IMT 573: Problem Set 6 - Learning from Data

Miloni Desai

Due: Tuesday, November 12, 2019

Collaborators: StackOverFlow, OpenIntroStats

Instructions:

Before beginning this assignment, please ensure you have access to R and RStudio; this can be on your own personal computer or on the IMT 573 R Studio Server.

1. Download the `problemset6.rmd` file from Canvas or save a copy to your local directory on RStudio Server. Open `problemset6.rmd` in RStudio and supply your solutions to the assignment by editing `problemset6.rmd`.
2. Replace the "Insert Your Name Here" text in the `author:` field with your own full name. Any collaborators must be listed on the top of your assignment.
3. Be sure to include well-documented (e.g. commented) code chucks, figures, and clearly written text chunk explanations as necessary. Any figures should be clearly labeled and appropriately referenced within the text. Be sure that each visualization adds value to your written explanation; avoid redundancy – you do no need four different visualizations of the same pattern.
4. Collaboration on problem sets is fun and useful, and we encourage it, but each student must turn in an individual write-up in their own words as well as code/work that is their own. Regardless of whether you work with others, what you turn in must be your own work; this includes code and interpretation of results. The names of all collaborators must be listed on each assignment. Do not copy-and-paste from other students' responses or code.
5. All materials and resources that you use (with the exception of lecture slides) must be appropriately referenced within your assignment.
6. Remember partial credit will be awarded for each question for which a serious attempt at finding an answer has been shown. Students are *strongly* encouraged to attempt each question and to document their reasoning process even if they cannot find the correct answer. If you would like to include R code to show this process, but it does not run withouth errors you can do so with the `eval=FALSE` option.
7. When you have completed the assignment and have **checked** that your code both runs in the Console and knits correctly when you click `Knit PDF`, rename the knitted PDF file to `ps6_ourLastName_YourFirstName.pdf`, and submit the PDF file on Canvas.

Setup

In this problem set you will need, at minimum, the following R packages.

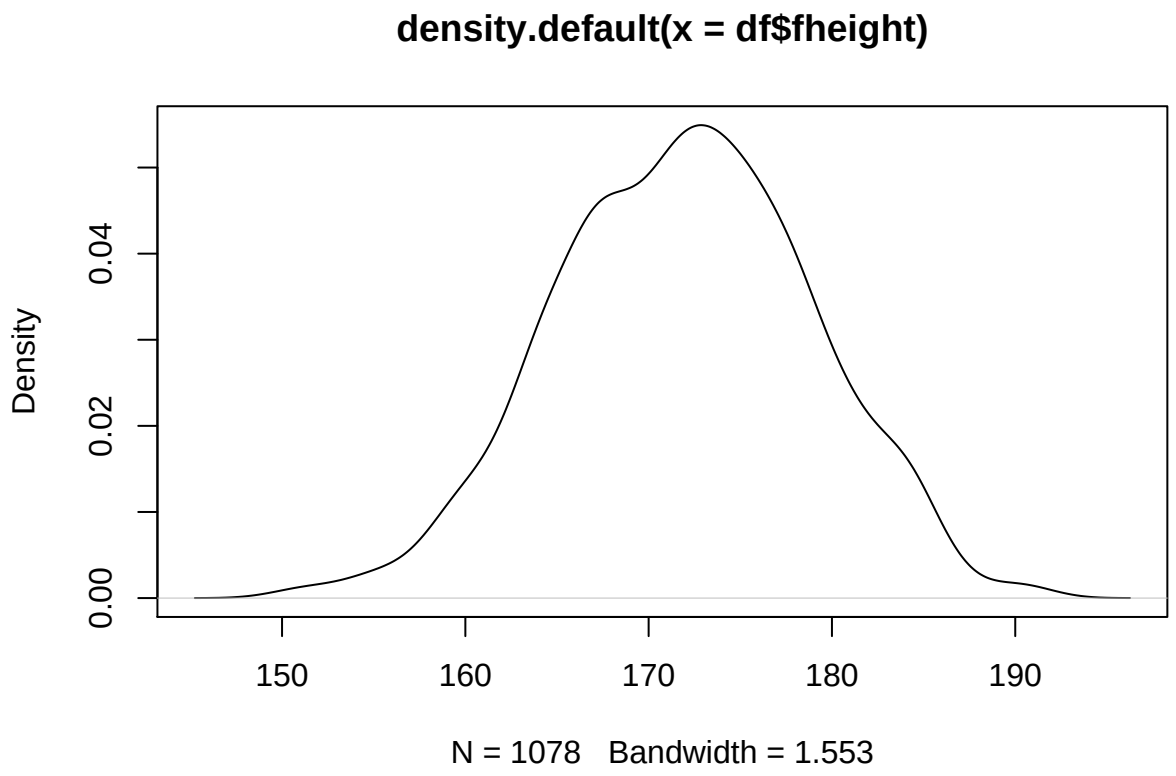
```
# Load standard libraries
library(tidyverse)
library(gridExtra)
```

