Lab: Week 8

AUTHOR
Moksha Sharma

Last 2 Problem Sets: To divide or not to divide?

I have gotten a lot of questions about when we need to divide by N in calculating the standard error.

In answering this question, we need to consider what we are trying to capture with the SE. It is a measure of dispersion/precision of a given statistic.

When we calculate the SE for a sample mean, we are measuring by how much the sample mean is expected to be different from the population mean. In this context, the size of the sample is indispensable: with bigger samples, the difference between the sample mean and the population mean will be smaller. Thus, this difference is inversely proportional to the size of the sample, and we divide the variance by N and then take the square root. The dispersion in which we are interested here is the dispersion among the sample means derived from various samples so the size of the sample is inevitably important.

On the other hand, when we calculate the SE for the sum of 2 random variables, we are estimating the variability/precision of the sum. The sample size does not matter here, as all we care about is how the 2 variables vary independently and how they vary together.

- What do we do with bootstrap estimates? Hint: how does the underlying distribution of our data matter when we are bootstrapping?
  - Answer: don't divide by $\sqrt{n}$ because the whole point of bootstrapping is to capture the variability in the statistic without making any assumptions about the distribution of the underlying data

Regression Continued

```
#run this code to load the dataset
data_location <- "https://github.com/UChicago-pol-methods/IntroQSS-F23/r

load(url(paste0(data_location, "prestige.Rdata")))
```

Let's continue working with the Prestige dataset. Regress prestige on the type of the industry.

- What kind of variable is the type of industry?
  - Answer: categorical

```r
# your code here
reg_1 <-lm(prestige~type, prestige)
coef(reg_1)
```

```
(Intercept)    typeprof      typewc 
  35.527273   32.321114    6.716206 
```

- How would you interpret these regression coefficients?
- What is the omitted category?
- What is the estimated average prestige of a working class job?

```r
sum(coef(reg_1)[c("(Intercept)", "typewc")])
```

```
[1] 42.24348
```

Standard Error

Task: estimate the standard error for the estimated average prestige of working class jobs.

Recall the formula for the variance of the sum of random variables : $Var[X+Y] = Var[X] + Var[Y] + 2Cov[X, Y]$.

- Question: Consider the regression above. How would you express the variance of the perceived prestige of a "working class" job as a sum of 2 variables?
    - Answer: $Var[intercept + typewc] = Var[intercept] + Var[typewc] + 2Cov[intercept, typewc]$

Thankfully, R calculates these variances and covariances and stores them in a "variance covariance" matrix. It is a square matrix i.e. # of rows = # of columns. The elements along the main diagonal of the matrix are the variances of each regression parameter, while the off-diagonal elements are the covariances of pairs of variables.

$$\Sigma = \begin{bmatrix} \text{Var}(X_1) & \text{Cov}(X_1, X_2) & \dots & \text{Cov}(X_1, X_n) \\ \text{Cov}(X_2, X_1) & \text{Var}(X_2) & \dots & \text{Cov}(X_2, X_n) \\ \vdots & \vdots & \ddots & \vdots \\ \text{Cov}(X_n, X_1) & \text{Cov}(X_n, X_2) & \dots & \text{Var}(X_n) \end{bmatrix}$$

- Question: What would be the dimensions of the variance-covariance matrix for the regression we ran above?
    - Answer: 3x3

Let's print the variance-covariance matrix for the regression above and see what it looks like.

```
(vcov1 <- vcov(reg_1))
```

```
            (Intercept)  typeprof    typewc
(Intercept)    2.050546 -2.050546 -2.050546
typeprof      -2.050546  4.960998  2.050546
typewc        -2.050546  2.050546  5.973329
```

- Can you read the covariance of the intercept and typewc off of the matrix? How about the covariance of typeprof and typewc?
    - Answer:
- Interpret the covariance between the intercept and typewc.
    - Answer: This means that the covariance between the coefficient estimate for the intercept and the coefficient estimate for the job being working class is -2.050546. In other words, when the slope on typewc is large, the intercept is small, and vice-versa.

Let's get back to the task on hand, which is estimating the standard error for the estimate of average prestige of working class jobs. Now that we have the variance-covariance matrix, we need to extract the relevant variances and covariances. Which values do we need to extract here?

- How can we extract them?
    - Recall:
        - which rows do we want?
        - which columns do we want?
        - how can we extract select rows and columns from a matrix?

```
#partial vcov
coef_names <- c("(Intercept)", "typewc")
(vcov_part <- vcov1[coef_names,coef_names])
```

```
            (Intercept)    typewc
(Intercept)    2.050546 -2.050546
typewc        -2.050546  5.973329
```

And finally we are ready to calculate the SE. Do we divide by n here?

```
#calculate se
sqrt(sum(vcov_part))
```

```
[1] 1.980602
```

Let's make sure we are on the same page about the SE we calculated above. Extract the standard errors from the model summary for the regression above.

```
summary(reg_1)$coefficients[, "Std. Error"]
```

```
(Intercept)    typeprof      typewc 
   1.431973    2.227330    2.444039 
```

- How would you interpret the SE on typewc here?
    - Answer: it is the standard error of our estimate for the regression coefficient on typwc.
- How does it relate to the SE we calculated above? Hint: look at the variance-covariance matrix.
    - Answer: The square of the SE from the model summary is the variance of the regression coefficient on typewc. What we calculated before was the SE on estimated average prestige of jobs that are working class.

```
sqrt(vcov_part["typewc","typewc"])
```

```
[1] 2.444039
```

- Are we assuming homoskedasticity here? Why or why not?
    - Answer: Yes we are assuming homoskedasticity because we used classical standard errors. We can correct for it by using `lm_robust()` .

(More) Data Wrangling Skills

So far, we have created empty vectors and added individual values to them one by one. Some questions on this week's problem set require you to create an empty storage unit and then add vectors to them one by one.

Example: Suppose you have to sample with replacement 100 times, calculate the means for 3 variables in your dataset in one vector, and then store them all together.

- What would be the appropriate data structure for storing vectors?
- What should the storage ….. look like ie what should its dimensions be?

```
# creating a .....
```

```
# naming rows and columns
```

Revisiting [] brackets:

```
two_times <- c(2,4,6,8,10,12)
two_times[two_times>8]
```

```
[1] 10 12
```

```
two_times>8
```

```
[1] FALSE FALSE FALSE FALSE  TRUE  TRUE
```

```
two_times <- c(2, 4, 6, 8, 10, 12, 14, 16, 18)
logicals <- c(TRUE, FALSE, TRUE, FALSE, TRUE, TRUE, TRUE, TRUE, FALSE)
two_times[!logicals]
```

```
[1]  4  8 18
```

```
!logicals
```

```
[1] FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE
```

Regression with Interaction (cover if there is time)

```
int_reg <-lm(prestige~income + women + income*women, prestige)
```

```
coef(int_reg)
```

```
  (Intercept)        income         women  income:women 
 2.392333e+01  2.580530e-03 -1.645859e-01  7.336995e-05 
```

How would you interpret these coefficients?

- intercept ie β_0:
- coefficient on income ie β_1:
- coefficient on women ie β_2:
- coefficient on the interaction ie β_3: