181   182   183   184   185   186   187   188   189   190   191   192
##     4     9     6     8     3     2     9     5     5     6     7     4
##   193   194   195   196   197   198   199   200   201   202   203   204
##     7     1     6     8     4     2     7     4     9     6     4     2
##   205   206   207   208   209   210   212   213   214   215   216   217
##     4     4     4    10     2     4     1     3     3     4     3     5
##   218   219   220   221   222   223   224   225   226   227   228   229
##     1     5     5     8     2     4     5     3     7     3     2     2
##   230   231   232   233   234   235   236   237   238   239   240   241
##     6     6     5     3     8     5     5     6     4     6     2     4
##   242   243   244   245   246   247   248   249   250   251   252   253
##     5     5     1     2     1     1     4     1     2     1     4     2
##   254   255   256   257   258   259   260   261   262   263   264   265
##     2     2     3     1     3     4     2     4     3     4     1     1
##   266   267   268   269   270   271   272   275   276   277   278   279
##     2     4     5     2     5     4     3     6     4     1     1     2
##   280   281   282   283   284   285   286   287   288   289   290   291
##     1     3     5     2     2     2     3     2     4     1     1     4
##   293   295   296   297   298   300   301   302   304   305   306   307
##     1     1     2     1     4     1     3     1     1     1     2     1
##   308   309   310   311   313   314   315   316   317   319   320   321
##     3     4     2     1     2     3     1     2     3     3     1     1
##   322   323   324   325   326   327   328   329   330   331   332   333
##     3     4     1     2     2     1     1     6     1     2     2     2
##   334   336   337   338   339   340   341   342   343   344   346   347
##     2     2     1     1     1     1     1     2     2     3     2     2
##   348   349   350   352   354   356   357   358   359   360   363   364
##     1     1     1     2     1     1     1     1     1     1     1     1
##   366   367   368   369   370   372   373   374   375   376   377   378
##     3     1     1     2     1     2     1     2     1     2     2     1
##   379   381   382   383   384   385   386   387   388   389   390   391
##     1     2     2     1     4     2     2     1     3     1     1     2
##   392   393   394   395   396   397   398   399   401   403   406   407
##     4     1     2     1     2     1     4     1     1     3     1     1
##   409   410   412   415   416   417   418   419   420   421   422   425
##     3     1     1     1     3     1     4     1     1     2     2     1
##   427   428   430   435   436   437   438   440   441   442   444   445
##     1     1     1     3     1     1     2     2     4     1     1     1
##   446   447   448   451   453   455   456   457   458   459   460   464
##     1     1     2     1     1     2     2     1     1     2     1     2
##   466   468   472   476   480   481   485   487   489   492   493   494
##     1     2     1     1     1     1     2     4     1     1     1     1
##   496   497   498   507   508   510   512   515   516   517   520   522
##     1     2     1     1     1     1     1     1     1     1     2     2
##   523   524   525   526   527   528   532   534   537   539   540   542
##     1     2     1     1     1     2     2     1     1     1     2     1
##   543   544   547   562   566   570   572   573   575   577   580   590
```

