Stat 243
Problem set 3

A. Strand, ID: 540441, GitHub: AndreasStrand

September 28, 2015

Problem 1

(b) I chose to read paper *i.*, By Gentzkow and Shapiro.

(c) Will there be exercises in the Statistical Computing course that includes normalizing tables in a database and merging such tables in order to do computations?

Problem 2

```
##################### R-code ###################
library(XML)
library(curl)
library(stringr)

# (a)
# Retrieving the URL of the requested debates
url = htmlParse('http://www.debates.org/index.php?page=debate-transcripts')
nodes = getNodeSet(url, "//a[@href]")
debates = sapply(nodes, xmlGetAttr, "href")
years = seq(1996,2012,4)
yearIdxs = sapply(years,function(x
) grep(paste(pattern = x,'-debate', sep = ""), debates)[1])
relevantDebates = sapply(debates[yearIdxs],htmlParse)

# (b)
# Trying to extract the relevant text from the transcript urls.
# One solution would be to extract the text part within
# the node //p/p, using xmlValue, but I could not figure
# out how to extract an XMLNode from a XMLNodeSet in
# order to use xmlValue().
textNodes = sapply(relevantDebates, getNodeSet, "//p/p")
text = sapply(textNodes, function(x) xmlValue(x)) # Does not work.

# Instead the transcripted were manually saved in txt-files in
# the current folder with their origin year as name.

# (c)
# Cleaning blank lines, and non-spoken text.
files= sapply(years, function(x) readLines(paste(x, ".txt", sep = "")))
cleanFiles1 = sapply(files,function(x) x[grep('^[a-zA-Z]', x)])
concIdx = sapply(cleanFiles1,function(x) grep('^[A-Z]{2}', x))

# Concatenating preceding chunks
cleanFiles2 = list()
for (i in 1:length(concIdx)){
  j=1
  str=character(0)
  while (j <= tail(concIdx[i], n=1)){
```

```
    str = paste(str, readLines(cleanFiles1[i,j], n=1),sep = "\n")
    if (match(j,conxIdx[i])){
      cleanFiles2[i] = str
      }
    j = j+1
  }
} # Error: (list) object cannot be coerced to type 'double'

# (d)
# Creating vectors with single words or sentences.
# Using cleanFiles1 due to an error in cleanFiles2.
wordFiles = sapply(cleanFiles1, function(x) unlist(strsplit(x, " ")))
sentFiles = sapply(cleanFiles1, function(x) unlist(strsplit(x, "(?<=[a-zA-Z][\\.\\?])",

# (e)
# Creating matrix of candidates of each debate manually.
candidates = matrix(c("CLINTON","GORE","KERRY","MCCAIN","ROMNEY","DOLE","BUSH","BUSH
                    nrow=5, ncol=2)
# The answer will not be completely right when using cleanFiles1
# since the grep will not capture the complete chunks.
# Ideally I would use cleanFiles2.
firstCands = list()
secCands = list()
for (i in 1:length(cleanFiles1)){
  firstCands[i] = cleanFiles1[i,grep(candidates[i,1], cleanFiles1[i])]
  firstCands[i] = cleanFiles1[i,grep(candidates[i,2], cleanFiles1[i])]
}
# The for-loop do not do what I indended. But the plan for the rest of
# Problem two is to do the procedure in (d) for the first and the second
# Candidate, and then use grep and sum on the word vectors to find the word
# count on the requested words. My guess is that the replublican candidates
# use the religious words more frequent.
```

Problem 3

I did unfortunately not have enough time to work through problem 3, but I will try to catch up during the week.