Problem Set 3

Answer Key

ECON 480 — Fall 2021

Answers generally go above and beyond what I expect from you. They are meant to show you the correct answer, explain *why* it is correct, and potentially show *several methods* by which you can reach the answer.

Concepts

Question 1

In your own words, describe what exogeneity and endogeneity mean, and how they are related to bias in our regression. What things can we learn about the bias if we know X is endogenous?

The OLS estimators $\hat{\beta}_0$ and $\hat{\beta}_1$ are unbiased estimates of the true population parameters β_0 and β_1 if and only if X is *exogenous.* That is to say, if $cor(X, u) = 0$ (i.e. there is no correlation between X and any unobserved variable that affects Y), then $E[\hat{\beta}_1] = \beta_1$.

If X *is* correlated with the error term, then X is *endogenous.* The true expected value of the OLS estimator is

$$E[\hat{\beta}_1] = \beta_1 + cor(X, u)\frac{\sigma_u}{\sigma_X}$$

The bias is $(E[\hat{\beta}_1] - \beta_1)$, i.e. the difference between average estimated sample slope and the 'true' population slope, so we can determine first the *size* of the bias based on how large $cor(X, u)$ is. The stronger the correlation, the larger the bias.

Second, we can determine the *direction* of the bias depending on the sign of $cor(X, u)$.

- If X and u are positively correlated (move in the same direction), we know that we have *overstated* the true effect of ΔX on ΔY, since a change in Y is picking up both a change in X and a further change (in the same direction as X) in the unobserved u.
- If the correlation is negative (move in opposite directions), we know that we have *understated* the true effect of ΔX on ΔY, since a change in Y is picking up both a change in X that is dampened by a change in the opposite direction of u.

Question 2

In your own words, describe what R^2 means. How do we calculate it, what does it tell us, and how do we interpret it?

The R^2 is a measure of how well the OLS regression line "fits" our observed data points. It is the proportion of the total variation in Y (TSS) that is *explained* by the variation from our model (ESS):

$$R^2 = \frac{ESS}{TSS} = \frac{\sum(\hat{Y}_i - \bar{Y})^2}{\sum(Y_i - \bar{Y})^2}$$

Equivalently, it can be found by subtracting the proportion of *unexplained* variation in Y (SSE/TSS) from 1:

$$R^2 = 1 - \frac{SSE}{TSS} = 1 - \frac{\sum(u_i)^2}{\sum(Y_i - \bar{Y})^2}$$

This is because $\frac{SSE+ESS}{TSS} = 1$. Finally, R^2 is the square of the correlation coefficient between X and Y, $R^2 = (r_{X,Y})^2$

The closer R^2 is to 1, the better the fit, the closer to 0, the poorer the fit.

Question 3

In your own words, describe what the standard error of the regression (SER) means. How do we calculate it, what does it tell us, and how do we interpret it?

SER ($\hat{\sigma_u}$) is the average size of the error (or residual), $\hat{u_i}$, that is, the average distance from the regression line to the actual data value for Y at a given X. The goal of OLS is to minimize this (well, technically minimize the *sum* of *squared* errors!).

$$SER = \sqrt{\frac{1}{n-2}\sum \hat{u_i}^2}$$
$$SER = \sqrt{\frac{SSE}{n-2}}$$

We calculate it by squaring the residuals (to get a positive distance) and taking the mean of them by adding them all up and dividing by $n-2$, and then taking the square root to return to normal (non-squared) units.

We divide by $n-2$ rather than by n due to the degrees of freedom correction for calculating two prior statistics with our data already, $\hat{\beta_0}$ and $\hat{\beta_1}$.

Question 4

In your own words, describe what homoskedasticity and heteroskedasticity mean: both in ordinary English, and in terms of the graph of the OLS regression line.

Homoskedasticity means the errors are distributed with the same variance for all levels of X. Knowing anything about X will not tell us anything about the distribution of errors at that level of X.

Heteroskedasticity means the errors are distributed differently for different levels of X. So, at different levels of X, there will be much more or much less variation in the residuals.

Question 5

In your own words, describe what the variation in $\hat{\beta_1}$ (either variance or standard error) means, or is measuring. What three things determine the variation, and in what way?

The variation of $\hat{\beta_1}$ (either it's variance or standard error) is a measure of how *precise* our estimate is. This idea comes from the sampling distribution of $\hat{\beta_1}$, since it is a random variable: if we were to take other samples and calculate the slope of a regression line $\hat{\beta_1}$ for each, the estimate would vary from sample to sample.

