IMT 573: Problem Set 6 - Learning from Data

Divya Gaurav Tripathi

Due: Tuesday, November 12, 2019

Collaborators:

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.

```r
# Load standard libraries
library(tidyverse)
library(gridExtra)
library(ggplot2)
```

Problem 1: Are sons taller than fathers?

Here we analyze the dataset of father's 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. HWhat do they look like? What do they suggest in terms of fathers' and sons' relative height?

```
heightdata <- read.csv("fatherson.csv",stringsAsFactors=FALSE)
str(heightdata)
```

```
## 'data.frame':    1078 obs. of  1 variable:
##  $ fheight.sheight: chr  "165.2\t151.8" "160.7\t160.6" "165\t160.9" "167\t159.5" ...
```

```
seperate_heights <- separate(heightdata, fheight.sheight,
                c("fatherheight", "sonheight"), sep = "\\b\\s\\b",
                remove = TRUE)

#We seperated father's heights and son's heights on white space.

str(seperate_heights)
```

```
## 'data.frame':    1078 obs. of  2 variables:
##  $ fatherheight: chr  "165.2" "160.7" "165" "167" ...
##  $ sonheight   : chr  "151.8" "160.6" "160.9" "159.5" ...
```

```
heights_tidydata <- seperate_heights %>% mutate_if(is.character, as.numeric)
#We converted the heights from char to num format

str(heights_tidydata)
```

```
## 'data.frame':    1078 obs. of  2 variables:
##  $ fatherheight: num  165 161 165 167 155 ...
##  $ sonheight   : num  152 161 161 160 163 ...
```

```
view(heights_tidydata)
```

We might have to put all heights in one column, lets do that also using gather function.

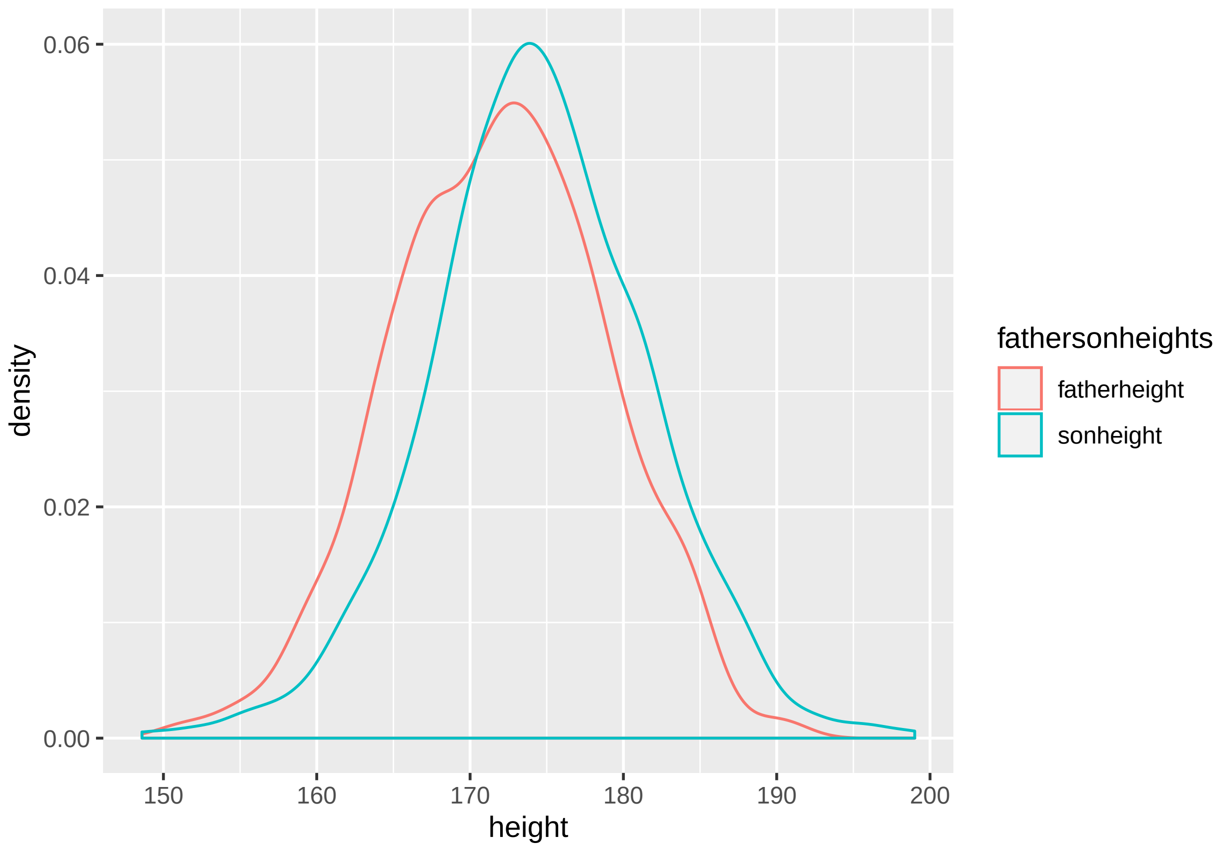
```
allheights <-  gather(heights_tidydata, "fathersonheights","height",1:2)

#We used gather function, one column would be for father/son, the other column would tell height.
```

