STAT GR5206 Homework 3 [40 pts]
Due 11:59pm Thursday, October 11th on Canvas

Your homework should be submitted on Canvas as an R Markdown file. Please submit the knitted .pdf (or .html) file **along with the .Rmd file**. We will not (and cannot) accept any other formats. Please clearly label the questions in your responses and support your answers by textual explanations and the code you use to produce the result. Note that you cannot answer the questions by observing the data in the "Environment" section of RStudio or in Excel – you must use coded commands.

Goals: regular expressions, character functions in `R`, and web scraping.

In this assignment, we're going to scrape the 2018-2019 Brooklyn Nets Regular Season Schedule (they're a basketball team from Brooklyn that plays in the NBA). We will take the regular season schedule from `http://www.espn.com/` and reassemble the game listings in an `R` data frame for computational use.

To do this, perform the following tasks:

i. Open the link `http://www.espn.com/nba/team/schedule/_/name/BKN/seasontype/2`. Save the page as `NetsSchedule1819` using a .html extension. Once the file is saved, check that you can open the file by a text editor or import it in `R`.

ii. Use the `readLines()` command we studied in class to load the `NetsSchedule1819.html` file into a character vector in `R`. Call the vector `nets1819`.

   a. How many lines are in the `NetsSchedule1819.html` file?

   b. What is the total number of characters in the file?

   c. What is the maximum number of characters in a single line of the file?

iii. Open the webpage. You should see a table listing all the games scheduled for the 2018-2019 NBA season. There are a total of 82 regular season games scheduled. Who and when are they playing first? Who and when are they playing last?

iv. Open `NetsSchedule1819.html` using your browser and again look at its source code. What line in the file holds information about the game of the regular season (date, time, opponent)? It may be helpful to use `CTRL-F` or `COMMAND-F` here and also work between the file in `R` and in the text editor.

Using `NetsSchedule1718.html` we'd like to extract the following variables: the date, the game time (ET), the opponent, and whether the game is home or away. Looking at the file in the text editor, locate each of these variables. For the next part of the homework we use regular expressions to extract this information.

v. Write a regular expression to extract the line that contains the time, location, and opponent of all games.

vi. Write a regular expression to split the whole line into 82 lines, with each line displaying the information of one game. (You may obtain some hint from problem (vii).)

vii. Write a regular expression that will capture the date of the game. Then using the `grep()` function find the lines in the file that correspond to the games. Make sure that `grep()` finds 82 lines, and the first and last locations `grep()` finds match the first and last games you found in (ii).

viii. Using the expression you wrote in (vii) along with the functions `regexpr()` and `regmatches()`, extract the dates from the text file. Store this information in a vector called `date` to save to use below. HINT: We did something like this in class.

ix. Use the same strategy as in (vii) and (viii) to create a `time` vector that stores the time of the game.

x. We would now like to gather information about whether the game is home or away. This information is indicated in the schedule by either an '@' or a 'vs' in front of the opponent. If the Nets are playing '@' their opponent's court, the game is away. If the Nets are playing 'vs' the opponent, the game is at home.

Capture this information using a regular expression. You may want to use the HTML code around these values to guide your search. Then extract this information and use it to create a vector called `home` which takes the value 1 if the game is played at home or 0 if it is away.

HINT: In my solution, I use the fact that in each line, the string `<div class=\"flex items-center opponent-logo\"><span class=\"pr2\">` appears before this information. So my regular expression searches for that string followed by '@' or that string followed by 'vs'. After I've extracted these strings, I use `substring()` to finally extract just the '@' or the 'vs'.

xi. Finally we would like to find the opponent, again capture this information using a regular expression. Extract these values and save them to a vector called `opponent`. Again, to write your regular expression you may want to use the HTML code around the names to guide your search.

xii. Construct a data frame of the four variables in the following order: `date`, `time`, `opponent`, `home`. Print the frame from rows 1 to 10 Does the data match the first 10 games as seen from the web browser?