The standard error of $\hat{\beta_1}$ (square this to get variance) is:

$$se(\hat{\beta_1}) = \frac{\sigma_u}{\sqrt{n} \times se(X)}$$

The three things that affect it are:

1. Goodness of Fit of the Regression (σ_u) or SER. The worse the fit, the higher the SER, and the worse the precision (higher standard error) of $\widehat{\beta}_1$.
2. Sample size, n: the more data, the better the precision (lower standard error) of $\widehat{\beta}_1$.
3. Standard error of X: the more variation (spread) in X-values, the better the precision (lower standard error) of $\widehat{\beta}_1$.

See the graphs in slides 6-8 of class 2.5 for more.

Question 6

In your own words, describe what a p-value means, and how it is used to establish statistical significance.

The p-value is the probability that, if the null hypothesis were true, of observing a test statistic at least as extreme as the one found in our sample. Specifically, if $H_0 : \beta_1$, it is the probability of getting a sample slope at least as extreme as the one in our sample, if the slope were truly 0.[1]

$$Prob(\delta \geq \delta_i | H_0 \text{ is true})$$

where δ is a test-statistic and δ_i is the test statistic we obtained from our sample.

Another way to interpret this is that the p-value is the probability we commit a Type I error: the probability that, if the null hypothesis were true, we falsely reject it from our sample evidence.

Be careful, the p-value is not the probability that our alternative hypothesis is true given our findings (commonly believed)! In fact it is basically the opposite, the probability of our findings being valid given the null hypothesis!

Theory and Applications

Question 7

A researcher is interested in examining the impact of illegal music downloads on commercial music sales. The author collects data on commercial sales of the top 500 singles from 2017 (Y) and the number of downloads from a web site that allows 'file sharing' (X). The author estimates the following model

$$\text{music sales}_i = \beta_0 + \beta_1 \text{illegal downloads}_i + u_i$$

The author finds a large, positive, and statistically significant estimate of $\widehat{\beta}_1$. The author concludes these results demonstrate that illegal downloads actually *boost* music sales. Is this an unbiased estimate of the impact of illegal music on sales? Why or why not? Do you expect the estimate to overstate or understate the true relationship between illegal downloads and sales?

Does knowing the amount of illegal downloads an artist has convey any information about other variables that affect music sales? In other words, we are asking if $E[u|X] = 0$ (or more simply, $cor(X, u) = 0$).

[1] Note in the classic sense, the p-value is actually measuring the probability of a *test statistic* (t) being at least as extreme as ours. The test statistic essentially standardizes our sample statistic ($\widehat{\beta}_1$) so that it measures standard deviations from the null-hypothesized value (i.e. 0), much like a Z-score.

It is likely that artists and songs that are the most heavily pirated are the most popular ones, and also are likely have very high music sales. Economists say piracy is like a tax on success–it happens more to those who are already successful and less to those who are still trying to make it big.

In any case, illegal downloads is probably endogenous. Since there is likely a positive correlation between music sales and popularity (in the error term), and popularity is also positively correlated with music sales, it is likely that we are *overstating* the effect of illegal downloads on sales. In other words, $\hat{\beta}_1$ is also picking up the positive effect of popular songs, and is too large. The true estimate of β_1 is likely much lower than measured.

Question 8

A researcher wants to estimate the relationship between average weekly earnings (AWE, measured in dollars) and Age (measured in years) using a simple OLS model. Using a random sample of college-educated full-time workers aged 25-65 yields the following:

$$\widehat{AWE} = 696.70 + 9.60\, Age$$

Part A

Interpret what $\hat{\beta}_0$ means in this context.

$\hat{\beta}_0$ is 696.70. This is the vertical intercept of the regression line. It means that a person that is 0 years old earns a \$696.70 per week on average. This is often nonsensical, so we don't often care about the economic meaning of the intercept.

Part B

Interpret what $\hat{\beta}_1$ means in this context.

$\hat{\beta}_1$ is 9.60 This is the slope of the regression line. It means that for every year older a person is, they can expect their wages to increase by \$9.60, on average. This is the marginal effect of Age on AWE (and the causal effect if this model were exogenous).

Part C

The $R^2 = 0.023$ for this regression. What are the units of the R^2, and what does this mean? R^2 has no units, it is the proportion of variation in AWE that is explained by our model, between 0 and 1. This model explains only 2.3% of the variation in AWE, meaning this model is poor, and the line does not fit the data points well.

