Biostat 561: Homework 2

Instructor: Amy Willis, Biostatistics, UW

October 5, 2017

Instructions and preliminaries

See the specification for Homework 1 regarding academic integrity, expectations about answers, and where to find answers to questions on this homework that were not discussed in class.

Please complete Question 0 before Tuesday 10 October at 3pm, and attend office hours if you have trouble. The remaining questions are due Thursday 12 October at 2pm sharp.

Question 0: Easy version control with git

1. Go to your materials folder that you created last week and type `git pull`. This should download Lecture 2 and Homework 2 to your local copy, as well as the revised syllabus. You should repeat this command every week to give you the latest course information and resources.
2. Complete the following questions in your responses folder, then use the workflow described on Slide 2.23 to upload Homework 2 to github classroom.

This is the last homework where these instructions will be formally repeated, but all future homeworks should be submitted in this way.

Question 1: Writing your own function

Write a function that takes a vector of positive numeric data and does both of the following

- Plots a horizontal stripchart of the data, but also imposes a vertical line indicating where either the mean or median of the data lies. The user should be able to choose between these different measures of central tendency, and you should allow for an `na.rm` argument. `abline()`, `lines()` or `segments()` may be helpful for adding the line, so look up the documentation to see how they work.
- Returns a named factor giving both measures of central tendency and the number of missing observations.

Show me the output of

```
set.seed(1234)
my_data <- c(rgamma(50, shape = 5, rate = 1), NA)
your_function(data = my_data, na.rm = TRUE)
```

Question 2: Looping with functional programming

a) On Slide 2.22 we saw an example of a double loop, where the outer loop uses `by()` and is over subsets of the data, and the inner loop uses `apply()` and is over the columns of the data. Code the same example with the loops reversed, i.e., where the outer loop is over the columns in the inner loop is over the subsets. Verify that your answers match the slides.

b) 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 3: 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.

The bootstrap is a broadly applicable tool for the modern statistician. You will see it again and again in your graduate program.