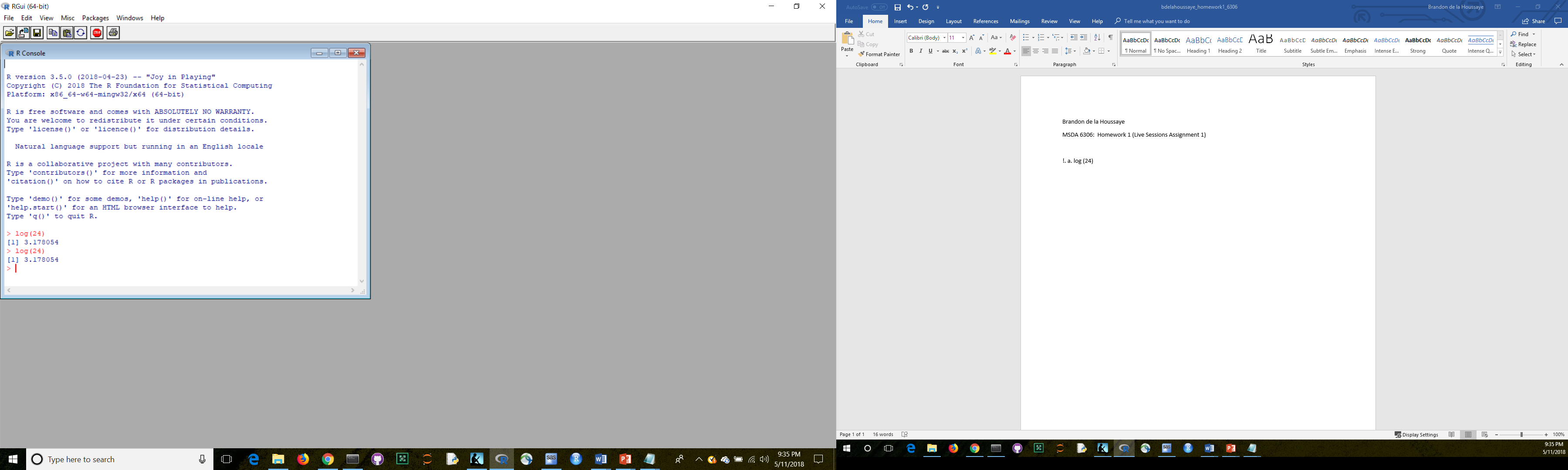
Brandon de la Houssaye

MSDA 6306 (4033): Homework 1 (Live Sessions Assignment 1)

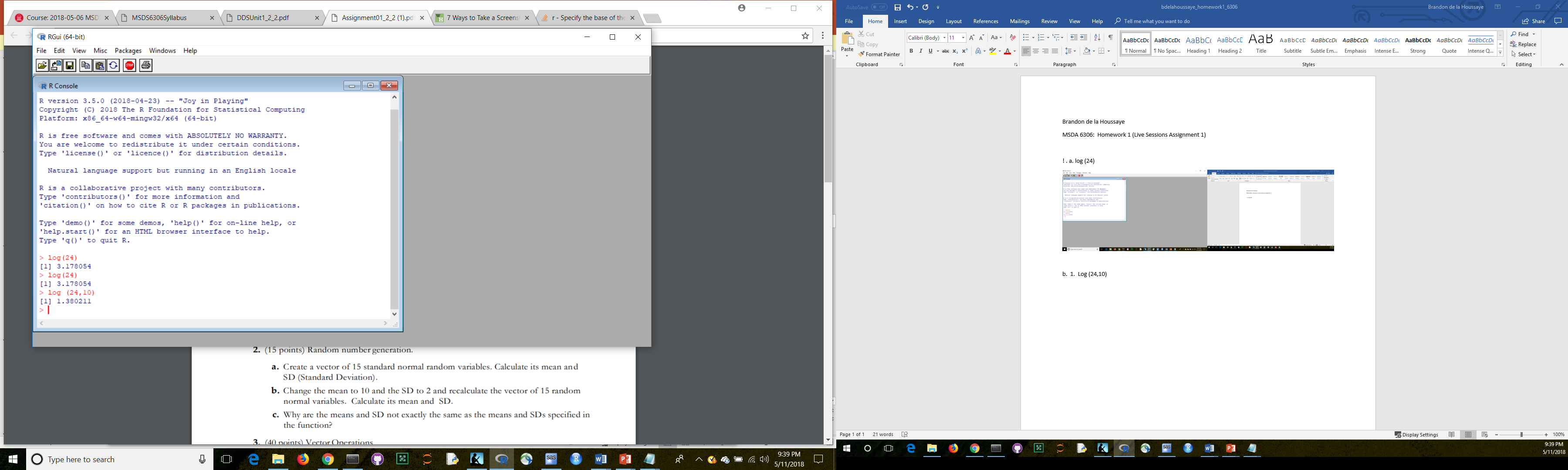
! . a. log (24)