allheights is our data frame in cleaned format. Let us try to plot density plot for father's and son's heights.

```
fathersonheight_plot <- ggplot(allheights, aes(x= height, color = fathersonheights)) +
  geom_density()

fathersonheight_plot
```



The density plot tells that the variation of father's heights and son's heights are not very different. The son's heights are a bit higher than father's heights.

(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!

Hint: read OIS 7.3 We are checking the difference between two population means(son's heights and father's heights) are due to chance or statistically significant.

Our null hypothesis is that there is no difference in mean son's heights and mean father's heights. Our alternate hpothesis is that sons are taller than fathers, so its a one sided t-test.

H_null = son's and father's heights are not different(Height_son = Height_father) H_alternate = sons are taller than fathers (Height_son > Height_father) We would take differeence of two means t-test.

We are doing a one sided test.

```r
summary(heights_tidydata)
```

```
##   fatherheight     sonheight    
##  Min.   :149.9   Min.   :148.6  
##  1st Qu.:167.1   1st Qu.:170.0  
##  Median :172.1   Median :174.3  
##  Mean   :171.9   Mean   :174.5  
##  3rd Qu.:176.8   3rd Qu.:179.0  
##  Max.   :191.6   Max.   :199.0
```

```r
fatherheights_sd <- sd(heights_tidydata$fatherheight)
#standard deviation for fathers heights
fatherheights_sd
```

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

```r
sonheights_sd <- sd(heights_tidydata$sonheight)
#standard deviation for the sons heights
sonheights_sd
```

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

For fathers heights: mean = 171.9, sd_father = 6.972, median = 172.1 For son's heights: mean = 174.5, sd_son = 7.15, median = 174.3 difference of means = means_diff = mean son's heights - mean father's heights = 174.5-171.9 = 2.6

Here our population parameters are well documented and their standard deviations are almost similar. So we can use pooled standard deviation.

Number of observations of father's heights = n_father = 1078 Number of observations of son's heights = n_son = 1078

pooled standard deviation = sd_pooled =((sd_father^2 *(n_father-1)+ sd_son^2*(n_son-1))/(n_father + n_son - 2))^0.5 = ((6.972^2 *(1078-1) + 7.15^2*(1078-1))/(1078 + 1078 - 2))$^{0.5\ =(49.86)}$0.5 =7.0611

degrees of freedom = df = n_father + n_son -2 = 1078 + 1078 - 2 = 2154

standard error = se = (sd_pooled^2/n_father + sd_pooled$^{2/n_son)}$0.5 = (7.0611^2/1078 + 7.0611$^{2/1078)}$0.5 = 0.3041

For father's heights: n_father -1 = 1078 For son's heights: n_sons -1 = Our null hypothesis is thta there is no difference between father's and son's heights. T score = T = (means_diff-0)/se = (2.6-0)/0.3041 = 8.552 We had also calculated degree of freedom = df = 2154

If you check the t-table for degree of freedom 2154 and p value of 0.5(on sided), we find that the T_critical is between 1.65 and 1.64.

Our observed T score is greater than T critical, so we reject the null hypothesis.

(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!

