Biostat 561: Homework 1

Instructor: Amy Willis, Biostatistics, UW

03 April, 2019

Instructions and preliminaries

Question 0: Due by the end of office hours (Monday 3:30pm) via github classroom (instructions below).

Questions 1-7: Due 2:30pm Wednesday 10 April via GitHub Classroom (instructions below).

- You may discuss the questions with other students or the instructor, but you should write up your own answers in your own words. The only exception to this is Question 0 Parts 1-7, which you are welcome to discuss in full detail with classmates.
- Instructions for uploading homeworks are contained in Question 0. Please complete Question 0 by Monday, and if you have trouble, please bring your laptop to office hours on Monday at 2:30pm.
- Answer all questions in full sentences and paragraphs; supplying just code or its output is not a good-faith attempt to answer the question.
- Many of the questions in this homework were not discussed in class. They are research questions, intended to make you think about what happens "underneath" `R`. I expect you to make extensive use of online resources to complete these exercises.

Question 0

Please complete Steps 1-9 before Monday midday, and attend office hours if you are having trouble!

1. If you do not have `R` installed on your computer, install it. Google "how to install R" for instructions.

2. Install RStudio on your computer.

3. Download and install git and create an account on `github.com`. Choose a professional-sounding username! (avoid your birthdate/year/this year)

4. Create a folder on your computer for this class, e.g. "/Users/your_name/Documents/coursework/biost561/". Create 2 subfolders: "materials" and "responses".

5. At the command line (Terminal for Mac, `cmd.exe` for Windows), change directory into "/Users/your_name/Documents/coursework/biost561/materials/" and type "`git clone http://github.com/adw96/biostat561 .`"

   - Note that there is a full stop (.) in the command at the end!
   - Material for the class will be updated throughout the quarter. To update your materials to the latest copy (e.g., in a week), return to your materials folder and type `git pull`

6. Sign into our virtual classroom at classroom.github.com/classrooms/48962297-biostat561-2019 and create a repository with your *UW NetID* (not necessarily your github username) followed by "-561". eg. "adwillis-561" would be ideal for me.

7. Copy a blank (or random) pdf to "/responses/", and call it "hw1-response.pdf"

8. Read:

   - guides.github.com/activities/hello-world/
   - kbroman.org/github_tutorial/pages/init.html

- stackoverflow.com/questions/6143285/git-add-vs-push-vs-commit

9. Initialise a git repository in "/responses/" (your local copy) to "[your-name]-561" in github classroom (your remote copy). Add the pdf "homework1-response.pdf", commit the pdf, and push it to your classroom (remote) repository. Google (or similar) error messages you get. You may need to add an SSH key to your computer.

Work on your response to the remaining questions in this folder. Overwrite the blank pdf as you go through your responses. *Repeat Step 9 with your actual submission to submit your homework!*

Most of the problems that newbies have with git relate to being in the wrong directory! Use `pwd` (`echo %cd%` for Windows) and `git status` liberally!

Our repositories on github classroom are private, so your classmates cannot see your solution (but I can). You should practice adding, committing and pushing your homework as you work (e.g. as you finish each question) to get into good habits! I will not begin grading your solutions until after the deadline.

Question 3/4: RMarkdown

In our class github, there is a template for completing your homework submission in RMarkdown. If you open a `.Rmd` file in RStudio, you will be able to "knit" them easily. Have a look at `homework1.Rmd` to see how I produced this homework assignment using RMarkdown.

For more information on producing RMarkdown documents, read through rmarkdown.rstudio.com/lesson-1.html and subsequent lessons to learn how to interperse text and R code (with code and output included and suppressed). Note that pdf documents are only one type of document that can be produced with RMarkdown (html documents is another common option).

For all of your homework submissions for BIOST561, submit both your `Rmd` and your `pdf` files. There is no need to show *all* code and *all* output – use your judgement and be succinct.

Question 1: I don't know

`R`'s code for missing data is `NA`. Thinking of it as "I don't know" helps understand its behavior, e.g. `42+NA` returns `NA`; the sum of 42 and a number I don't know is another number I don't know.

1. Explain what these mathematical operations return, and why this is sensible:
    1. `FALSE & NA` – note that the "&" symbol is R's expression for a logical AND. See R's Logic help page if this is unfamiliar to you.
    2. `TRUE & NA`
    3. `TRUE | NA` – note that the "|" symbol is logical OR.
    4. `mean(c(3,5,7,NA))`
2. Why is the `is.na()` function needed to identify elements of a vector that are missing? In other words, explain why for vector

    ```
    myvec <- c(1,6,2,NA,500)
    ```

    we have to locate the missing value by using

    ```
    is.na(myvec)
    ```

    instead of

    ```
    myvec == NA
    ```

Question 2: Generic functions

Recall our example `plot(log)` from class. Executing

```
plot
```

doesn't tell you anything about plotting in `R`. To see the generic function `plot.function`, we need

```
graphics:::plot.function
```

The `:::` is needed because plot.data.frame is an `internal function`, i.e., not in the namespace of the R package `graphics`. We will revisit this in later lectures when we discuss R packages and namespaces.

Read through the source code for way that R plots objects of class `function`. In your own words, briefly describe what you think each line of the code does. If you are unsure about what any individual line does, say so.

Question 4: Complex numbers

While you will probably use them only rarely, `R` can handle complex numbers:

```
class(0+1i)
```

Euler's formula gives us that $e^{i\pi} + 1 = 0$. Can `R` confirm this? If not, what's going on? You should research the binary accuracy of `R` to develop your answer.

Question 5: Looping with the `apply` family

We discussed `apply()` looping over rows or columns of a dataset, but the "apply family" is much broader. Describe what `lapply()` and `mapply()` do, using your own words, and giving an example of them being used on one of the built-in datasets. `data()` lists built-in datasets.

Question 6: Writing your own function

A common data structure in microbiome science is a tab delimited file counting the number of times each micobial taxon was observed each sample. Here's an example:

https://raw.githubusercontent.com/adw96/stamps/master/FWS_OTUs.txt

Sometimes the information is transposed, but in this case the rows indicate the taxa and the columns indicate the sample.

Write a function, `microbial_diversity()`, that takes in a matrix or data frame of microbial counts (with rows = taxa and columns = samples) and returns the number of species observed at least once in each sample. Use `stop()` or `stopifnot()` to perform some checks that the given arguments to the function are of the correct form.

Write another function, `microbial_diversity_floor()`, with mandatory argument `counts` and optional argument `floor`, that counts the number of species observed at least `floor` times in each sample. Choose a sensible default value of `floor`.

Question 7: Bootstrapping

Suppose I am interested in my teaching evaluations from a class of 1000 students. However, I can only survey 5 students, who ranked my helpfulness as 1, 5, 8, 3 and 7 (on a scale from 1 to 10 – unbelievable, right?!). I'm interested in the median ranking of the evaluations of the 1000 students, but I'll have to get by with the information from the 5 students. My sample median score is 5, but how variable is that? If I had sampled 5 different students would I have had a very different score? (These are rhetorical questions!)

The (independent) bootstrap works by sampling observations with replacement from the dataset that was observed, and calculating the statistic of interest (here, the median) on the resample. By repeating this process many times, you end up with bootstrapped estimates of the median, and you can calculate the variance in these bootstrap estimates to approximate the out-of-sample variance (the variance that you would have observed if you could repeatedly sample 5 students).

Read about the function `sample()` and find out how to take a resample from the 5 observations (1, 5, 8, 3, 7) with replacement. Write a script to draw 10000 bootstrap resamples, and use these samples to give me an estimate of the variance of the median of my teaching scores.