[1] 3.178054



b. 1. Log (24,10)

[1] 1.380211



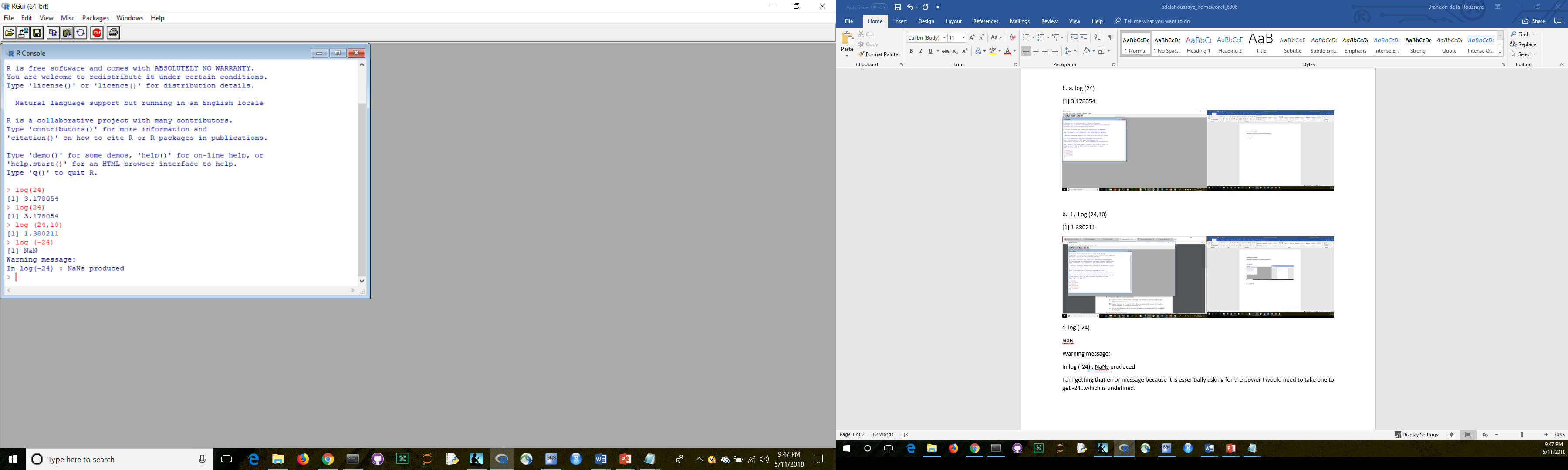
c. log (-24)

NaN

Warning message:

