BIOM262 Winter 2020. Homework #2: Module 3 (Statistics)

Note: For loading the appropriate libraries for the shiny app, there is a line at the top that says:

```
pacman::p_load(shiny,pander,markdown,stringr,grDevices)
```

You can just install and load each of these packages inside the parentheses as you would normally. Using the Install button and then load them by checking the checkbox or using library().

Homework Questions

Question 1

In class, we went over how to generate a null distribution and calculate a p-value for the statistic *difference of group means* to test for an association between a binary categorical variable and a quantitative variable. We also covered how to generate a confidence interval (see the R notebook on the web for how to find the endpoints of this interval using the built-in function `quantile()`.

For each of the following modifications listed below:

i. generate and plot a null distribution of the requested statistic and calculate the two-tailed p-value
ii. generate and plot a bootstrap distribution and report the confidence interval for the requested population parameter.

A two tailed p-value means that you find the fraction of the statistics in the null distribution that are more extreme than your actual statistic *in either direction*.

Use the `NHANES` dataset and do the appropriate selecting and filtering to prune it to what you want and to remove NAs.

Modification 1

Choose one categorical variable and one quantitative variable. If there are more than two unique values of your categorical variable, filter it down to two. Your statistic should be the **ratio of the variances** of the quantitative variables in the two groups.

The function in R to calculate the variance is `var()`.

Be sure to take a look at the slight modification to how to calculate the final statistic in the web notebook.

Note: When you calculate the p-value, think carefully about what the null hypothesis is - where the null parameter value would be if the variances were the same. Your actual statistic will lie on one side of the null parameter value. How would you find fraction of the distribution that is more extreme on the other side of the null value? Plotting the histogram might help.

BIOM262 Winter 2020. Homework #2: Module 3 (Statistics)

Question 1, *continued*

Modification 2

Choose two categorical variables. Filter each down to two values so that you are examining the relationship between two binary categorical variables. For example, let's say variable A has the possible values a1 and a2. Variable B has possible values b1 and b2. Choose one variable as the *grouping* variable, let's say variable A. For your statistic use the difference between the proportion of value b1 in the a1 group and the proportion of b1 in the a2 group.

Modification 3

Use your pruned dataset from Modification 1. Make up your own statistic [i.e. a new one] to compare the two groups. Write a function to calculate it. Your function should take a numeric vector as input and should return a single number as output. This statistic should summarize a property of the quantitative variables in one group. For your final statistic you will need to compare the values of your statistic in the two groups. [like the ratio in (Modification 1) or the difference in (Modification 2) or some other way]. If you designed your statistic to capture some property of your data [like the way the mean captures the center], then explain in a sentence or two.

Question 2

In the `NHANES` dataset use the variables Height and Pulse rate. These are both quantitative variables, so conventional statistics are correlation or regression to ask about association. For this problem, use **regression** and check whether you can predict pulse rate from height.

Regression is an example of a linear model. It finds the best fitting line to your data. You'll be seeing lots of linear models in this course, some more complicated than simple regression. For your calculating your statistic, the following might help: You can estimate a linear model in R in the following way:

```
linearModelResult <- lm(responseVariableName ~ explanatoryVariableName,
       data= dataFrameName)
```

Notice that the names are unquoted in the code above.

This returns a `lm` object which is an object with several different fields. In class we saw how to extract the slope of the best fitting line. Do the tasks for hypothesis testing and estimation for the slope of this relationship as in Question 1.