"I pledge my honor that I have not violated the Honor Code during this assignment"

Jeong Lim Kim
Hye-Min Jung
Jayoung Kang
Chloe Fu

BUS 41201 Homework 1 Assignment

Amazon Reviews

The dataset consists of 13 319 reviews for selected products on Amazon from Jan-Oct 2012. Reviews include product information, ratings, and a plain text review. The data consists of three tables:
##Review subset.csv is a table containing, for each review, its

Word freq.csv
is a simple triplet matrix of word counts from the review text including

Words.csv

contains 1125 alphabetically ordered words that occur in the reviews.

Data exploration

The code below loads the data.

```
Library(tidyverse)
library(knitr) # library for nice R markdown output
setwd("C:/Users/13124/Desktop/Harris/2020-2 Spring/Big Data/HW/hw1/")

# READ REVIEWS

data<-read.table("Review_subset.csv",header=TRUE)
dim(data)
```

[1] 13319 9

```
# 13319 reviews
# ProductID: Amazon ASIN product code
# UserID:  id of the reviewer
# Score: numeric from 1 to 5
# Time: date of the review
# Summary: text review
# nrev: number of reviews by this user
# Length: length of the review (number of words)

# READ WORDS

words<-read.table("words.csv")
words<-words[,1]
length(words)
```

```
[1] 1125
```

```
#1125 unique words

# READ text-word pairings file

doc_word<-read.table("word_freq.csv")
names(doc_word)<-c("Review ID","Word ID","Times Word" )
# Review ID: row of the file  Review_subset
# Word ID: index of the word
# Times Word: number of times this word occurred in the text
```

Marginal Regression Screening

We would like to pre-screen words that associate with ratings. To this end, we run a series of (independent) marginal regressions of review Score on word presence in review text for each of 1125 words.

In the starter script below, you will find a code to run these marginal regressions (both in parallel and sequentially). The code gives you a set of p-values for a marginal effect of each word. That is, we fit

$$\text{stars}_i = \alpha + \beta_j I[x_{ji} > 0] + \varepsilon_{ji}$$

for each word term j with count x_{ji} in review i, and return the p-value associated with a test of $\beta_j \neq 0$. We'll use these 1125 independent regressions to screen words.

```
# We'll do 1125 univariate regressions of
# star rating on word presence, one for each word.
# Each regression will return a p-value, and we can
# use this as an initial screen for useful words.

# Don't worry if you do not understand the code now.
# We will go over similar code in  the class in a few weeks.

# Create a sparse matrix of word presence


library(gamlr)
## Loading required package: Matrix
##
## Attaching package: 'Matrix'
## The following object is masked from 'package:tidyr':
##
##     expand
spm<-sparseMatrix(i=doc_word[,1],
                  j=doc_word[,2],
                  x=doc_word[,3],
                  dimnames=list(id=1:nrow(data),words=words))

dim(spm)
```

```
[1] 13319 1125
```

```
# 13319 reviews using 1125 words

# Create a dense matrix of word presence
```

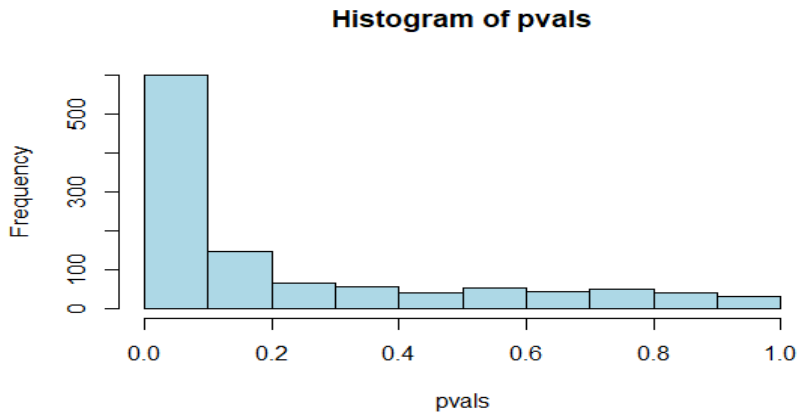
```r
P <- as.data.frame(as.matrix(spm>0))

library(parallel)

margreg <- function(p){
    fit <- lm(stars~p)
    sf <- summary(fit)
    return(sf$coef[2,4])
}

# The code below is an example of parallel computing
# No need to understand details now, we will discuss more later

cl <- makeCluster(detectCores())

# Pull out stars and export to cores

stars <- data$Score

clusterExport(cl,"stars")

# Run the regressions in parallel # MRG

pvals <- unlist(parLapply(cl,P,margreg))

# If parallel stuff is not working,
# you can also just do (in serial):
# mrgpvals <- c()
# for(j in 1:1125){
#   print(j)
#   mrgpvals <- c(mrgpvals,margreg(P[,j]))
# }
# make sure we have names

names(mrgpvals) <- colnames(P)

# The p-values are stored in mrgpvals
```