Problem 1: Are sons taller than fathers?

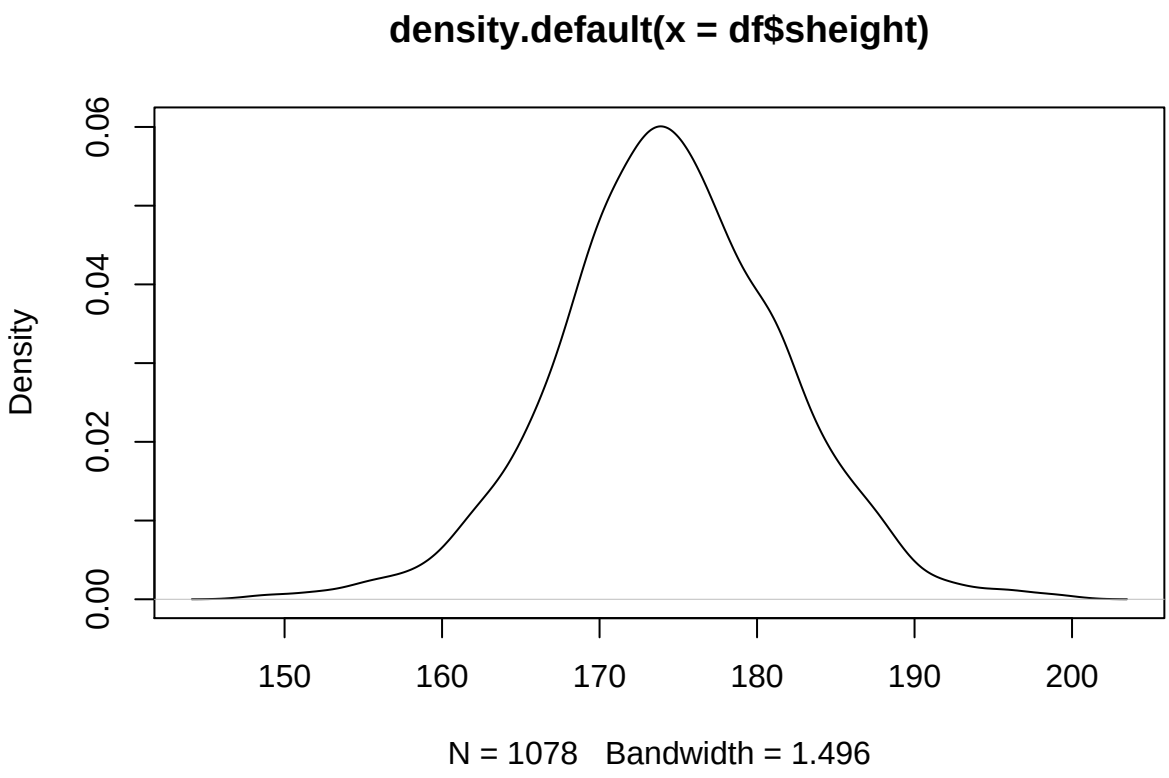
#Here we analyze the dataset of fathers and sons height, used by Pearson and which we saw in the last problem set. It contains two variables, fathers height and sons height. If you take a simple mean, you see that in average sons are taller than fathers. But can this difference just be due to chance? Let's find out.

