# HW #3 Stat 346 Fall 2020

Prof. De Veaux

Due October 2 by 11:45 AM

1. **Simulations to see unbiased variances** We have proved that $E(\sum(y_i - \bar{y})^2) = (n-1)\sigma^2$ and not $n\sigma^2$, but let's just prove it to ourselves via a simulation. Use set.seed(100) Generate $40,000$ $y$ values from a $N(5,3)$ – mean 5, sd 3. Now put them into a matrix 10,000 rows by 4, so that we can consider them 10,000 random samples of size 4. Now, we're "God", and we know $\mu = 5$, so find the 10,000 estimates of $\sigma^2$ knowing that. That is calculate $\sum(y_i - 5)^2$ for each row.

a. What should the average of those be? Is it?
b. Now, instead, for each row, calculate $\sum(y_i - \bar{y})^2$. How big should it be, on average? Is it?

2. **Same thing for regression** So, I claimed that now for regression we should divide $\sum e_i^2$ by $n-2$ not $n-1$. Let's show that's right (first by simulation, then later by math). Use set.seed(100) again. Let x be 100 values again from N(5,3). Do a for loop, where each time you let $y = 4 + 3x + \varepsilon$ , where $\varepsilon \sim N(0,1)$. (But don't change the $x$) For each regression calculate $\sum e_i^2$. Remember there are $n = 100$ values for each regression. We want to estimate $\sigma^2$ which we know is 1. How big is $\sum e_i^2$ on average. So, what should I divide $\sum e_i^2$ by to get an unbiased estimate of $\sigma^2$?

3. **To warm up, let's prove some things about the slope estimate** Remember that we can write $b_1 = \sum k_i y_i$ where $k_i = \frac{(x_i - \bar{x})}{S_{xx}}$ and $S_{xx} = \sum(x_i - \bar{x})^2$ is the same for all $x_i$ since it sums over all of them. Because $b_1$ is just a linear combination of the $y_i$ (the $k_i$ are all constants, since we assume the $x_i$ are given), then

$$E(b_1) = \sum k_i E(y_i)$$

Remember that this is regression so we assume $y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$ where $\varepsilon \sim N(0, \sigma^2)$ Use that fact to show that $b_1$ is unbiased for $\beta_1$.

4. **Variance of the slope** Last week you bootstrapped the slope to see how much it varies. Redo that just to remember what the sd of the slope is. Use set.seed(100) and we can do the simulation in one line, using mosaic:

```
set.seed(100)
 res=do(1000)*coef(lm(Weight~Height,data=resample(bodyfat)))[2]
```

I get 0.598 for the sd of the slopes.

a. What does summary() tell us the sd of the slope is for the regression on the original data set?

b. Ok, now let's prove what it should be. Remember again that $b_1$ is a linear combination of the $y_i$ which are independent. If $b_1 = \sum k_i y_i$, what is $Var(b_1)$ in terms of $k_i$ and the variance of $y_i$ which is $\sigma^2$? Hint: $S_{xx} = \sum(x_i - \bar{x})^2 = \sum(x_i - \bar{x})x_i$ (Same trick we used with $y_i$)

c. Show that the formula works for the *Weight* on *Height* regression.

5. **One more – the hard one** Finally, let's prove that $E(SSE) = (n-2)\sigma^2$.

a. Again, start by writing $\hat{y_i} = \bar{y} + b_1(x_i - \bar{x})$. so first show that $SSE = \sum(y_i - \bar{y})^2 - b_1^2 S_{xx}$

We'll call $\sum(y_i - \bar{y})^2$ $SSTO$. Hint: First expand to show

$$SSE = SSTO - 2b_1 S_{xy} + b_1^2 S_{xx} = SSTO - b_1^2 S_{xx}$$

So to compute $E(SSE)$ we need both $E(SSTO)$ and $E(b_1^2 S_{xx})$.

b. Let's start with the second term. $S_{xx}$ is a constant, but $b_1^2$ is not. Remember what $Var(b_1)$ and $E(b_1)$ are (from the problems above!) and remember that we showed that for any random variable, $E(X^2) = Var(X) + [E(X)]^2$. Use that to show: t

$$E(b_1^2 S_{xx}) = \sigma^2 + \beta_1^2 S_{xx}$$

c. Now the trickier one: $E(SSTO) = E[\sum (y_i - \bar{y})^2]$. It's tempting to think that this should be $(n-1)\sigma^2$, but .... that's only true if the $y_i$ have the same mean. Here they don't. $E(y_i) = \beta_0 + \beta_1 x_i$ which are *not* the same. That's going to add extra variation.