Homework Questions:

(1) Plot the p-values and comment on their distribution. (1 point)

- There is a spike around zero. This indicates a signal that there may be discoveries of covariates that reject the null hypothesis because there are more p-values close to zero than expected under the null.

```r
hist(mrgpvals,col="lightblue",breaks=10)
```



(2) Let's do standard statistical testing. How many tests are significant at the alpha level 0.05 and 0.01? (1 point)

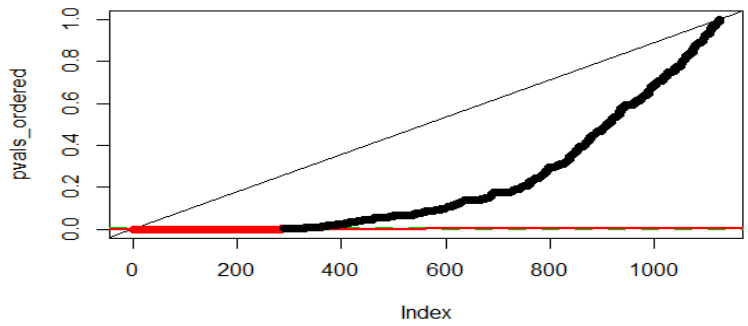
- Tests significant at alpha level 0.05: 461
- Tests significant at alpha level 0.01: 348

```
table(mrgpvals<=0.05)
table(mrgpvals<=0.01)
```

(3) What is the p-value cutoff for 1% FDR? Plot and describe the rejection region. (1 point)

- The p-value cutoff for 1% FDR is 0.0024
- The plot indicates that the p-values located below the red line, highlighted as red dots, are in the rejection region. This means that the variable with those p-values are potential signals that reject the null hypothesis.

```
pvals_ordered<-pvals[order(mrgpvals,decreasing=F)]

p<-1125

{plot(pvals_ordered,pch=19)
    abline(0,1/p)}

q<-0.01

source("fdr.R")

cutoff <- fdr_cut(mrgpvals, q)
cutoff

signif <- pvals_ordered <= cutoff

{plot(pvals_ordered,pch=19)
    abline(0,1/p)
    abline(h=cutoff,lty=2,col=3,lwd=3)
    abline(0,q/p,col=2,lwd=2)
    points(pvals_ordered,
       col=signif+1,pch=19)} # The red dots are discoveries
```



(4) How many discoveries do you find at q=0.01 and how many do you expect to be false? (1 point)

- Discoveries found at q=0.01: 290
- Number of expected false discoveries: 290*0.01 ≈ 3

```
table(mrgpvals<=cutoff) # number of discoveries and non-discoveries
```

(5) What are the 10 most significant words? Do these results make sense to you? What are the advantages and disadvantages of our FDR anaysis? (1 point)

- 10 most significant words: not, horrible, great, bad, nasty, disappointed, new, but, same, poor
- Based on the results we can see that usually the presence of words that carry strong negative sentiments have significant effect on the review rating, which seems reasonable.
- This FDR analysis is effective in showing which words are individually significant on the rating score and allows us to control for the problem of multiplicity.
- However, it does not account for the potential that the individual tests may not be independent. Realistically it seems difficult to assume that the tests are completely independent of each other.
- Another limitation is that this analysis does not account for how the frequency of word usage affects review rating in conjunction with the effect of individual words.

```
names(pvals)[order(mrgpvals)[1:10]]
```