(a) To begin load the `fatherson.csv` data. Create density plots of both heights on the same figure. Comment the plots. What do they look like? What do they suggest in terms of fathers and sons relative height?

```
df <- read.csv(file = "fatherson.csv.bz2", sep="\t")
#View(df)
d1 <- density(df$fheight) # returns the density data
plot(d1) # plots the results
```




```
d2 <- density(df$sheight) # returns the density data
plot(d2) # plots the results
```




```
x <- data.frame(df$fheight, df$sheight)
#ggplot(data,aes(x=value, fill=variable)) + geom_density(alpha=0.25)
```

#The plots show us that the sons heights tend to generally be higher than the fathers. We can also see that the sons heights are more normally distributed. Relatively the sons avg height and range span more than the fathers.

(b) But is this difference statistically significant? Let's do a t-test. Here I ask you to *compute yourself the t-value*, do not use any pre-existing functions! What do you find? Why did you use/did you not use pooled standard deviations? Explain!

#Understanding the basics statistics of the two populations

```
library("psych")
```

```
##
## Attaching package: 'psych'
```

```
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

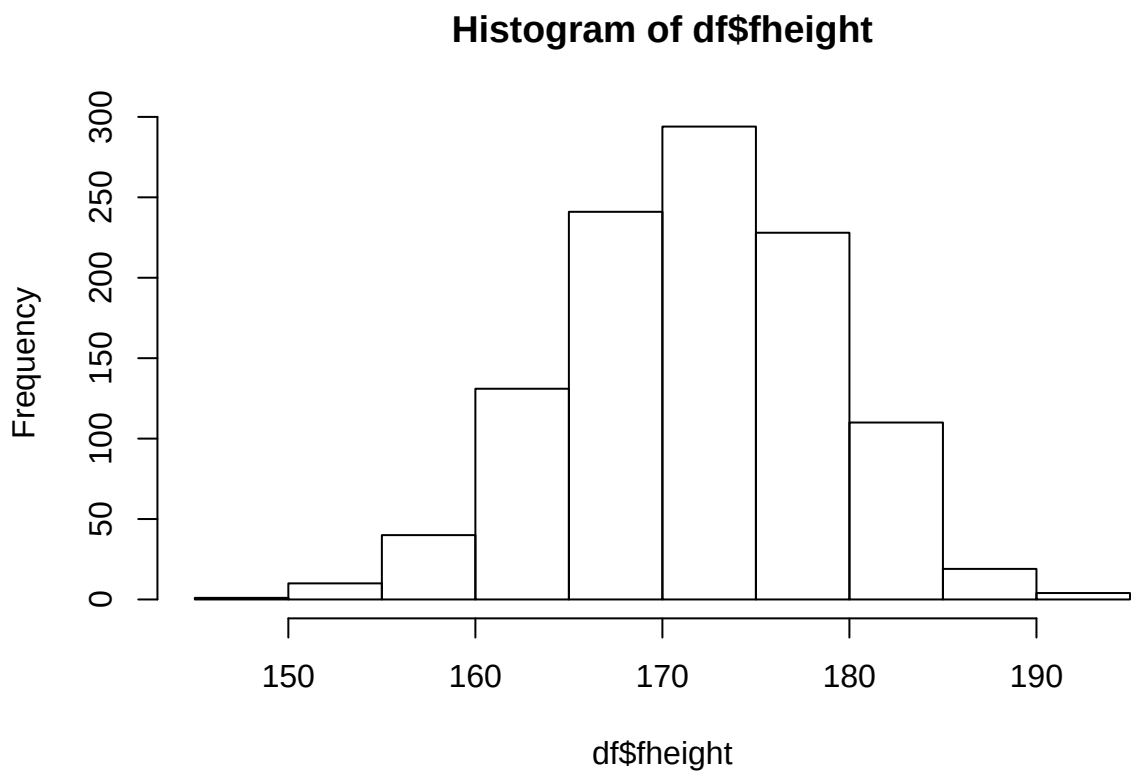
```
describe(df$fheight)
```