First, note that we can write $SSTO = \sum y_i^2 - n\bar{y}^2$. So, $E(SSTO) = \sum E(y_i^2) - nE(\bar{y}^2)$

and remember again that $E(X^2) = Var(X) + [E(X)]^2$. So apply that to each part.

Use the model: write $y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$. This tells us that $E(y_i) = \beta_0 + \beta_1 x_i$ and $Var(y_i) = \sigma^2$. Remember also that $Var(\bar{y}) = \sigma^2/n$. The only weird part is that $E(\bar{y}) = \beta_0 + \beta_1 \bar{x}$. Is that weird?

You should get that $E(SSTO) = (n-1)\sigma^2 + \beta_1^2 S_{xx}$

d. Put the parts together and prove the result.

6. **Exploring the t distribution** So what's really up with the $t$ distribution? Let's look at Gosset's (https://en.wikipedia.org/wiki/William_Sealy_Gosset) original data to see what he saw. He collected 3000 heights and left middle finger lengths of criminals from an article in *Biometrika* (which is still publishing today).

```
require(stats)
dim(crimtab)
```

```
## [1] 42 22
```

```
utils::str(crimtab)
```

```
##  'table' int [1:42, 1:22] 0 0 0 0 0 0 1 0 0 0 ...
##  - attr(*, "dimnames")=List of 2
##   ..$ : chr [1:42] "9.4" "9.5" "9.6" "9.7" ...
##   ..$ : chr [1:22] "142.24" "144.78" "147.32" "149.86" ...
```

```
## for nicer printing:
local({cT <- crimtab
       colnames(cT) <- substring(colnames(cT), 2, 3)
       print(cT, zero.print = " ")
})
```

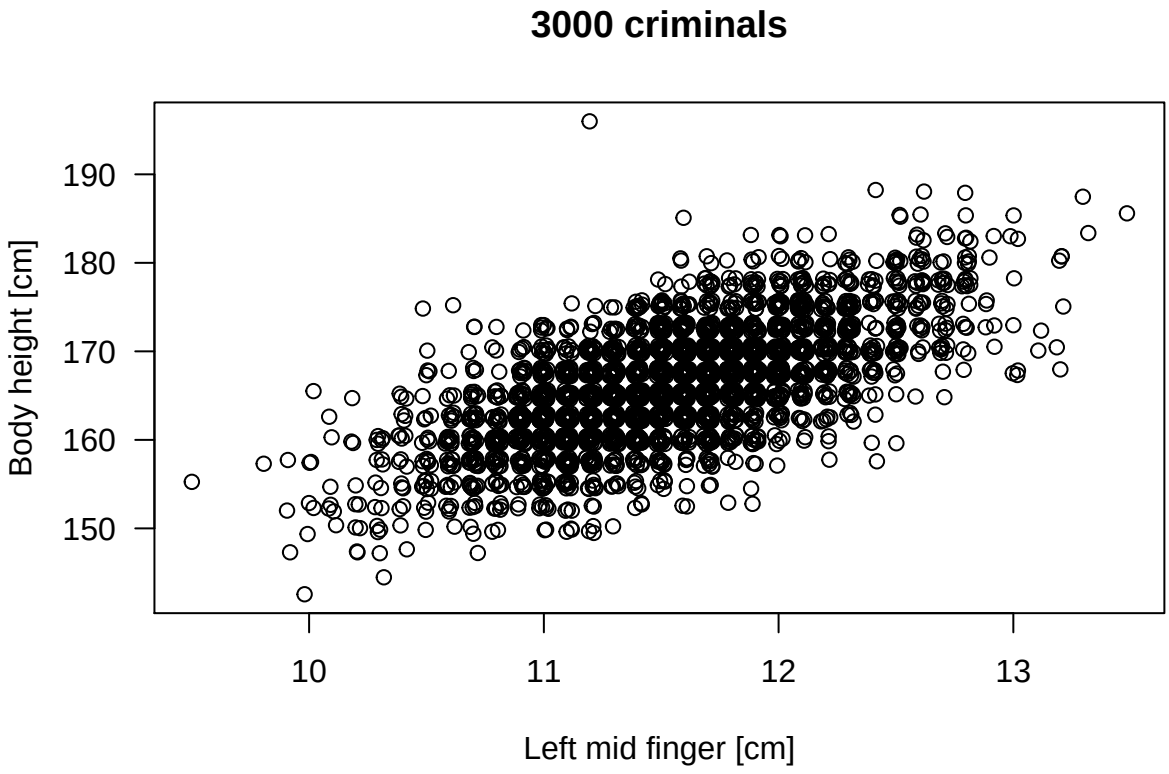
```
##      42 44 47 49 52 54 57 60 62 65 67 70 72 75 77 80 82 85 87 90 93 95
## 9.4
## 9.5                  1
## 9.6
## 9.7
## 9.8                     1
## 9.9         1     1     1
## 10    1        1  2     2        1
## 10.1           1  3  1     1  1
## 10.2        2  2  2  1     2     1
## 10.3     1  1  3  2  2  3  5
## 10.4        1  1  2  3  3  4  3  3
## 10.5           1  3  7  6  4  3  1  3  1     1
## 10.6           1  4  5  9 14  6  3  1        1
```

```
## 10.7        1  2  4  9 14 16 15  7  3  1  2                                    
## 10.8           2  5  6 14 27 10  7  1  2  1                                    
## 10.9              2  6 14 24 27 14 10  4  1                                    
## 11             2  6 12 15 31 37 27 17 10  6                                    
## 11.1           3  3 12 22 26 24 26 24  7  4  1                                 
## 11.2           3  2  7 21 30 38 29 27 20  4  1                        1        
## 11.3           1     5 10 24 26 39 26 24  7  2                                
## 11.4              3  4  9 29 56 58 26 22 10 11                                 
## 11.5                 5 11 17 33 57 38 34 25 11  2                             
## 11.6              2  1  4 13 37 39 48 38 27 12  2  2     1                    
## 11.7                 2  9 17 30 37 48 45 24  9  9  2                          
## 11.8              1     2 11 15 35 41 34 29 10  5  1                          
## 11.9              1  1  2 12 10 27 32 35 19 10  9  3  1                       
## 12                      1  4  8 19 42 39 22 16  8  2  2                       
## 12.1                       2  4 13 22 28 15 27 10  4  1                       
## 12.2                    1  2  5  6 23 17 16 11  8  1  1                       
## 12.3                          4  8 10 13 20 23  6  5                          
## 12.4                    1  1  1  2  7 12  4  7  7  1        1                  
## 12.5                       1     1  3 12 11  8  6  8     2                    
## 12.6                             1     3  5  7  8  6  3  1  1                 
## 12.7                             1  1  7  5  5  8  2  2                       
## 12.8                                1  2  3  1  8  5  3  1  1                 
## 12.9                                   1  2  2     1  1                       
## 13                                  3     1     1     2  1                    
## 13.1                                   1  1                                   
## 13.2                                1  1     1     3                         
## 13.3                                                1     1                  
## 13.4                                                                         
## 13.5                                                   1                     
```