Tscore = 8.552 df = 2154

```r
p_value <- pt(8.552,2154,lower.tail = FALSE)

p_value
```

```
## [1] 1.128582e-17
```

In the previous part we found that T observed was greater than T critical, so we reject the null hypothesis. We found the p value to find the liklihood that such t value happens by chance.

We see that the p value is 1.128582e-17 which is less than alpha(0.05). The probablity that this height difference is observed by chance is 1.128582e-17, which is extremely less than 0.05. So at the significance level of 0.05, we would reject the null hypothesis.

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

We did a t test which tells that the difference in height is not due to chance. We would reject the null hypothesis that father's and son's heigts are equal. It means sons are taller than fathers.

Problem 2: Fathers and Sons - the 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.

The data frame allheights has father's and son's heights in one column.

```
totalmean = mean(allheights$height)
totalmean
```

```
## [1] 173.1912
```

```
#This gives the mean when we consider all the heights irrespective of whether it is father's or son's

totalsd = sd(allheights$height)
totalsd
```

```
## [1] 7.173111
```

```
#This gives the standard deviation when we consider all the heights irrespective of whether it is fathe
```

We have combined mean = totalmean = 173.19, and standard deviation = totalsd = 7.17 Let us generate random normal variables.

```
fathers <- rnorm(1078, totalmean, totalsd)
#We created 1078 normal random variables and termed it father

sons <- rnorm(1078, totalmean, totalsd)
#We created 1078 normal random variables and termed it son
#diff <- sons-fathers

mean_diff <- mean(sons) - mean(fathers)
#We calculated the mean height difference

mean_diff
```

```
## [1] -0.436322
```

Here we got the mean height difference between father and son = 0.58 In Problem 1(when we did t test), this difference was 2.6

Please note that as we are generating random normal variables, each time you run the code we would get different mean values corresponding to the generated random variables.

(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)?

We would create an empty vector(mean_vec). We would run the above steps to find mean difference from fathers and sons vectors 1200 times, and on each iteration we would keep on appending this mean difference to our mean_vec.

```
mean_vec <- vector()
#We initialised an empty vector

for (i in 1:1200) {
  #We take a for loop that would take the mean difference between fathers and sons 1200 times

fathers <- rnorm(1078, totalmean, totalsd)
#We created 1078 normal random variables and termed it father

sons <- rnorm(1078, totalmean, totalsd)
#We created 1078 normal random variables and termed it son
#diff <- sons-fathers
mean_diff <- mean(sons) - mean(fathers)
#We calculated the mean height difference

mean_vec[i] <- mean_diff
#For every ith iteration, we append the mean_diff to mean_vec
}

str(mean_vec)
```

```
##  num [1:1200] 0.265 -0.395 -0.385 -0.291 0.112 ...
```

```
max_mean_vec <-max(abs(mean_vec))
#This gives the maximun value of mean_vec
```