```
##     1     1     1     2     1     1     1     1     2     1     2     1
##   591   596   597   599   600   601   602   603   611   613   620   624
##     2     2     2     1     1     1     1     3     2     1     2     1
##   625   626   627   628   629   649   653   656   657   658   661   662
##     1     1     1     2     1     1     1     1     1     1     1     1
##   664   665   666   667   670   673   675   690   692   696   698   705
##     1     1     1     1     1     1     1     1     1     1     1     1
##   707   709   713   721   724   725   727   732   734   736   740   742
##     1     1     1     1     1     1     2     1     1     2     1     2
##   745   746   748   765   767   769   775   781   782   785   790   796
##     2     1     1     1     1     1     2     1     1     1     1     1
##   802   804   805   807   812   814   818   820   829   834   837   849
##     1     1     1     1     1     1     1     1     1     1     1     1
##   855   858   870   878   880   887   894   897   903   917   922   933
##     1     2     1     1     1     1     1     1     1     1     1     3
##   945   948   950   953   961   965   968   971   978   985  1009  1010
##     1     1     1     1     1     1     1     1     1     1     1     1
##  1016  1018  1025  1030  1035  1036  1039  1054  1074  1076  1077  1078
##     1     1     1     1     1     1     1     1     1     2     2     1
##  1079  1080  1083  1088  1089  1111  1132  1133  1136  1149  1152  1153
##     1     1     1     1     1     1     1     1     1     1     1     1
##  1157  1171  1183  1190  1199  1202  1234  1241  1242  1253  1255  1260
##     1     1     1     1     1     1     1     1     1     1     1     1
##  1264  1270  1275  1285  1293  1308  1312  1317  1320  1326  1340  1351
##     1     1     1     1     1     1     2     1     1     1     1     1
##  1359  1395  1433  1463  1481  1496  1511  1512  1514  1518  1542  1578
##     1     1     1     1     1     1     1     1     1     1     1     1
##  1582  1621  1626  1631  1668  1687  1705  1734  1799  1802  1808  1840
##     1     1     1     1     1     1     1     1     1     1     1     1
##  1966  2040  2060  2124  2190  2197  2202  2215  2270  2272  2291  2295
##     1     1     1     1     1     1     1     1     1     1     1     1
##  2305  2364  2373  2413  2542  2773  2864  2884  2908  3236  3897  4187
##     1     1     1     1     1     1     1     1     1     1     1     1
##  5440  5608  5683  6196 10253
##     1     1     1     1     1
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     1.0     1.0     1.0    11.1     4.0 10250.0
##
## includes extended item information - examples:
##                                       labels
## 1                          ''N'' Is for Noose
## 2 ' Allo 'Allo: the War Diaries of Rene Artois
## 3                              ' Boule De Suif
##
## includes extended transaction information - examples:
##   transactionID
## 1            10
```

```
## 2          1000
## 3        100001
```

```
inspect(BOOKBASKETS[1:5]) #Examine the first five transactions
```

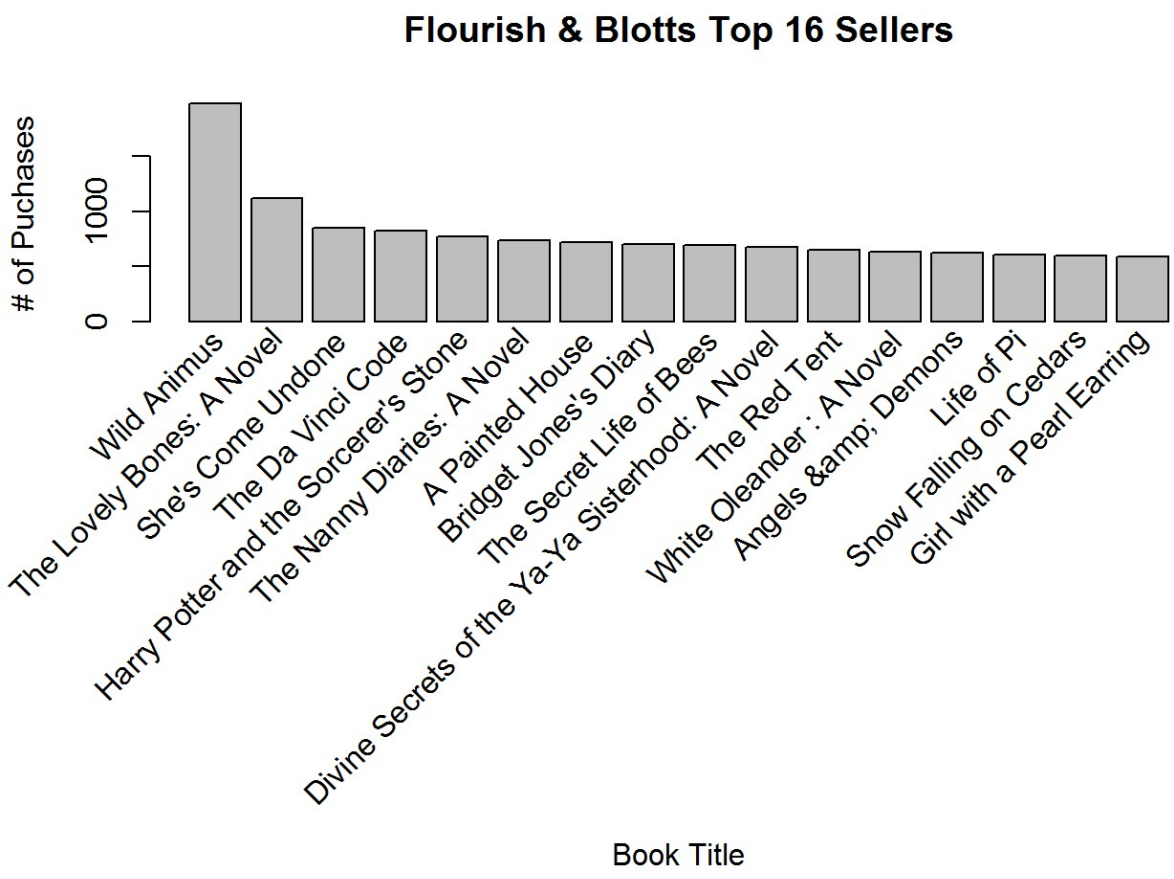
```
##    items
transactionID
## [1] {New Vegetarian: Bold and Beautiful Recipes for Every Occasio
n}                                          10
## [2] {Il Dio Delle Piccole Cos
e}                                                                    1
000
## [3] {Cybernatio
n}
             100001
## [4] {Lasher: Lives of the Mayfair Witche
s}                                                                 100002
## [5] {Chicken Soup for the Teenage Soul on Tough Stuff : Stories of Tough Tim
es and Lessons Learned,
##      Dragon Ball Z, Vol.
1,

