# Statistics 5014: Homework 3

Due Monday September 19, 10 am

*2018-09-10*

## Problem 4

In the lecture, there were two links to programming style guides. What is your takeaway from this and what specifically are *you* going to do to improve your coding style?

*The Google style guide has some good points, but in my practice I beleive it is a little to anal. I align more with the style guide of Hadley Wickham, due to the principles that here adheres to as well as is less detailed approach to the spacing. The biggest takeaway I get from both style guides is the spacing in long functions. This is probably somomething I will try to improve my style on as well. My current approach is to type out long complicated functions, but by spacing the complexity by lines, it will alow for easier to read code.*

## Problem 5

```
library(lintr)
lint(filename = "C:/Users/jwilliams10/Desktop/STAT Programming/STAT_5015_homework/HW2.Rmd")
```

From the messages, what are some things you need to change in your code?

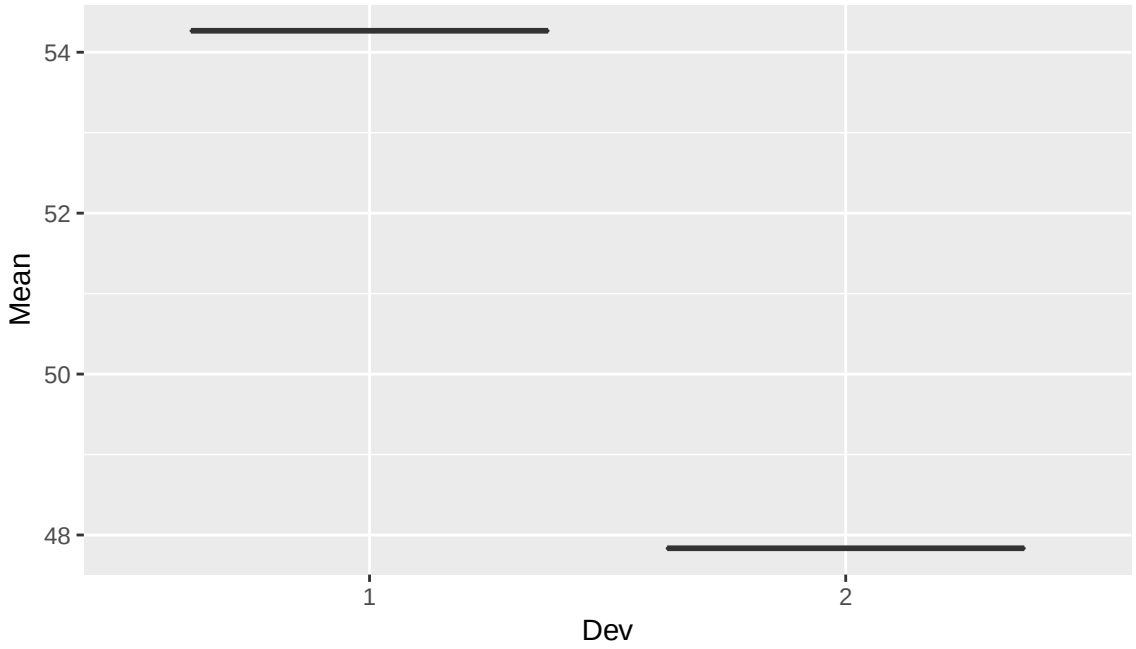
*I have alot of issues with not putting spaces after my commas. I should name functions with lowercase names. Place spaces around infix operators and reduce my lines of code to have less than 80 characters. Most of the errors that popped up were about my spacing.*

