Stats 102A - Homework 3 Instructions

Homework 3 Requirements

You will submit two files.

The files you submit will be:

1. `102a_hw_03_output_First_Last.Rmd` Take the provided R Markdown file and make the necessary edits so that it generates the requested output. The first line of your .Rmd file should be to source the R script file you wrote.
2. `102a_hw_03_output_First_Last.pdf` Your output PDF file. This is the primary file that will be graded. Make sure all requested output is visible in the output file.

There is no script file to submit

Failure to submit all files will result in an automatic 40 point penalty.

Academic Integrity

At the top of your R markdown file, be sure to include the following statement after modifying it with your name.

"By including this statement, I, Joe Bruin, declare that all of the work in this assignment is my own original work. At no time did I look at the code of other students nor did I search for code solutions online. I understand that plagiarism on any single part of this assignment will result in a 0 for the entire assignment and that I will be referred to the dean of students."

If you collaborated verbally with other students, please also include the following line to credit them.

"I did discuss ideas related to the homework with Josephine Bruin for parts 2 and 3, with John Wooden for part 2, and with Gene Block for part 5. At no point did I show another student my code, nor did I look at another student's code."

Reading:

Read the following:

a. tidy data: https://r4ds.had.co.nz/tidy-data.html
b. data transformation: https://r4ds.had.co.nz/transform.html
c. How dplyr replaced my most common R idioms: http://www.onthelambda.com/2014/02/10/how-dplyr-replaced-my-most-com
d. regular expressions tutorial http://regexone.com/

Part 1 - dplyr exercises

Using `dplyr`, answer the following questions about the `vehicles` dataset. You are allowed to use non-dplyr functions, but I do believe dplyr makes the questions much easier.

a. How many unique vehicle makers (variable `make`) are included in the dataset?
b. How many vehicles made in 2014 are represented in the dataset?
c. For the year 2014, what was the average city mpg (gas mileage) for all compact cars? What was the average city mpg for midsize cars in 2014?
d. For the year 2014, compare makers of midsize cars. Find the average city mpg of midsize cars for each manufacturer. For example, in 2014, Acura has 5 midsize cars with an average city mpg of 20.6, while Audi has 12 midsize cars with an average city mpg of 19.08. Produce a table showing the city mpg for 2014 midsize cars for the 27 manufacturers represented in the table. Arrange the results in descending order, so that the manufacturer with the highest average mpg will be listed first.

e. Finally, for the years 1994, 1999, 2004, 2009, and 2014, find the average city mpg of midsize cars for each manufacturer for each year. Use tidyr to transform the resulting output so each manufacturer has one row, and five columns (a column for each year). Print out all the rows of the resulting tibble. You can use `print(tibble, n = 40)` to print 40 rows of a tibble.

Make sure your output is visible for each question

Part 2 - more dplyr

I have uploaded a dataset called `dr4.Rdata`. It contains the dates that fictional users visited a fictional website. The website is able to track if the same user visited the site more than once. For the particular date range, the site had 395 visitors, and 130 of them visited more than once. Some of them (13 people) visited the site 5 times.

Using dplyr, find the average time between repeated visits to the site.

You will want to find the total average.

Be careful when calculating this.

For example, the first user to visit the site more than once (row 2, ,YPELGRZNOQUTNPOH) visited on 6-29, 7-27, 8-3, and 8-11. The time difference for the repeated visits are: 28 days, 7 days, and 8 days, respectively, for an average of 14.33 days.

The next user with repeated visits is row 3 (SNTCUXUDIHCCSPJA). This person visited on 6-15 and 8-17, a difference of 63 days.

If your dataset had only these two rows, the average time between visits would be (28 + 7 + 8 + 63) / 4 = 26.5 days. It is not (14.33 + 63) / 2 = 38.66 days.

When I first attempted this, I used `filter(), mutate(), rowwise(), ungroup()`, and `summarise()`. Upon further review, I realized that it is entirely possible to complete this task using only `filter()` and `mutate()` commands. I do not care what combination of commands you use. I do care that you get the correct final result.

Make sure your final output shows the desired average number of days between visits.

Part 3 - rvest

You will use the package rvest to scrape data from the website baseball-reference.com.

Begin at the teams page http://www.baseball-reference.com/teams/.

For each active team (there are 30), visit each team's page and download the "Franchise History" table. The node you will want to use is `#franchise_years`. Combine all the tables in one. Note that some franchises have names and locations. To keep track of the team, add a column to the dataframe called "current_name" which will contain the current name of the team. (For example, in the 'current_name' column, the row for 1965 Milwaukee Braves will contain the value 'Atlanta Braves')

Hint: After identifying the node, I used the function `html_table()` to extract the table from each team's page.