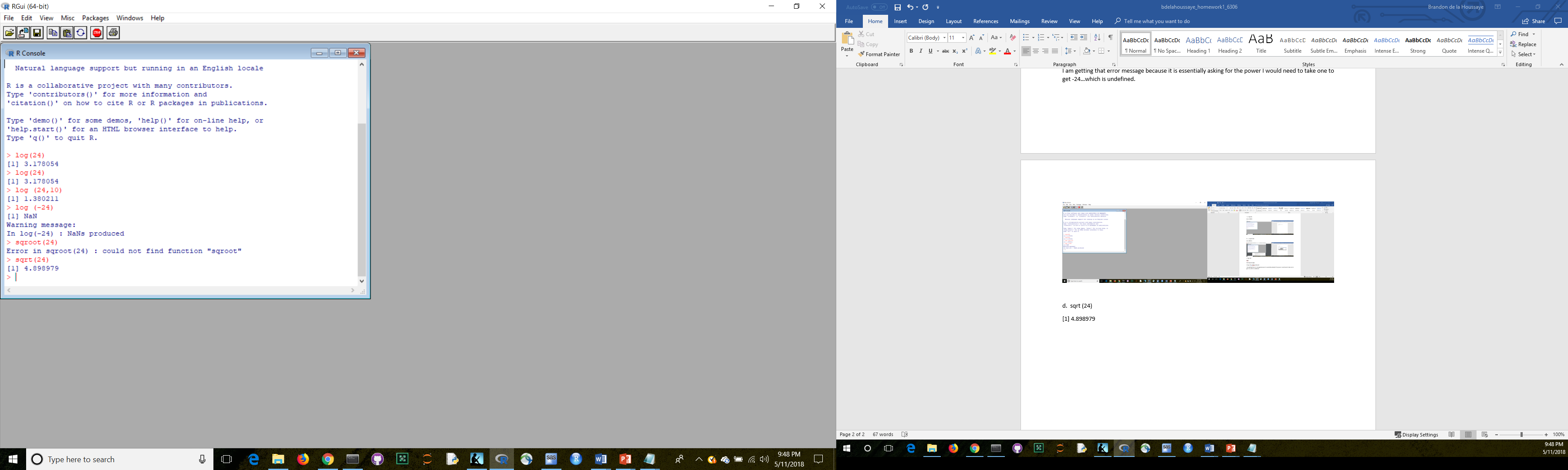
In log (-24) : NaNs produced

I am getting that error message because it is essentially asking for the power I would need to take one to get -24…which is undefined.



d. sqrt (24)

[1] 4.898979



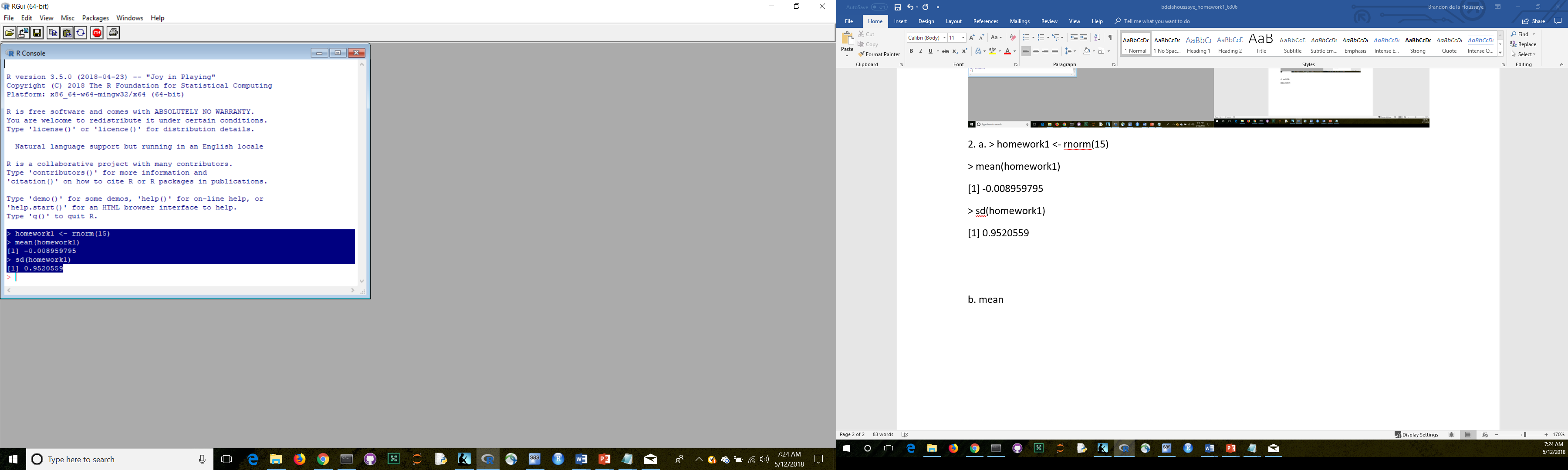
2. a. > homework1 <- rnorm(15)

> mean(homework1)

[1] -0.008959795

> sd(homework1)