##      Harry Potter and the Chamber of Secret
s,
##      Harry Potter and the Sorcerer's Ston
e,
##      Hole
s,

##      Pre
y,

##      Primary Colors: A Novel of Politic
s,
##      Rising Su
n,

##      The Cat Who Smelled a Ra
t,

##      The Fellowship of the Rin
g,

##      The Hunt for Red Octobe
r,

##      The Return of the Kin
g,

##      The Two Tower
s}
          100004
```

```
BASKETSIZES<-size(BOOKBASKETS) #Calculate number of books purchased by "userID"

BOOKFREQUENCY<-itemFrequency(BOOKBASKETS) #Calculate the support for each book
title

BOOKCOUNT <- (BOOKFREQUENCY/sum(BOOKFREQUENCY))*sum(BASKETSIZES)
#   Get the absolute count of book occurrences.

BOOKBASKET_USE<-BOOKBASKETS[BASKETSIZES>1] #Only keep transactions with more th
an one book purchased.
```

From here I was able to assess that the average amount of items that were purchased in a normal transaction was 11 items. This meant that the average person was on average going to buy 11 books. I also conducted some other technique to ultimately identify the total amount of each item that was purchased. Here I was able to assess the top 15 sellers are the book titles that are identified in the graph below:

THE BEST SELLERS

```
FREQ_BOOKS_DATA_FRAME <- as.data.frame(itemFrequency(BOOKBASKETS))