Part D

The SER, $\hat{\sigma_u} = 624.1$ for this regression. What are the units of the SER in this context, and what does it mean? Is the SER large in the context of this regression?

SER is measured in the same units as the dependent variable, AWE, so it is measured in dollars. It is the average error or residual for an individual, the difference (in dollars) between OLS' predicted $\widehat{AWE}$ for that person, and their true AWE in the data. This SER is quite big, \$624 in average weekly earnings.

Part E

Suppose Maria is 20 years old. What is her predicted $\widehat{AWE}$?

$$\begin{aligned}\widehat{AWE}_{Maria} &= 696.70 + 9.60(20) \\ &= 888.70\end{aligned}$$

She is predicted to earn $888.70 per week, according to our model.

Part F

Suppose the data shows her *actual* AWE is $430. What is her residual? Is this a relatively good or a bad prediction?[2]

$$\begin{aligned}\hat{u}_{Maria} &= Y_{Maria} - \widehat{Y}_{Maria} \\ &= 430 - 888.70 \\ &= -458.70\end{aligned}$$

Her residual, i.e. the error in the prediction of her wages, is -$488.70 (she *actually* earns $488.70 less than her predicted wage).

While this sounds large, it actually a relatively good prediction, as it is much lower than the average prediction error (SER), which was $624.10.

Part G

What does the error term, $\hat{u}_i$ represent in this case? What might individuals have different values of u_i?

The error term represents *all* factors *other* than age that affects an individual's average weekly earnings. This could include things like experience, ability, job type, education level, conscienciousness etc.

Part H

Do you think that Age is exogenous? Why or why not? Would we expect $\hat{\beta}_1$ to be too *large* or too *small*?

It's very unlikely that Age is exogenous. Knowing someone's age likely gives us information about u: we can guess about their experience or level of education (they are likely higher for older people), and most of these positively affect wages. Thus, we have probably *overstimated* the effect of age on earnings (i.e. $\hat{\beta}_1$), and the true β_1 is likely smaller.

Question 9

Suppose a researcher is interested in estimating a simple linear regression model:

$$Y_i = \beta_0 + \beta_1 X_i + u_i$$

In a sample of 48 observations, she generates the following descriptive statistics:

- $\bar{X} = 30$
- $\bar{Y} = 63$
- $\sum_{i=1}^{n}(X_i - \bar{X})^2 = 6900$
- $\sum_{i=1}^{n}(Y_i - \bar{Y})^2 = 29000$
- $\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y}) = 13800$
- $\sum_{i=1}^{n}\hat{u}^2 = 1656$

[2]Hint: compare your answer here to your answer in Part D.

Part A

What is the OLS estimate of $\hat{\beta}_1$?

The formula for $\hat{\beta}_1 = \dfrac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sum_{i=1}^{n}(X_i - \bar{X})^2} = \frac{cov(X,Y)}{var(X)} = \frac{13800}{6900} = 2$

Part B

What is the OLS estimate of $\hat{\beta}_0$?

The formula for $\hat{\beta}_0 = \bar{Y} - \hat{\beta}_1 \bar{X} = 63 - 30(2) = 3$

Part C

Suppose the OLS estimate of $\hat{\beta}_1$ has a standard error of 0.072. Could we probably reject a null hypothesis of $H_0 : \beta_1 = 0$ at the 95% level?

Yes, we could reject the null hypothesis as the estimate of $\hat{\beta}_1 = 2$ is more than 2 times its standard error of 0.072. The test-statistic would actually be

$$\begin{aligned} t &= \frac{\hat{\beta}_1 - \beta_{1,0}}{se(\hat{\beta}_1)} \\ t &= \frac{2-0}{0.072} \\ t &\approx 27.78 \end{aligned}$$

This is well beyond the critical value needed to reject H_0, and the p-value would be basically 0.

Part D

Calculate the R^2 for this model. How much variation in Y is explained by our model?

We know TSS (4th bullet point) and SSE (last bullet point).

$$\begin{aligned} R^2 &= 1 - \frac{SSE}{TSS} \\ &= 1 - \frac{1656}{29000} \\ &= 1 - 0.057 \\ &= 0.943 \end{aligned}$$

This model explains 94.3% of the variation in Y_i.

Part E

How large is the average residual?

We need to find the standard error of the regression (SER), but luckily we know the SSE (last bullet point)

$$
\begin{aligned}
SER &= \sqrt{\frac{SSE}{n-2}} \\
&= \sqrt{\frac{1656}{48-2}} \\
&= \sqrt{36} \\
&= 6
\end{aligned}
$$

This tells us the average residual is 36 (units of Y).

R Questions

Answer the following questions using `R`. When necessary, please write answers in the same document (knitted `Rmd` to `html` or `pdf`, typed `.doc(x)`, or handwritten) as your answers to the above questions. Be sure to include (email or print an `.R` file, or show in your knitted `markdown`) your code and the outputs of your code with the rest of your answers.

Question 10

- `mlbattend.csv`

Download the `MLBattend` dataset. This data contains data on attendance at major league baseball games for all 32 MLB teams from the 1970s-2000. We want to answer the following question:

> "How big is home-field advantage in baseball? Does a team with higher attendance at home games over their season have score more runs over their season?"

Part A

Clean up the data a bit by `mutate()`-ing a variable to measure home attendance in millions. This will make it easier to interpret your regression later on.

```
# first load tidyverse
library(tidyverse)

# import data, save as mlb
mlb <- read_csv("../data/MLBAttend.csv") # path on my website is different!
```

```
# make home attendance variable in millions
mlb <- mlb %>%
  mutate(home_attend_mil = home_attend/1000000)
```

Part B

Get the correlation between Runs Scored and Home Attendance.

```
# summarize and get correlation
mlb %>%
  summarize(Correlation = cor(runs_scored, home_attend_mil))
```

```
## # A tibble: 1 x 1
##   Correlation
##         <dbl>
## 1       0.494
```

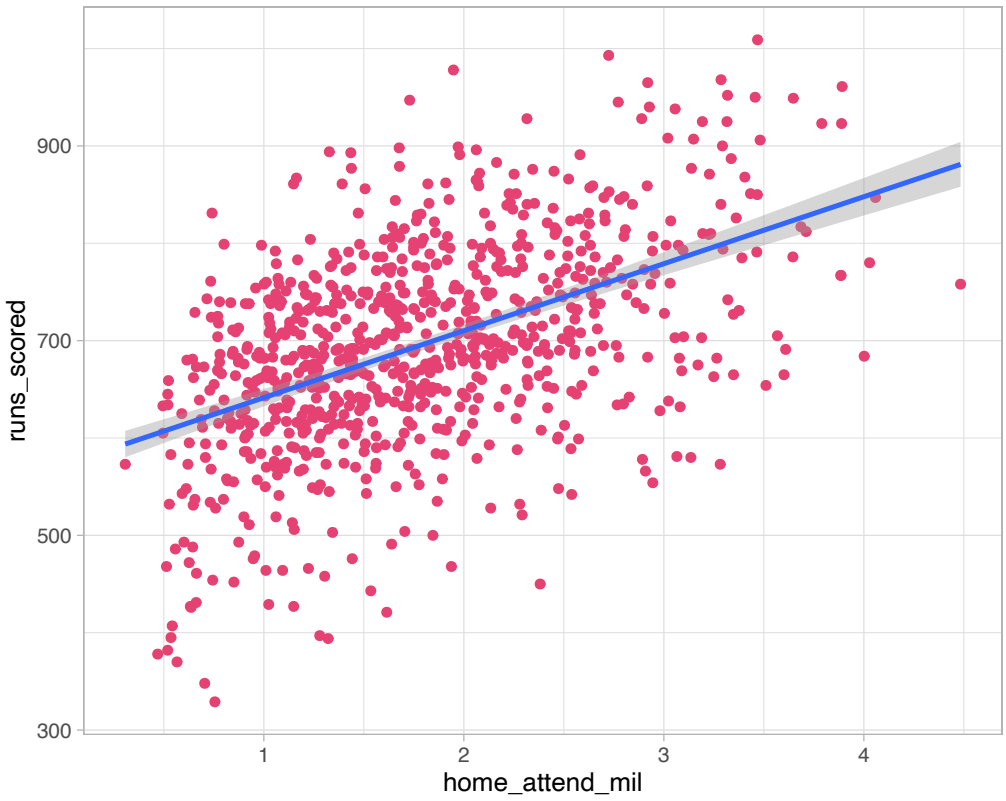
The correlation is 0.49, indicating a moderate positive relationship.

