ar_hw.R

alexaubrey

Wed Jun 6 07:04:28 2018

```
library(arules)
```

```
## Warning: package 'arules' was built under R version 3.4.4
```

```
## Loading required package: Matrix
```

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

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

```
pre_data <- read.csv("maincharlesbook.csv")

pre_data$ChildBks <- ifelse(grepl("^0$", pre_data$ChildBks), NA, "ChildBks")
pre_data$YouthBks <- ifelse(grepl("^0$", pre_data$YouthBks), NA, "YouthBks")
pre_data$CookBks <- ifelse(grepl("^0$", pre_data$CookBks), NA, "CookBks")
pre_data$DoItYBks <- ifelse(grepl("^0$", pre_data$DoItYBks), NA, "DoItYBks")
pre_data$RefBks <- ifelse(grepl("^0$", pre_data$RefBks), NA, "RefBks")
pre_data$ArtBks <- ifelse(grepl("^0$", pre_data$ArtBks), NA, "ArtBks")
pre_data$GeogBks <- ifelse(grepl("^0$", pre_data$GeogBks), NA, "GeogBks")
pre_data$ItalCook <- ifelse(grepl("^0$", pre_data$ItalCook), NA, "ItalCook")
pre_data$ItalAtlas <- ifelse(grepl("^0$", pre_data$ItalAtlas), NA, "ItalAtlas")
pre_data$ItalArt <- ifelse(grepl("^0$", pre_data$ItalArt), NA, "ItalArt")
pre_data$Florence <- ifelse(grepl("^0$", pre_data$Florence), NA, "Florence")

pre_data <- pre_data[,c("ChildBks", "YouthBks", "CookBks", "DoItYBks", "RefBks", "ArtBk
s", "GeogBks", "ItalCook",
                        "ItalAtlas", "ItalArt", "Florence")]

write.table(pre_data, file = "maincharlesbook_clean.csv", sep=",", row.names = FALSE, co
l.names=FALSE)

### Clean the data with our python script ###
system("python strip_data2.py")

### Contents of python script ###
# import os
# import re
#
# ### Take out all instances of null strings values and shift them to the left after we
 transformed the data in R ###
# outfile = open("maincharlesbook_cleaned.csv","w")
#
# for line in open("maincharlesbook_clean.csv"):
#   #Replace all instances that exist in the line
#   for item in range(0, line.count('NA,')):
#     line = line.replace('NA,', '')
#
# #Take care of potential NA on the end
# line = line.replace(',NA', '')
#
#
# outfile.write(line)
#
# outfile.close()
#
# ### Take out all the  null values and shift them to the left after we took out the nul
l strings ###
# outfile = open("maincharlesbook_clean_final.csv", 'w')
#
# for line in open("maincharlesbook_cleaned.csv"):
#   #Replace all instances that exist in the line
#   for item in range(0, line.count(',,')):
#     line = line.replace(',,', ',')
```

```
#
# #Replace comma at the end of the line
# line = re.sub('\,$', '',line)
#
# #Don't want anything where no one bought any of our books
# if re.match('^NA$', line):
#   pass
# else:
#   outfile.write(line)
#
# outfile.close()
#
# #Don't need the tmp file anymore..
# os.system("rm maincharlesbook_cleaned.csv")

data = read.transactions("maincharlesbook_clean_final.csv", format="basket", sep=",")

rules <- apriori(data=data, parameter = list(supp = 0.1, conf=0.8, minlen=2),
                 appearance=list(default="lhs", rhs="CookBks"),
                 control=list(verbose=F))

rules <- sort(rules, decreasing = TRUE, by="confidence")

print("What is the top item(s) that indicates customers will also buy or get Cook Book
s?")
```

```
## [1] "What is the top item(s) that indicates customers will also buy or get Cook Book
s?"
```

```
print("Top items are Child Books and Youth Books")
```

```
## [1] "Top items are Child Books and Youth Books"
```

```
inspect(rules[1])
```

```
##     lhs                    rhs       support  confidence lift     count
## [1] {ChildBks,YouthBks} => {CookBks} 0.148745 0.8135593  1.579637 480
```

```
print("Will your answer to the last question change if you use lift to select the best r
ule?")
```

```
## [1] "Will your answer to the last question change if you use lift to select the best
rule?"
```

```
print("No, it appears Child Books and Youth Books are still the top suggestion.")
```

```
## [1] "No, it appears Child Books and Youth Books are still the top suggestion."
```

```
rules <- sort(rules, decreasing = TRUE, by="lift")
inspect(rules[1])
```

```
##     lhs                     rhs       support  confidence lift     count
## [1] {ChildBks,YouthBks} => {CookBks} 0.148745 0.8135593  1.579637 480
```

```
rules <- apriori(data, parameter = list(supp = 0.01, conf=0.5),
                 appearance=list(default="rhs", lhs=c("ChildBks","YouthBks")),
                 control=list(verbose=F))

rules <- sort(rules, decreasing = TRUE, by="confidence")
print("What is the top items(s) that customers will also buy or get if they ahve already
 picked or bought child books and youth books together?")
```

```
## [1] "What is the top items(s) that customers will also buy or get if they ahve alread
y picked or bought child books and youth books together?"
```

```
print("Supporting our last analysis, customers will likely pick up Cook Books")
```

```
## [1] "Supporting our last analysis, customers will likely pick up Cook Books"
```

```
inspect(rules[1])
```

```
##     lhs                     rhs       support  confidence lift     count
## [1] {ChildBks,YouthBks} => {CookBks} 0.148745 0.8135593  1.579637 480
```

```
print("Will your answer to the last question change if you use lift to select the best r
ule?")
```

```
## [1] "Will your answer to the last question change if you use lift to select the best
rule?"
```

```
print("Yes, it will indeed change if we choose to use lift as a way of selecting the bes
t rule")
```

```
## [1] "Yes, it will indeed change if we choose to use lift as a way of selecting the be
st rule"
```

```
print("Now, customers will likely pick up Do it yourself books as a top suggestion.")
```

```
## [1] "Now, customers will likely pick up Do it yourself books as a top suggestion."
```

```
rules <- sort(rules, decreasing = TRUE, by="lift")
inspect(rules[1])
```

```
##     lhs                     rhs        support    confidence lift    count
## [1] {ChildBks,YouthBks} => {DoItYBks} 0.09947319 0.5440678  1.72297 321
```