itemFrequencyPlot(BOOKBASKET_USE, topN=16, type="absolute", ylab = "# of Puchas
es", xlab="Book Title", main="Flourish & Blotts Top 16 Sellers")
```



The graph above identifies the top 16 books that were purchased the most at Flourish and Blotts. I dedcided to choose the top 16 books since Flourish & Blotts did not wish to display the book titeled “Wild Animus”. Flourish and Blotts did not appear to care about the overall display of the book as it appears that a lot of copies of the book are readily available compared to some of the major top sellers. We were able to assess the number of times that the books were most purchased by calculating the items overall absolute support, whcih essentially just counts the times that the item was purchased in all the transactions that were in the data set. Therefore, the top 15 books that were purhcased the most often (excluding Wild Animus) were:

- The Lovely Boones: A Novel
- She’s Come Undone
- The Da Vinci Code
- Harry Potter and the Sorcerer’s Stone
- The Nanny Diaries: A Novel
- A Painted House
- Bridget Jones’s Diary
- The Sexret Life of Bees
- Divine Secrets of the Ya=Ya Sisterhood: A Novel
- The Red Tent
- White Oleander: A Nove
- Angels & Demons
- Life of Pi

- Snow Falling on Cedars
- Girl with a Pearl Earing

After analyzing all of these transactions we were able to determine that the books provided in the list above are the most selling books that appear in more than one transaction.

Furthermore, I was able to use the data set that included books purchased in more than one transaction into the next question that needed to be identified which was “What books would be best displayed for a book club display?”. I was able to do this by using association rule mining. Association Rule Mininng is a method that is used for unstructured data like transaction data to help determine frequent patterns in the data. The technique that we will specifically used is called frequent pattern analysis or known as FPA. FPA uses algorithms to donduct a search for frequent generated item sets. Generally, an item set is a combination of items that are purchased together. So, the algorithmruns through a tree like structure and helps determine the total amount of item sets, the amount of times they appear and ultimately how often they purchased out of the overall data set. The FPA algorithms use a prperty called downware closure property that essentially does not allow a less frequent itemlist supersese another itemset in the heirarchal search tree that it used. Keeping this property allows us to determine the minimum support and amount of items that we want in our item set. While this is a general explanation we will more specifially use an algorith named apriori to help determine the most common puchased item sets. We utlimately hope to determine which itemsets are the most purchased to help determine what item sets would be good recommendations for book clubs and general displays. Ultimately, the hope is to use this analysis to have people purchase more books. However, first we must conduct some external research to help dictate what books are book club books.

So, first we did some high level research on what exactly a book club was. From the following link:

http://www.ilovelibraries.org/booklovers/bookclub (http://www.ilovelibraries.org/booklovers/bookclub)

WE were able to determine that a good book club should consist of a following: - A common topic that books are about. - A commmon time of meeting. - A common theme that the books should pertain to. - The books should faciliate discussion about general aspects that the members like to discuss.

Ultimately, we were also provided a list of books from Flourish & Blotts that had books from Oprah’s Book Club as good recommendations. The list can be found below:

Oprah’s Book Club

However, after doing some research on book clubs and books used in them I was able to do some analysis using the FPA Algorithm known as apriori to help with the recommendations oof books that should appear on the book club display and other books display.The reason that I used apriori was because it is the most frequenty used method and it is highly scalabale on large data sets, such as Flourish & Blotts data sets. Apriori works as the following:

- it first scans all transactionsin one item set, then it continues to add one item that was purchased with that additional item. From there it is able to assess the length and frequency of each generated item set. The algorith m is then able to test the candidate itemsets against the dataset and determine the min support of each threshold. IT is also able to terminate any itemset that is not frequent to keep the accuracy of the algorithm.

This is the method that was used to determine the overall recommendations of books that should be displayed in the display cases. The work conducted to complete the analysis is provided below:

```
BOOKBASKETS_RULES <- apriori(BOOKBASKET_USE, parameter = list(support = 0.000
1, confidence = 0.8, minlen = 2))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.8    0.1    1 none FALSE            TRUE       5   1e-04      2
##  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: 4
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[216031 item(s), 40822 transaction(s)] done [0.68s].
## sorting and recoding items ... [36574 item(s)] done [0.07s].
## creating transaction tree ... done [0.03s].
## checking subsets of size 1 2
```

```
## Warning in apriori(BOOKBASKET_USE, parameter = list(support = 1e-04,
## confidence = 0.8, : Mining stopped (time limit reached). Only patterns up
## to a length of 2 returned!
```

```
##  done [6.99s].
## writing ... [5296 rule(s)] done [4.28s].
## creating S4 object  ... done [0.34s].
```