[1] 0.9520559



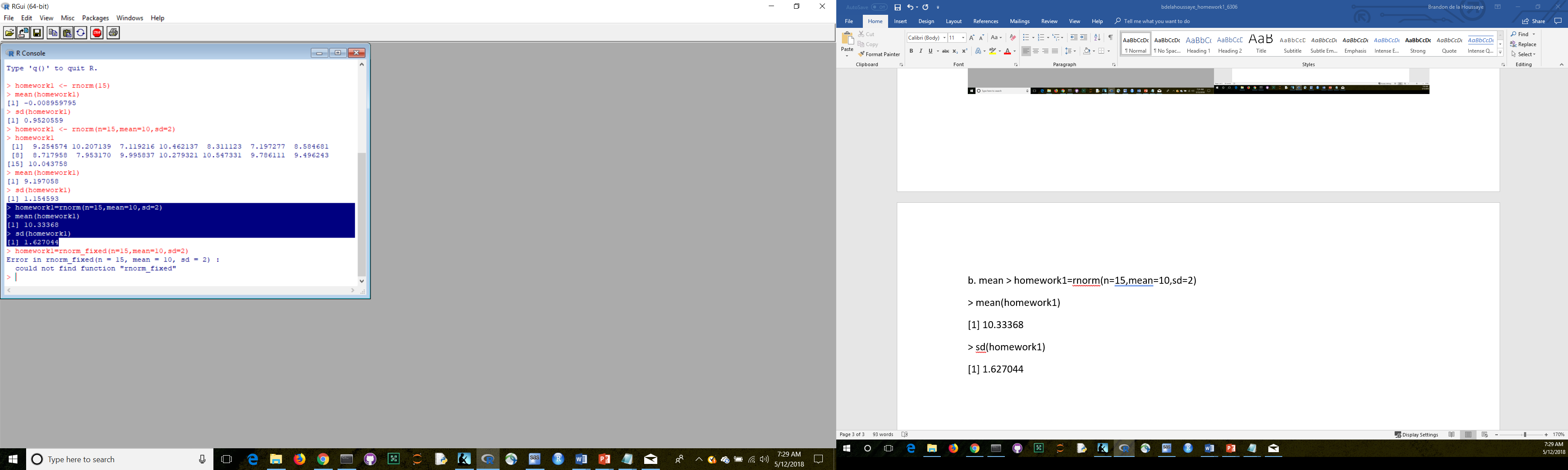
b. mean > homework1=rnorm(n=15,mean=10,sd=2)

> mean(homework1)

[1] 10.33368

> sd(homework1)

[1] 1.627044



c. The function specified parameters for the random set from within a population. This means that the defined population (from which to pull the random observations) had a specified mean of 10 and standard deviation of 2 and the results are normally distributed. However, when it is time to pull the random set (of n, or 15, observations) many of those observations may be from one tail or another. Thus, the observed mean and standard deviation would be different from those specified in the function. Importantly, as the sample size increases (i.e., n grows), I would expect the difference between the mean and standard deviation in the random sample and the population to decrease.

3. a. , b., and c.

> weight <- c(60,72,57,90,95,72)

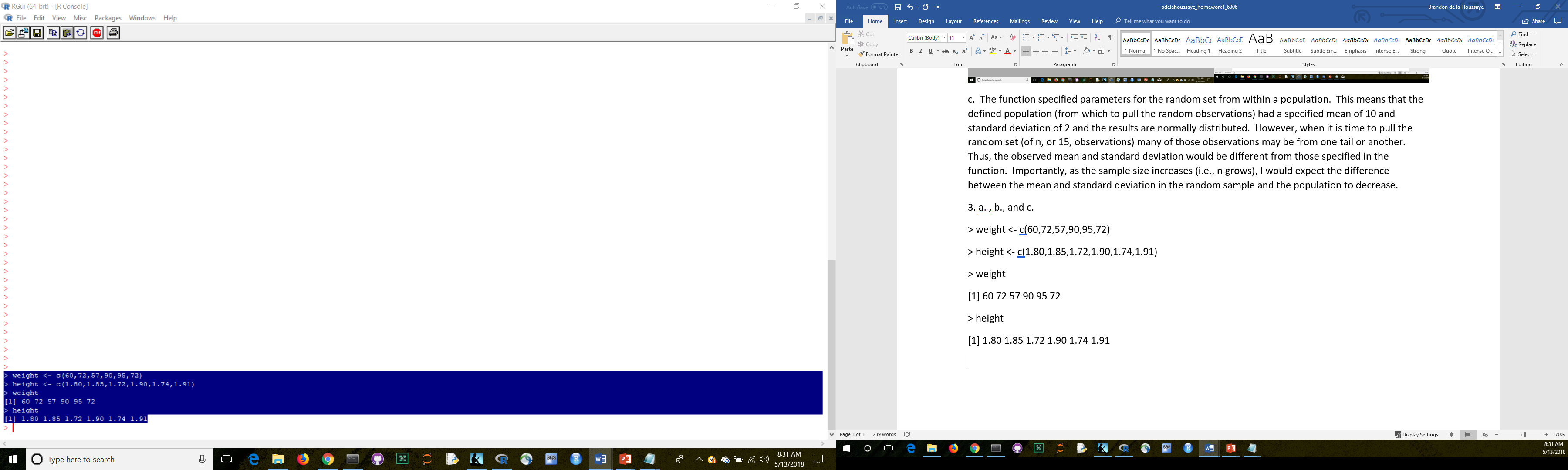
> height <- c(1.80,1.85,1.72,1.90,1.74,1.91)

> weight

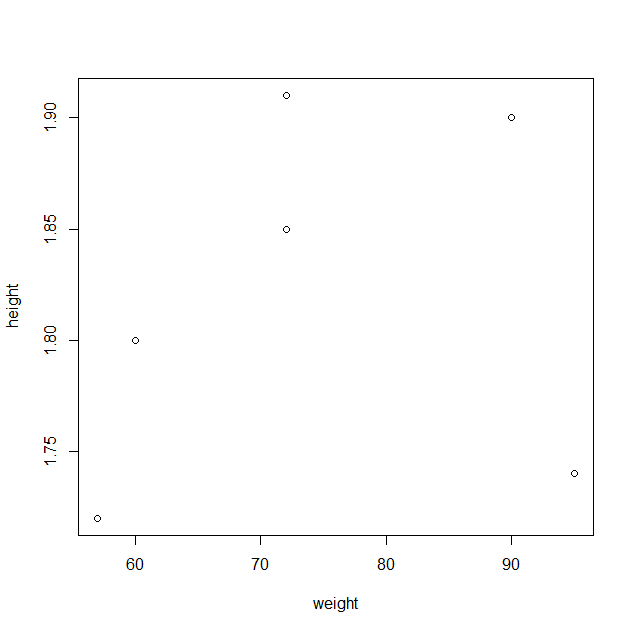
[1] 60 72 57 90 95 72

> height

[1] 1.80 1.85 1.72 1.90 1.74 1.91



d. > plot (weight, height)



e. > BMI = weight/height

> BMI

[1] 33.33333 38.91892 33.13953 47.36842 54.59770 37.69634

f. > WMean = mean(weight)

> WMean

[1] 74.33333

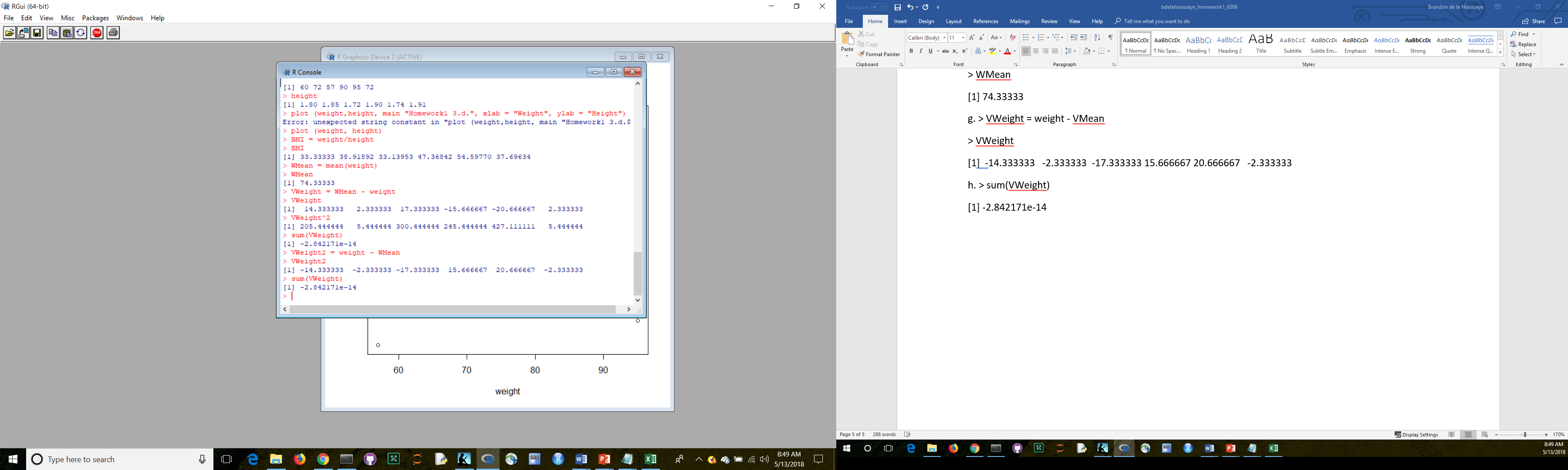
g. > VWeight = weight - VMean

> VWeight

[1] -14.333333 -2.333333 -17.333333 15.666667 20.666667 -2.333333

h. > sum(VWeight)

[1] -2.842171e-14



4.