```
##    vars    n   mean   sd median trimmed  mad   min   max range  skew
## X1    1 1078 171.93 6.97  172.1  171.97 7.26 149.9 191.6  41.7 -0.09
##    kurtosis   se
## X1    -0.17 0.21
```

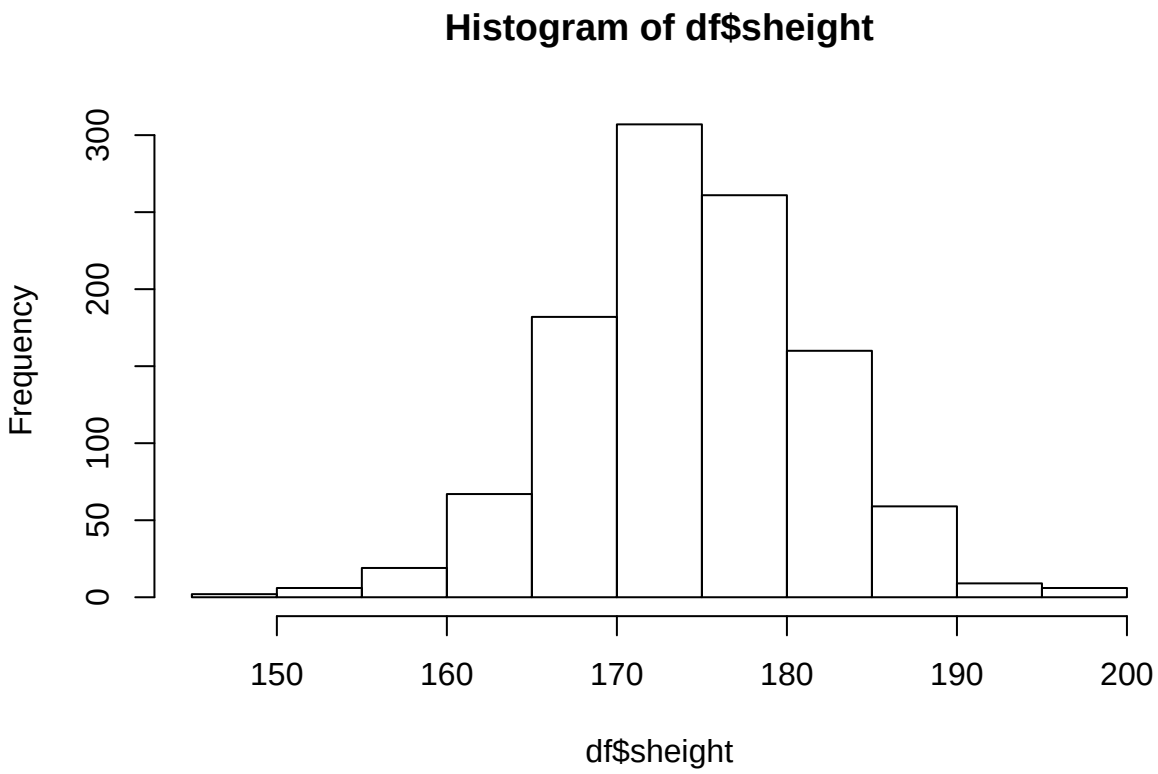
```r
describe(df$sheight)
```

```
##    vars    n   mean   sd median trimmed  mad   min max range  skew
## X1    1 1078 174.46 7.15  174.3  174.47 6.67 148.6 199  50.4 -0.04
##    kurtosis   se
## X1     0.52 0.22
```