```
summary(BOOKBASKETS_RULES)
```

```
## set of 5296 rules
##
## rule length distribution (lhs + rhs):sizes
##    2
## 5296
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       2       2       2       2       2       2
##
## summary of quality measures:
##     support            confidence          lift
##  Min.   :0.0001225   Min.   :0.8000   Min.   :  16.48
##  1st Qu.:0.0001225   1st Qu.:0.8333   1st Qu.:  87.26
##  Median :0.0001225   Median :0.8333   Median : 274.34
##  Mean   :0.0001565   Mean   :0.8823   Mean   :1165.31
##  3rd Qu.:0.0001470   3rd Qu.:0.9167   3rd Qu.:1936.23
##  Max.   :0.0024252   Max.   :1.0000   Max.   :8164.40
##
## mining info:
##            data ntransactions support confidence
##  BOOKBASKET_USE         40822   1e-04        0.8
```

```
BOOKBASKETS_RULES_SORTED <- sort(BOOKBASKETS_RULES, by = c("lift", "confidenc
e"))

BOOKBASKETS_RULES_SORTED_DF <- as(BOOKBASKETS_RULES_SORTED, "data.frame")

WITHOUT_HARRY_BOOKBASKETS <- sqldf("SELECT * FROM BOOKBASKETS_RULES_SORTED_DF W
HERE rules not like '%Harry Potter%'")
```

```
## Loading required package: tcltk
```

```
## Warning: Quoted identifiers should have class SQL, use DBI::SQL() if the
## caller performs the quoting.
```

```
WITHOUT_GREENMILE_BOOKBASKETS <- sqldf("SELECT * FROM WITHOUT_HARRY_BOOKBASKETS
WHERE rules not like '%Green Mile%'")
WITHOUT_ANIMUS_BOOKBASKETS <- sqldf("SELECT * FROM WITHOUT_GREENMILE_BOOKBASKET
S WHERE rules not like '%Animus%'")
```

We decided to chose the fact that an itemset at least had to have two items in it with more than an 80% confidence level to be right. We chose with at least two, because we felt that the hope was that the person would be to buy more than book in the purchase. From this outcome we also eliminated series

books such as The Harry Potter and The Green Mile from the algorithm. Also, the Wild Animus was eliminated too. Any purchase with those books we did not focus on as management made it clear not to analyze series books and the Wild Animus due to the findings.

Recommendations

Therefore, our findings are and recommendation for each display case is provided below:

The Book Club Display Case:

The Lovely Bones Book should be displayed with the following close to it: - The Passion Dream Book - The Duchess of Bloomsbury - The Woodchipper Murder - The Last Hours of Ancient Sunlight - Angel Letters

- The She's Come Undone Book should be displayed with the following close to it:
    - Bsc Mallory and Trouble
    - Logan Likes Mary Anne!
    - Maya's First Rose
    - Seven Hundred Kisses
    - The Void
- The Da Vinci Code should be displayed with the following close to it:
    - The Wreck of the Whaleship Essex
    - Some Kind of Incredible
    - The Longest Single Note
    - All's FAir, Love, War and Running for President
- The Nanny's Diaries: A Novel should be displayed with the following close to it:
    - Breakfast at Tiffany's
    - Home Run
    - Cold Feet
    - Lovingkindness
    - Answer is Yes

These books were determined based off of having a lower lift score and high confidence and support outcome. Thus, the following recommendations are based off of the lower lift score which recommedns that the lower the score the higher the likelihood of being purchased.

The Other Display Case should be organized to Best Sellers. The books that should be displayed are shown below: - A Painted House - Bridget Jones's Diary - The Sexret Life of Bees - Divine Secrets of the Ya=Ya Sisterhood: A Novel - The Red Tent - White Oleander: A Nove - Angels & Demons - Life of Pi - Snow Falling on Cedars - Girl with a Pearl Earing

The following recommendations were recommended based off of the overall score of the books absolute support value. Therefore, the books most purchased should be displayed at the front of the store with hopes of those people buying the most sought out after books.

It is these recommendations that Flourish & Blotts should take if they hope to increase their sales, by using Book Clubs and Best Selling Books in their display cases.

that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that

generated the plot.