## Problem 6

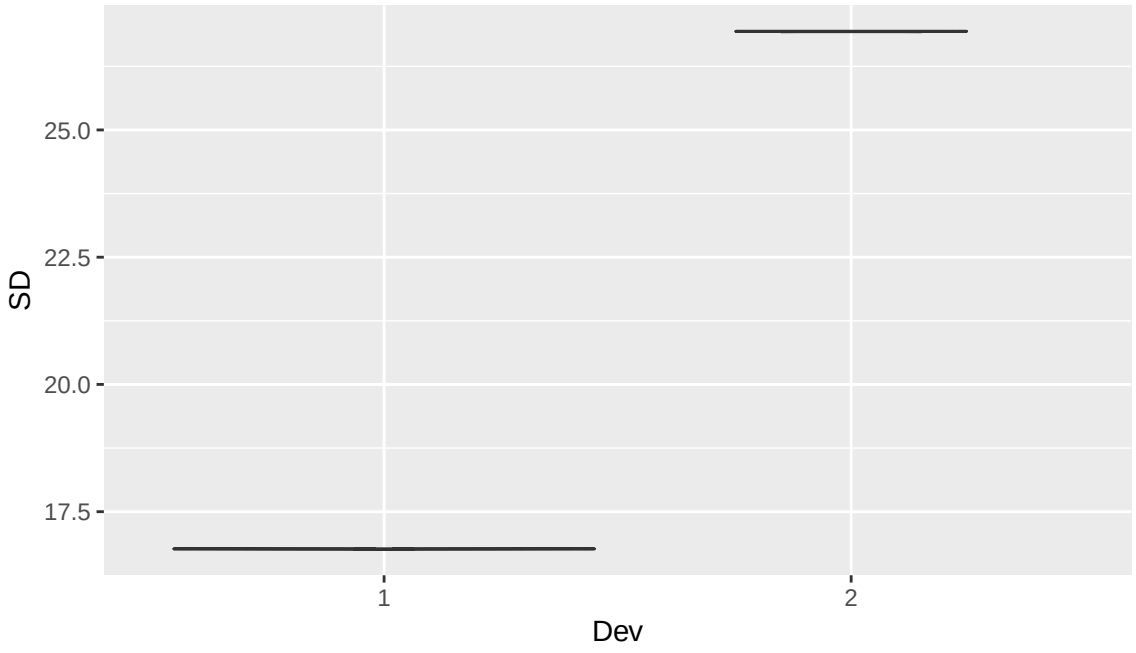
```
library(dplyr)
library(ggplot2)
Dat2<-readRDS("HW3_data.rds")
Dat2.1<- Dat2 %>% select(Observer,dev1,dev2) %>% group_by(Observer) %>% summarise(mean_dev1 = mean(dev1)
Dat2.1 #All correlations, means, and standard deviations
```

```
## # A tibble: 13 x 6
##    Observer mean_dev1 mean_dev2 sd_dev1 sd_dev2 cordevs
##       <dbl>     <dbl>     <dbl>   <dbl>   <dbl>   <dbl>
##  1        1      54.3      47.8    16.8    26.9 -0.0641
##  2        2      54.3      47.8    16.8    26.9 -0.0686
##  3        3      54.3      47.8    16.8    26.9 -0.0683
##  4        4      54.3      47.8    16.8    26.9 -0.0645
##  5        5      54.3      47.8    16.8    26.9 -0.0603
##  6        6      54.3      47.8    16.8    26.9 -0.0617
##  7        7      54.3      47.8    16.8    26.9 -0.0685
##  8        8      54.3      47.8    16.8    26.9 -0.0690
##  9        9      54.3      47.8    16.8    26.9 -0.0686
## 10       10      54.3      47.8    16.8    26.9 -0.0630
## 11       11      54.3      47.8    16.8    26.9 -0.0694
## 12       12      54.3      47.8    16.8    26.9 -0.0666
## 13       13      54.3      47.8    16.8    26.9 -0.0656
```

```r
Dat2.2<-data.frame(cbind(unlist(list(Dat2.1$mean_dev1,Dat2.1$mean_dev2)),unlist(list(Dat2.1$sd_dev1,Dat
colnames(Dat2.2) <- c("Mean","SD","Dev")
Dat2.2$Dev <- as.factor(Dat2.2$Dev)
ggplot(Dat2.2,aes(Dev,Mean))+geom_boxplot()
```




```r
ggplot(Dat2.2,aes(Dev,SD))+geom_violin(scale = "area")
```



## Problem 7 – redo

Same as last time, please create and annotate the process to create a tidy dataset from http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/BloodPressure.dat