```r
hist(df$fheight)
```




```r
hist(df$sheight)
```



#To determine if we will be using pooled sd we find the sd for both and if the condition that the larger sample std devitaion is more than twice the smaller sample std deviation then we perform analysis using unpooled methods

```
sd_f <- sd(df$fheight)
sd_s <- sd(df$sheight)
sd_f
```

```
## [1] 6.972346
```

```
sd_s
```

```
## [1] 7.150713
```

```
factor = sd_s/sd_f
print (factor)
```

```
## [1] 1.025582
```

#As the condition is not met we used pooled methods. #Now we find the t-statistic

```
n1=length(df$fheight)
xbar1=mean(df$fheight)
s1=sd(df$fheight)

n2=length(df$sheight)
xbar2=mean(df$sheight)
s2=sd(df$sheight)

obs=xbar1-xbar2
```

```
obs
```

```
## [1] -2.532004
```

```
diff = xbar2 - xbar1
diff
```

```
## [1] 2.532004
```

```
s.p=sqrt(((n1-1)*s1^2+(n2-1)*s2^2)/(n1+n2-2))
s.p
```

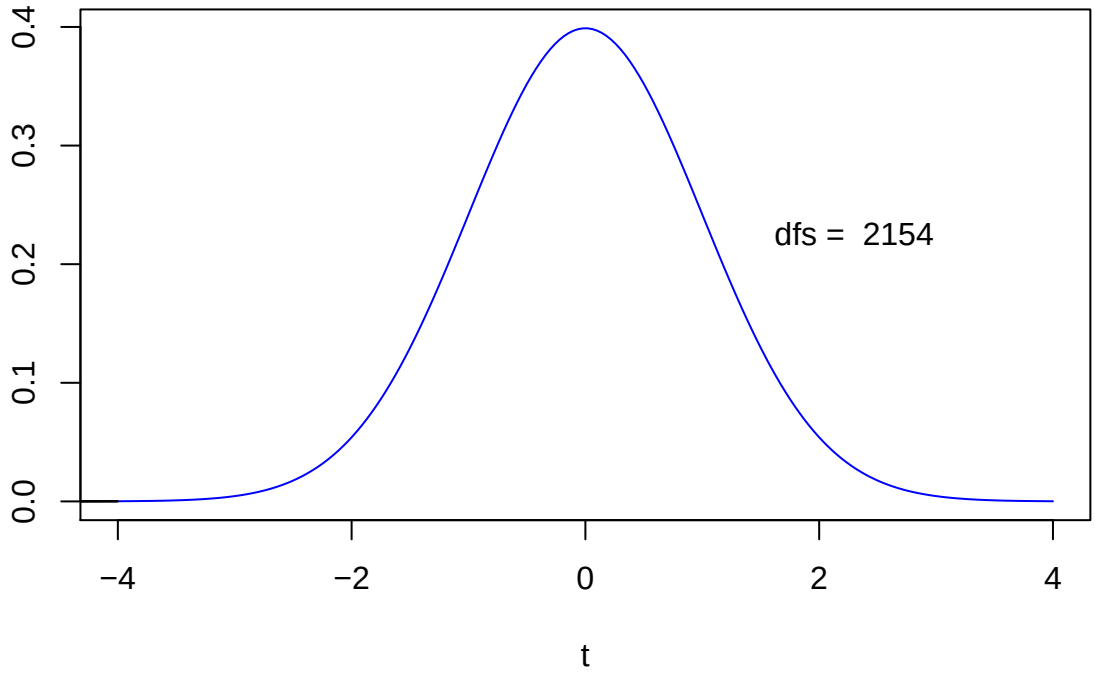
```
## [1] 7.062093
```

```
t.stat=(xbar1-xbar2)/(s.p*sqrt(1/n1+1/n2))
t.stat
```

```
## [1] -8.32387
```

#here we get a t-stat of -8.324 and we have used pooled std deviation to compute it as the condition above mentioned is met by our data.

