Week 4 test file

## R Markdown Example

This is an R Markdown document. In this plain-text section, you can write descriptions of what the document contains, what the analysis steps are, where you obtained the data, etc.

For this example, we could import the two SVLCallisaurus files, join them, and then produce a boxplot showing the snout-vent lengths of males and females separately.

### Setting the working directory

I don't bother setting a working directory when using RStudio to generate a Word or HTML document from this .Rmd file. The default behavior of RStudio when you hit the Knit button, is to use the .Rmd file's directory as the working directory. That means your directory paths to things like the data files need to be set relative to the storage directory of the .Rmd file.

# Here is a comment within a R code chunk  
# setwd("~/Dropbox/255E\_BIOL/Week\_4")

## Importing data

Next I'll import the two data files. Because the working directory used by RStudio (where this .Rmd file lives) is a sub-directory of my main 255E\_BIOL directory, I need to use the ../data/ syntax when trying to get to my data folder.

svl1 = read.csv("../data/SVLCallisaurus.csv")  
svl2 = read.csv("../data/SVLCallisaurus2.csv")

## Merging the two data sets

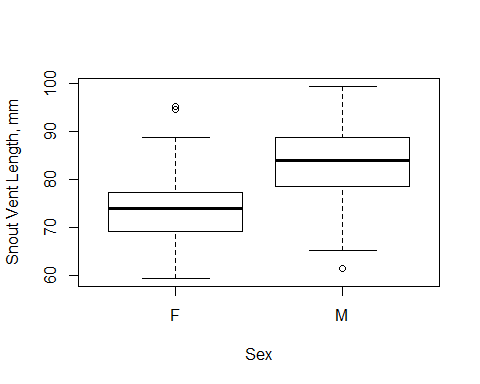
Because the two data sets share the same number of columns with the same data types in each, and the same column titles, we can used the rbind (row bind) function to merge them.

svlAll = rbind(svl1,svl2)

## Including Plots

Now we can make a boxplot showing the distributions of snout-vent length for males and females.

boxplot(Snout.Vent.Length..mm.~Sex, data = svlAll, xlab = 'Sex',  
 ylab = 'Snout Vent Length, mm')



Boxplot of salary distributions.

Note that the echo = FALSE parameter was added to the code chunk in the original .Rmd file to prevent printing of the R code that generated the plot.

## Including R results

We can also have R print out the results of a calculation to the output document, similar to how a R function might normally print its results to the console.

# Use aggregate function to calculate mean for each sex separately  
mymeans = aggregate(Snout.Vent.Length..mm.~Sex, data = svlAll,   
 FUN = mean)  
mymeans # print output

## Sex Snout.Vent.Length..mm.  
## 1 F 73.99388  
## 2 M 83.51368

We can also use the kable function to make a nicely formatted table out of a data frame.

kable(mymeans, caption = 'Mean lengths for each sex.')

Mean lengths for each sex.

|  |  |
| --- | --- |
| Sex | Snout.Vent.Length..mm. |
| F | 73.99388 |
| M | 83.51368 |