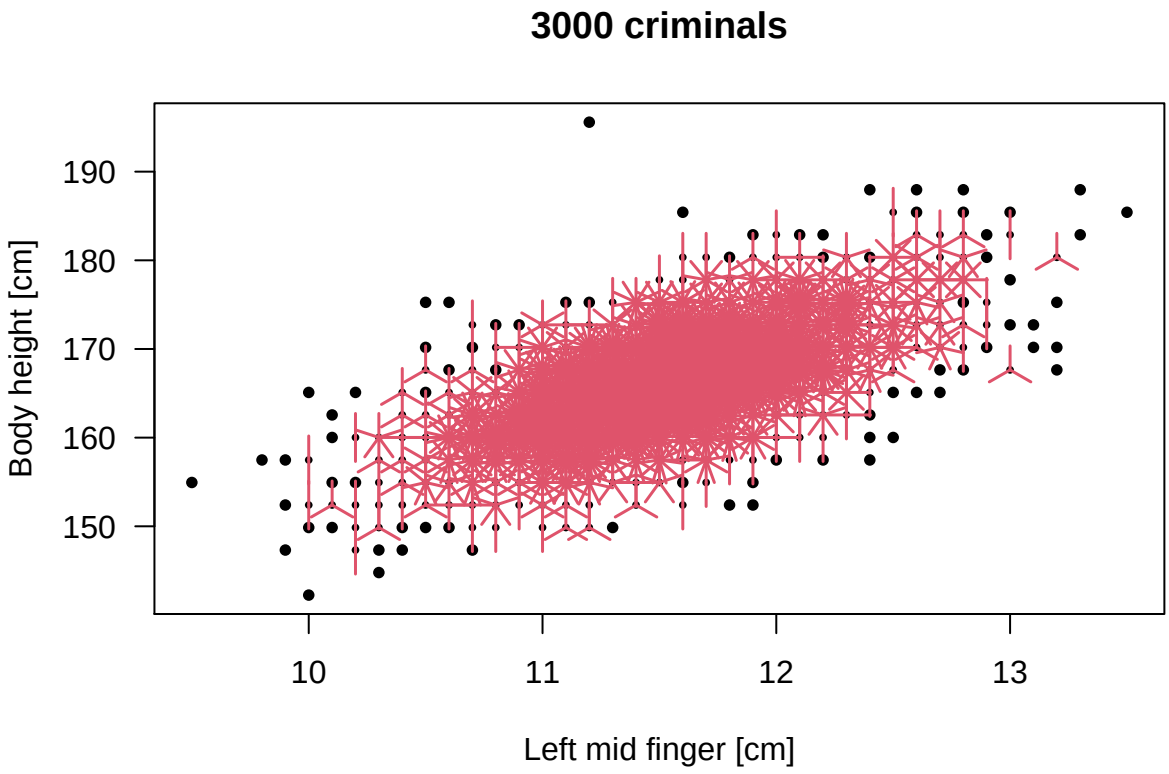
Hmm.. Let's make this look nicer:

```
crimtab.dft <- as.data.frame(crimtab)
expand.dft <- function(x, na.strings = "NA", as.is = FALSE, dec = ".") {
DF <- sapply(1:nrow(x), function(i) x[rep(i, each = x$Freq[i]),], simplify = FALSE)
DF <- subset(do.call("rbind", DF), select = -Freq)
for (i in 1:ncol(DF)) {
DF[[i]] <- type.convert(as.character(DF[[i]]), na.strings = na.strings,
as.is = as.is, dec = dec)
}
DF
}

crimtab.raw <- expand.dft(crimtab.dft)
x <- crimtab.raw[, 1]
y <- crimtab.raw[, 2]
plot(jitter(x), jitter(y), las = 1, main = "3000 criminals", ylab = "Body height [cm]",
xlab = "Left mid finger [cm]")
```




```
sunflowerplot(x, y, las = 1, main = "3000 criminals", ylab = "Body height [cm]", xlab = "Left mid finge
```