```
dfs=n1+n2-2
tt=seq(-4,4,length=200)
yy=dt(tt,dfs)
plot(tt,yy,type="l",col="blue",xlab="t",ylab="")
xx=seq(-4,t.stat,length=100)
yy=dt(xx,dfs)
polygon(c(-4,xx,t.stat),c(0,yy,0),col="gray")
arrows(t.stat,.25,t.stat,.15,length=0.1)
text(t.stat,.25,"-1.486",pos=3)
text(2.3,.2,paste("dfs = ",dfs),pos=3)
```




```
p=pt(t.stat,dfs)
p
```

```
## [1] 7.461564e-17
```

#Here we use the t-stat value to plot the sample t values to observe th distribution and then by using the area under the curve left of the t-stat value we find the p value. As we can see it is extremely small. This rejects our null hypothesis and supports our alternate hypothesis that the avg height of fathers is less than that of their sons.

Hint: read OIS 7.3

(c) Look up the t-distribution table. (Or compute the relevant quantiles). What is the likelihood that such a t value happens just by random chance? Hint: be sure to consider the *degrees of freedom* in current case carefully!

```
quantile (df$fheight)
```

```
##    0%   25%   50%   75%  100%
## 149.9 167.1 172.1 176.8 191.6
```

```
quantile (df$sheight)
```

```
##    0%   25%   50%   75%  100%
## 148.6 170.0 174.3 179.0 199.0
```

#The t-distribution with more degrees of freedom has thinner tails. This occurs because the t-distribution will reflect the uncertainity added with analyzing the sample(depends on its size). So we can say that a large sample will have a higher probability of having the sample statisic closer to the null hypothesis even

when the null hypothesis is true.Thus we can say as the number of degrees of freedom are large, we get such a small p-value.

(d) Based on your above analysis, state clearly your conclusion to the question - are sons taller than fathers?

#From our analysis we can conclude that all the statistical analysis points to the fact that the sons are taler than the fathers.

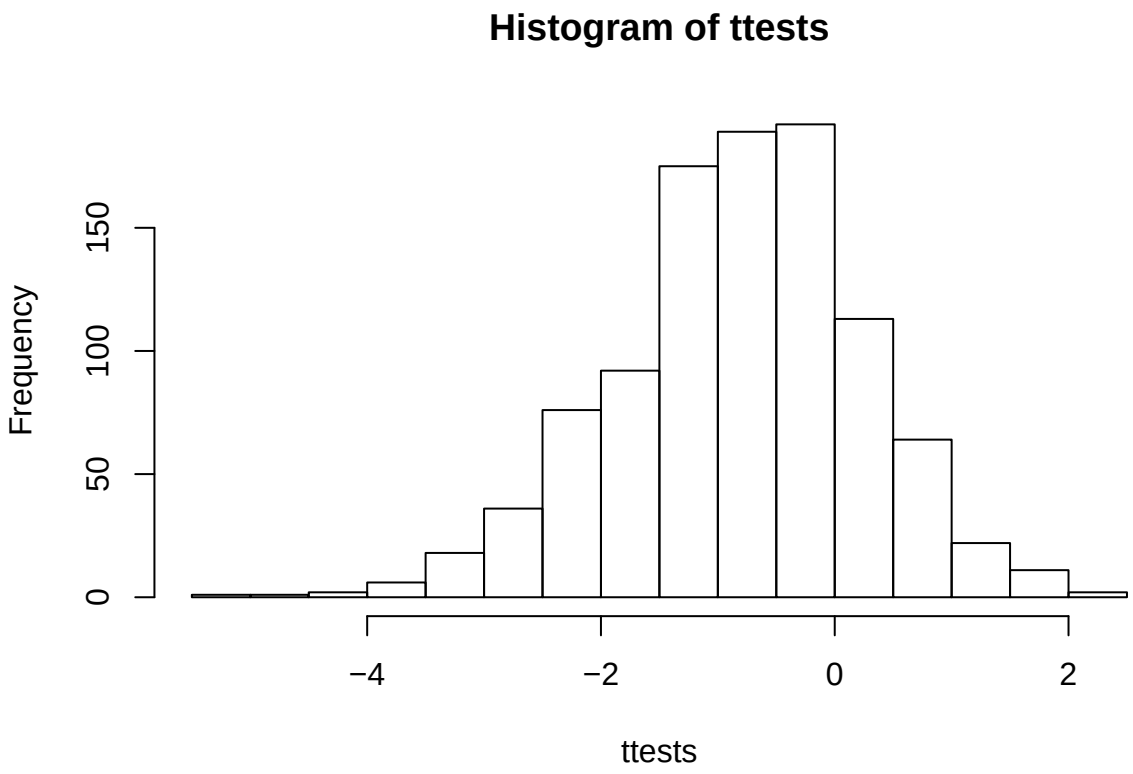
Problem 2: Fathers and Sons Monte Carlo approach

#Next, let's re-visit the fathers and sons height, but this time by doing Monte Carlo analysis on a computer. You will proceed as follows: create two samples of random normals, similar to the data above, using the mean and standard deviation over both fathers and sons. Call one of these samples fathers and the other sons. What is the difference in their means? And now you repeat this exercise many times and see if you can get as big a difference as what you saw above in the data.

(a) First, compute the overall mean and standard deviation of combined fathers' and sons' heights. Now create two sets of normal random variables, both with the same mean and standard deviation that you just computed above. Call one of these `fathers` and the other `sons`. What is the father-son mean difference? Compare the result with that you found in the previous problem.

