Stat243: Problem Set 3, Due Wednesday Oct. 1

September 23, 2014

Comments:

- This covers Unit 4, Section 8.
- It's due at the start of class on 10/1.
- As usual, simply providing the raw code is not enough; make sure to describe how you approached the problem and the steps you took. Also, please show us example output of what your code produces, but be selective - don't give us all of the output if there is a lot of it.
- Please note my comments in the syllabus about when to ask for help and about working together.
- As discussed in the syllabus, please turn in (1) a copy on paper, as this makes it easier for us to handle AND (2) an electronic copy through Git following Jarrod's instructions.
- There is only one problem but it's extensive, and there may be hurdles to surmount that will take you some time, so start early.

The goal of this PS is two-fold: first to give you practice with regular expressions and text manipulation and the second to have you thinking about writing well-structured, readable code. Regarding the latter, please focus your attention on writing short, modular functions that operate in a vectorized manner and also making use of *apply()*/*lapply()*/*sapply()* to apply functions to your data structures. Think carefully about how to structure your objects to store the speech information. You might have each speech be an element in a character vector or in a list.

1. The website The American Presidency Project at UCSB has the text from all of the State of the Union speeches by US presidents. (These are the annual speeches in which the president speaks to Congress to report on the situation in the country.) Your task is to process the information and produce data on the speeches. Note that while I present the problem below as subparts (a)-(i), your solution does NOT need to be divided into subparts in the same way. Your solution should do all of the downloading and processing from within R so that your operations are self-contained and reproducible.

    (a) From the website, you need to get the HTML file for each of the speeches. You'll need to start with this HTML file and extract the individual URLs for each speech. Then use that information to read each speech into R. You may need to use *readLines()* or *scan()* to read the HTML file into R and then need to concatenate the relevant lines back together. Or functions in the XML or other packages might be of assistance.

    (b) For each speech, write a function that extracts the body of the speech. You should also retain the name of the president and the year of the speech.

(c) Write a function that converts the text so that each speech is stored as a single character vector, with "\n" for line endings (i.e., the <p> from the HTML), and with all non-text stripped out (e.g., the tags for 'Laughter' and 'Applause'), including HTML codes. For the Laughter and Applause tags, retain information about the number of times it was occurred in the speech. The end result of your processing is that if you use the *cat()* function on the body of a given speech, it should print out the spoken text of the speech in a nicely-formatted manner.

(d) Write a function that extracts the words from a speech as individual elements of a (rather long) character vector. Write a function that takes a speech and extracts a count of all the words.

(e) Write a function that takes a speech and extracts to a character vector where each element is a separate sentence.

(f) Now, for each speech, extract data that allows an analyst to analyze how the speeches have changed over time. Here are some features of interest:

   i. Length in words and sentences.
   ii. Average word and sentence lengths.
   iii. Counts of the following words or word stems: I, we, America{,n}, democra{cy,tic}, republic, Democrat{,ic}, Republican, free{,dom}, war, God [not including God bless], God Bless, {Jesus, Christ, Christian}, and any others that you think would be interesting.

(g) The result of all of this activity should be well-structured data object(s) containing the information about the speeches: each speech as a single string, the vector of sentences, the vector of words, the word and sentence counts, and the additional quantification of variables about the speech from (f) as well as the laughter and applause variables from (c).

(h) Make some basic plots that show how the variables have changed over time and (for presidents since Franklin Roosevelt in 1932) whether they seem to differ between Republican and Democratic presidents (the Republican presidents have been {Eisenhower, Nixon, Ford, Reagan, G.H.W. Bush, G.W. Bush} and the Democrats have been {Roosevelt, Truman, Kennedy, Johnson, Carter, Clinton, Obama}. Your response here does not have to be extensive but should illustrate what you would do if you were to proceed on to do extensive exploratory data analysis.

(i) Extra credit: Do some additional research and/or additional thinking to come up with additional variables that quantify speech in interesting ways. Do some plotting that illustrates how the speeches have changed over time.

Hints: (1) There are characters in the text that are not standard ASCII characters. You'll likely need to use the R function *iconv()* so that the non-standard characters are human-readable or omitted - see Unit 3. (2) Depending on how your process the speeches, you may end up with lists for which the name of a list element is very long, such as the entire text of the speech or the entire HTML. Syntax such as `names(myObj) <- NULL` may be helpful.