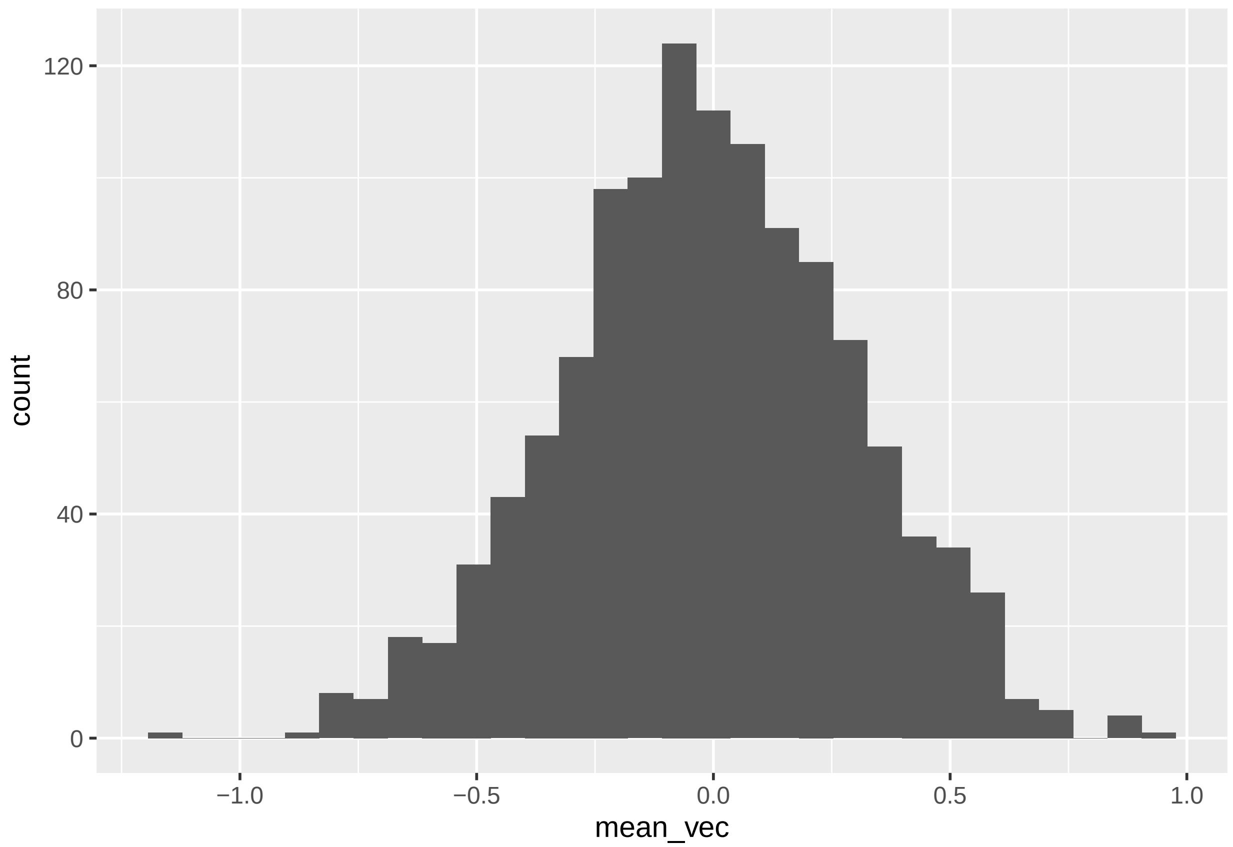
We got our mean_vec vector. Let us find its mean, standard deviation and make a density plot.

```
mean_from_meanvector <- mean(mean_vec)
sd_from_meanvector <- sd(mean_vec)
```

```
#mean_vecdensityplot <- ggplot(mean_vec,aes(x =mean_vec )) + geom_histogram()
#hist(mean_vec)

ggplot() + geom_histogram(aes(mean_vec))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Please note that as we are generating random normal variables, each time you run the code we would get different mean values corresponding to these random generated variables.

The largest difference(in absolute value) of the mean difference in mean_vec = max_mean_vec(we found above) = 1.02

The mean of mean_vec = mean_from_meanvector(we calculated above) = 0.006138692 The standard deviation of mean_vec = sd_from_meanvector(we calculated above) = 0.305387

These are very small values. We have generated two sets of normal variables(fathers and sons) with the same mean and standard deviation. Then we took their differences. So the mean and standard deviation are very small.

In problem 1 when we did the t test, the mean difference between son's and father's heights was 2.6, which was larger than what we found here. It was because in problem 1 in the begining of this assignment,the mean and standard deviation of father's and son's heights(when considered as seperate sets) are not exactly equal.

We made a histogram of our mean height difference vector which resembles a normal curve. The Central Limit Theorem states that the sampling distribution of the sample means approaches a normal distribution as the sample size gets larger — no matter what the shape of the population distribution. The distribution of our sample means also tends to be a normal distribution when we plot it for 1200 values. It proves the Central Limit Theorem.

(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.

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

```
quantile <- quantile(abs(mean_from_meanvector),0.95)
quantile
```

```
##        95%
## 0.01118772
```

Please note that as we are generating random normal variables, each time you run the code we would get different mean values corresponding to these random generated variables.

The 95% quantile is 0.01. It means 95% of the mean height difference is below this value of 0.01.The actual mean difference of father and son was 2.6 which was quite larger than this. It is because in problem 2, We have generated two sets of normal variables(fathers and sons) with the same mean and standard deviation. Then we took the differences of their means. So we got very small values.

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?