```
ttestgenerator <- function(n) {
  fathers <- sample(df$fheight, n, replace = TRUE)
  sons <- sample(df$sheight,n, replace = TRUE)
  tstat <- (mean(fathers)-mean(sons)) /
      sqrt( var(fathers)/n + var(sons)/n )
  return(tstat)
  }
ttests <- replicate(1000, ttestgenerator(10))

hist(ttests)
```



#The difference in mean is higher than what we got last time.The mean difference is almost 4 times.

(b) Now repeat the previous question a large number of times R (1000 or more). Each time store the difference, so you end up with R different values for the difference. What is the mean of the difference values? Explain what do you get. What is it standard deviation? Compare it to that you computed in the previous problem for the difference in data (when doing t-test). What is the largest difference (in absolute value)?

```
mean = 173.2
sd = 7.06
n = 10000
ttestgenerator <- function(n, mean= 173.2, sd= 7.06) {
  fathers <- rnorm(n,mean,sd)
  sons <- rnorm(n,mean,sd)
  tstat <- (mean(fathers)-mean(sons)) /
      sqrt( var(fathers)/n + var(sons)/n )
  return(tstat)
  }
ttest <- replicate(1000, ttestgenerator(10))
summary(ttest)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -4.06092 -0.72971  0.01555  0.01091  0.73198  4.20705
```

#Here what we are trying to essentially do is trying to see if generating n number of random samples of the data gives us similar values in mean and sd to see if our previous t-test that told us that sons are generally taller than fathers keeps up. This is a good way to estimate and predict. We see that as the number of

samples and degrees of freedom increase, the mean tends to flucate and in an increasing pattern. The mean is almost the same as that found in the previous section when the samples are close to the actual observations in our data frame.

(c) Find the 95% quantile of (the absolute value) your difference. Compare this number to the actual father-son difference you found in the data.

```
quantile (ttest,.95)
```

```
##      95%
## 1.71089
```

Hint: use the R function `quantile` for this.

Extra Credit: Parallel Computing

#Here your task is to repeat the previous exercise (only MC part of it) using parallel processing. Conduct the MC analysis using a parallel loop. Hint: check out the packages *parallel* and *foreach*.

#Time your code. Create a table that shows how the simulation time depends on the number of employed CPU cores. Can you get a noticeable speed improvement by running the simulation code in parallel?