Important: *It is bad manners to repeatedly hit a site with http requests, and could cause your IP to become banned. While you are testing out your code, be sure to test with only two or three teams at a time. Once you get your code running, then you may expand your code to download data for all 30 teams.*

After scraping the data, print out the dimensions of the resulting data.

Hint: When I ran my code, my table had 2684 rows and 22 columns.

Also print out the first few rows of the resulting data.

Part 4 - dplyr to summarize the baseball data

Unfortunately the baseball-reference site makes use the of the non-breaking space character and uses it in places like the space in "Atlanta Braves."

I've written some commands for you in the Rmd file that will replace all instances of the non-breaking space and replace it with a standard space character in the baseball table. I've done this part for you. You just need to run the code in the section **Some light text clean up**

Once you have created your table, use the data it contains to calculate some summary statistics.

For each franchise, filter the dataset to only include data from the years 2001 to 2019 (inclusive). If the franchise changed team names during this period, include the previous team's data as well. (e.g. the data for the Washington Nationals will also include data for the 2001-2004 Montreal Expos)

Then calculate the following summary statistics for each team across the 19 seasons:

- *total wins (TW)*
- *total losses (TL)*
- *total runs scored (TR)*
- *total runs allowed (TRA)*
- *total win percentage (wins / (wins + losses))*

Sort the resulting table (should have a total of 30 rows) in descending order by total win percentage. Be sure to print all rows and columns of the resulting summary table.

All requested columns must appear in the html to receive full credit.

Hint: At the top of my table, I had the NY Yankees, with a total win percentage of 1799 Total Wins, 1276 Total Losses, and a Total Win Percentage of 0.5850407.

Part 5 - Regular expressions for the Manager column

Using regular expressions, extract the wins and losses for the managers listed in the managers column. Do not use each season's number of wins or losses. You must extract the information from the managers column using regular expressions. That column has the information written in the form "F.LastName (82-80)". You will need to use capture groups in your regular expression to separate the different pieces of information.

Be careful as some of the rows contain information for more than one manager. Combine all of the manager information to get a total wins and loss value for each of the managers. Many managers have managed more than one team. Be sure to combine all of the win-loss information for the same manager. You may assume that entries that share the same first initial and last name are the same person.

Create a summary table with one line for each manager. The table should contain the following columns, and should be sorted descending by total number of games.

- *Manager's name (First initial and Last Name)*
- *Total number of games managed*
- *Total number of wins across career*
- *Total number of losses across career*
- *Total win percentage*

You can independently verify if your information is correct on baseball-reference.com. Each manager has his own page with a total count of wins and losses.

Figuring out the regular expression here is probably the trickiest part. There is also an instance where there are two different people with the same first initial and the same last name. Unfortunately, their information will end up being combined. For this homework assignment, that's okay.

Regarding the regular expression, you will need to use capture groups, and thus `str_match_all()`. We use the `_all` variant because some of the entries will have multiple managers.

All requested columns must appear in the html to receive full credit.

The first line of my table reads: C.Mack, 7679, 3731, 3948, 0.4858706, for manager, games, wins, losses, win percentage.

Also Watch out for T.La Russa who has a space in his name. He managed the second most number of games with a final record of 2728-2365. If you report his name as `T.La`, you will not get full credit.

Part 6 - Extra credit

This is completely optional. Up to 10 points.

IMDB webscraping and summarization. You will need to add this section to the Rmd file yourself.

This is the IMDB page for actor Keanu Reeves. https://www.imdb.com/name/nm0000206/?ref_=nv_sr_srsg_0

The task is to follow the links to all of the projects he had a role in from 2019 and earlier (*Hangin In* through *Between Two Ferns: The Movie*).

From each movie page, follow the link to *See full cast.* From that page, scrape all members of the cast (credited and uncredited). Do not scrape director, writer, or crew information.

After gathering this data, create a summary table of the actors that Keanu Reeves has worked with.

+7 points if you can identify the actors that have appeared in 2+ projects with Keanu Reeves. The table should include the total number of projects they appeared in together arranged in descending order. Actors will be arranged alphabetically for ties.

+3 additional points if your table can list the names of the projects. How you format and present this is up to you.

For example, the entry for Laurence Fishburne might look like this: (by the way, this example might not be accurate)

```
## Laurence Fishburne  6
## ....
```

Would yield 7 points

```
## Laurence Fishburne  6  "The Matrix"
##                        "The Matrix Reloaded"
##                        "The Matrix Revolutions"
##                        "Enter the Matrix"
##                        "John Wick: Chapter 2"
##                        "John Wick: Chapter 3 - Parabellum"
## ....
```

Would yield 10 points