Ok. Let's repeat essentially what Gosset did. (We'll look just at the heights.)

```
heights=crimtab.raw[,2]
```

First, find the mean and sd of all the heights. We'll use these as the "population" mean and sd.

```
pop.mean=mean(heights)
pop.sd=sd(heights)
```

Now, we'll turn these 3000 observations into 750 samples of size 4. (Use set.seed(200) so we all get the same samples)

```
set.seed(200)
heights=matrix(mosaic::shuffle(crimtab.raw[,2]),ncol=4) #shuffle the heights
heights=matrix(heights,ncol=4) # turn the 3000 into 750 samples -- one per row
```

a. Find the mean and sd of each row (the function apply is the way to go). Save these as *means* and *sds*. (There should be 750 of each)

b. For each sample we know from the Central Limit Theorem that (approximately)

$$\frac{\bar{y}-\mu}{\sigma/\sqrt{n}} = z$$

What's $n$ here? Now compute these 750 z-scores using $\mu = 166.3$ and $\sigma = 6.5$. Look at the histogram of these and a normal probability plot (qqnorm). Do they look normal? What are the $2.5^{th}$ and $97.5^{th}$ percentiles? What "should" they be?

c. Ok. Here's what Gosset did. For years, people had been just plugging in sd of each sample into the z formula above and assuming that would be ok. So, instead of using $\sigma = 6.5$ in each of those calculations, put the sample sd from each sample of size 4. Call those tscores. Look at the histogram, normal plot and quantiles of this distribution. Does using the sample sds change anything?

d. Write a paragraph summarizing what you (and Gosset) learned from this exercise.

7. **Roller Coasters!** Download the data set Coasters_2015 from DASL (https://dasl.datadescription.com/). These are 241 roller coasters from around the world with various measurements. The variable *Drop* measures how far the coaster drops while *Height* is the measurement to the highest point of the coaster. It seems that there should be a strong relationship.

a. Find the regression of *Drop* on *Height*

b. Check the degrees of freedom for the residual sum of squares. Is it n-2? Explain.

c. Calculate $SSTO$ and $SSE$ for this regression. Using the fact that $R^2 = 1 - \frac{SSE}{SSTO}$, show that you get the $R^2$ value from the summary().

d. Does the regression appear to be appropriate? Explain briefly.

e. Look at the residual plot vs. the predicted values and comment. Is there an outlier? Who is it?

f. Use the data base rcdb.com to verify the information. If it's not correct, rerun the regression and comment on the difference.