If you didn't use the tidy functions last time, this is an opportunity to try them out now that you have thought through what the final format should be. If you did it using functions from the tidyverse, copy paste from last time.

```
library(data.table)
library(tidyr)
url1 <- "https://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/BloodPressure.dat"
Dat1 <- fread(url1,sep=" ")
Dat1 <- gather(Dat1,Day)
colnames(Dat1) <- c("Day","Reader","Data")
Dat1$Day <- as.factor(Dat1$Day) #Changed Day to Factor
Dat1$Reader <- as.factor(Dat1$Reader)
summary(Dat1) #Summary
```

```
##       Day      Reader        Data
##  1      : 6   Dev1:15   Min.   :110.8
##  2      : 6   Dev2:15   1st Qu.:125.5
##  3      : 6   Dev3:15   Median :130.4
##  4      : 6   Doc1:15   Mean   :129.0
##  5      : 6   Doc2:15   3rd Qu.:134.3
##  6      : 6   Doc3:15   Max.   :139.6
##  (Other):54
```

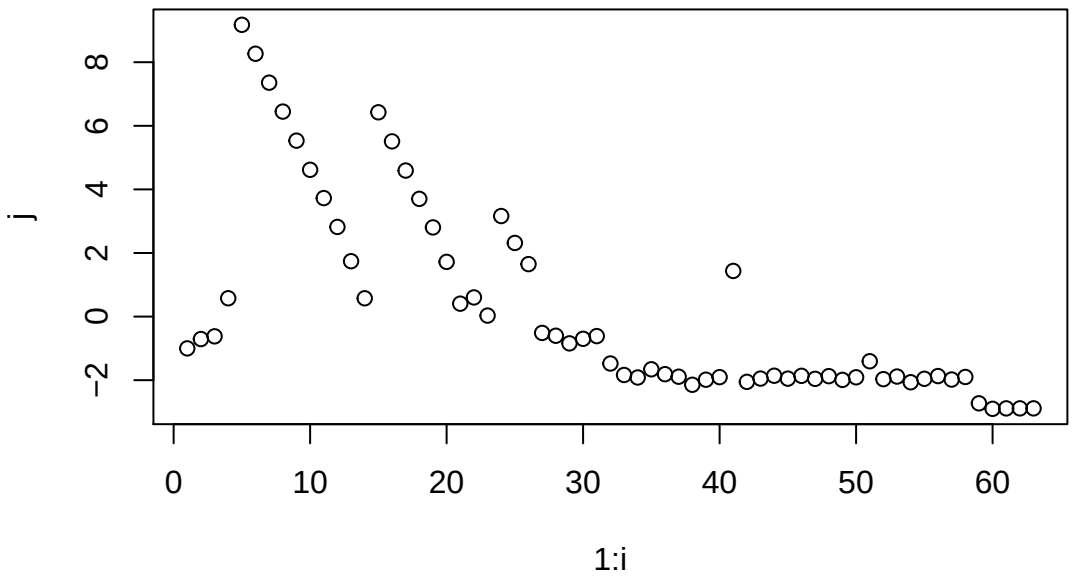
## Problem 8

Create a function to find solutions to (1) using Newton's method. The answer should include the solutions with tolerance used to terminate the loop and a plot showing the iterations on the path to the solution.

```
f <- function(x){
  3^(x) - sin(x) + cos(5*x)
}
g <- function(y){
  log(3)*(3^(y)) - cos(y) - 5*sin(5*y)
}

Newton_Method <- function(tol, intial_start){
j <- vector()
j[1] <- intial_start
i <- 1
while (abs(f(j[i])) > tol) {
  j[i+1] <- j[i] - f(j[i])/g(j[i])
  i <- i + 1
}

plot(1:i,j)
}
Newton_Method(.0000000000001,-1)
```




$$f(x) = 3^x - sin(x) + cos(5x) \tag{1}$$

## Problem 9

Finish this homework by pushing your changes to your repo and submitting me a pull request. In general, your workflow for this should be:

1. In R: git pull upstream – to make sure you have the most recent local repo
2. In R: do some work
3. In R: check files you want to commit
4. In R: commit, make message INFORMATIVE and USEFUL
5. In R: push – this pushes your local changes to the repo
6. In Github: submit a pull request – this tells me you are wanting me to pull in your changes to my master repo

If you have difficulty with steps 1-5, git is not correctly or completely setup.

**Only submit the .Rmd and .pdf solution files. Names should be formatted HW3_lastname_firstname.Rmd and HW3_lastname_firstname.pdf**

## Optional preperation for next class:

Next week we will talk about Exploratory Data Analysis and graphing. Swirl will be a bit part of this. Check out "Exploratory_Data_Analysis" Swirl lessons 1-10.