[Note my response from the pre-read: *computer programming =1 math = 3 statistics = 2 machine learning = 1 domain expertise = 1\* communication and presentation skills = 4 data visualization = 3 \*I would put my domain expertise of something like tax or transfer pricing at a 5. In this course, I would expect my computer programming skills to improve the most. In this program, I would expect a three way tie with statistics, computer programming, and data visualization to improve the most.*

I have updated/edited the rankings based on the live session.]

> Rankings <- c(1,3,2,1,4,4,3)

> Categories <- c("computer programming","math","statistics","machine learning","domain expertise","communication and presentation skills","data visualization")

> rankings

Error: object 'rankings' not found

> rankings = c(1,3,2,1,4,4,3)

> rankings

[1] 1 3 2 1 4 4 3

> categories = c("computer programming","math","statistics","machine learning","domain expertise","communication and presentation skills","data visualization")

> categories

[1] "computer programming" "math" "statistics" "machine learning"

[5] "domain expertise" "communication and presentation skills" "data visualization"

> Brandon = data.frame(rankings,categories)

> brandon

Error: object 'brandon' not found

> Brandon

rankings categories

1 1 computer programming

2 3 math

3 2 statistics

4 1 machine learning

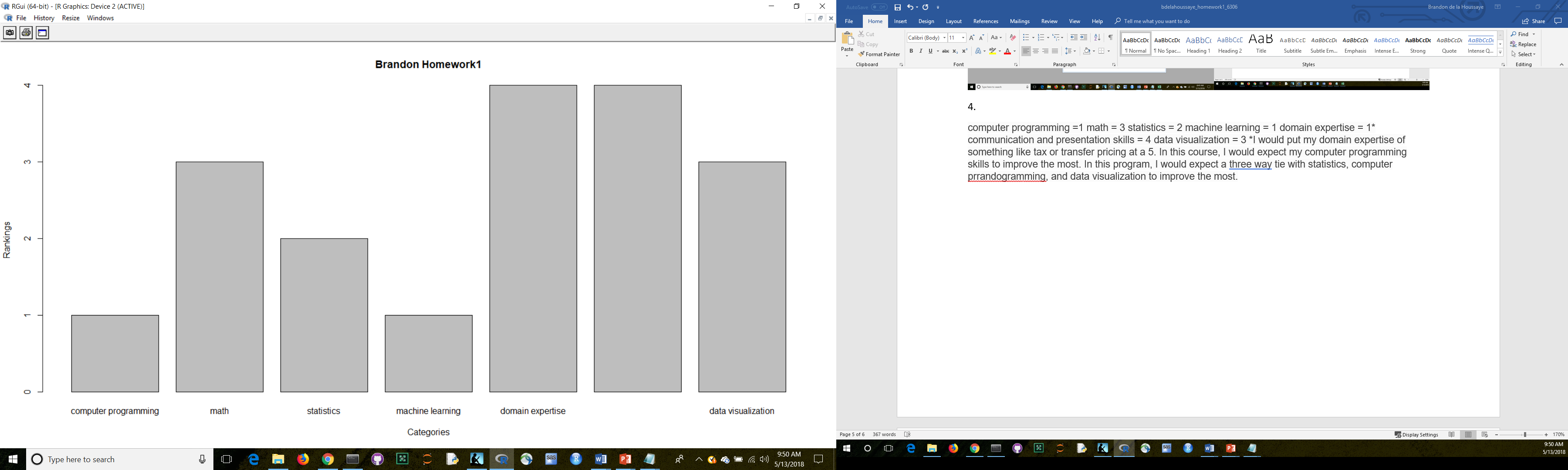
5 4 domain expertise

6 4 communication and presentation skills

7 3 data visualization

> barplot(rankings)

> barplot(rankings,names = categories,xlab = "Categories", ylab = "Rankings", main = "Brandon Homework1")



5. See second file.