Part C

Plot a scatterplot of Runs Scored (y) on Home Attendance (x). Add a regression line.

```
# create scatterplot with regression line
scatter <- ggplot(data = mlb)+
  aes(x = home_attend_mil,
      y = runs_scored)+
  geom_point(color = "#e64173")+
  geom_smooth(method = "lm")+
  theme_light()

# look at it
scatter
```



Part D

We want to estimate a regression of Runs Scored on Home Attendance:

$$\widehat{\text{runs scored}}_i = \beta_0 + \beta_1 \text{home attendance}_i$$

Run this regression in R.

What are $\hat{\beta}_0$ and $\hat{\beta}_1$ for this model? Interpret them in the context of our question. [Hint: make sure to save your regression model as an object, and get a summary() of it. This object will be needed later.]

```
# run regression, save as reg
reg <- lm(runs_scored ~ home_attend_mil, data = mlb)
```

```r
# get summary of reg
summary(reg)
```

```
##
## Call:
## lm(formula = runs_scored ~ home_attend_mil, data = mlb)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -295.566  -52.754    1.414   63.769  271.377
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      572.618      8.081   70.86   <2e-16 ***
## home_attend_mil   68.798      4.183   16.45   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 91.47 on 836 degrees of freedom
## Multiple R-squared:  0.2445, Adjusted R-squared:  0.2436
## F-statistic: 270.5 on 1 and 836 DF,  p-value: < 2.2e-16
```

```r
# Here I'm going to save beta 0 hat and beta 1 hat
# as objects to call up in the text of the markdown document
# We'll need broom and to tidy() our regression first
library(broom)
reg_tidy <- tidy(reg)

reg_tidy
```

```
## # A tibble: 2 x 5
##   term            estimate std.error statistic  p.value
##   <chr>              <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)        573.       8.08      70.9 0
## 2 home_attend_mil     68.8      4.18      16.4 7.13e-53
```

```r
# extract and save beta 0 hat

beta_0_hat <- reg_tidy %>%
  filter(term == "(Intercept)") %>% # look at intercept row
  pull(estimate) %>% # extract beta 0 hat
  round(., 3) # round to 3 decimal places

beta_0_hat # check
```

```
## [1] 572.618
```

```r
# extract and save beta 1 hat

beta_1_hat <- reg_tidy %>%
  filter(term == "home_attend_mil") %>% # look at X-variable row
  pull(estimate) %>% # extract beta 1 hat
  round(., 3) # round to 3 decimal places

beta_1_hat # check
```

```
## [1] 68.798
```

Part E

Write out the estimated regression equation.

$$\widehat{\text{Runs scored}}_i = 572.618 - 68.798 \text{ Home attendance (mil)}$$

```
# if you are using markdown, try out the equatiomatic package
#install.packages("equatiomatic")
library(equatiomatic)
extract_eq(reg, # the regression
           use_coefs = TRUE, # use the estimated numbers
           coef_digits = 3, # how many digits to show
           fix_signs = TRUE) # fix negatives
```

$$\hat{\text{runs\_scored}} = 572.618 + 68.798(\text{home\_attend\_mil})$$

Part F

Make a regression table of the output (using the `huxtable` package).

```
# load huxtable
library(huxtable)
huxreg(reg, # this is sufficient, the rest is customization
       coefs = c("Constant" = "(Intercept)",
                 "Home Attendance (Millions)" = "home_attend_mil"),
       statistics = c("N" = "nobs",
                      "R-Squared" = "r.squared",
                      "SER" = "sigma"),
       number_format = 3)
```

	(1)
Constant	572.618 ***
	(8.081)
Home Attendance (Millions)	68.798 ***
	(4.183)
N	838
R-Squared	0.244
SER	91.473

*** p < 0.001; ** p < 0.01; * p < 0.05.

Part G

Check the goodness of fit statistics. What is the R^2 and the SER of this model? Interpret them both in the context of our question.

```r
# Here I'm going to save these
# as objects to call up in the text of the markdown document
# Again we need broom and to glance() our regression first
r_sq <- glance(reg) %>%
  pull(r.squared) %>%
  round(., 3)

r_sq # check
```

```
## [1] 0.244
```

```r
ser <- glance(reg) %>%
  pull(sigma) %>%
  round (., 3)

ser # check
```

```
## [1] 91.473
```

R^2 is 0.244, meaning the model is able to explain about 24.4% of the variation in Runs Scored.

The SER, i.e. the average error is 91.473, meaning any team's season (one observation) has on average 91.473 more/fewer runs than its predicted number of runs.

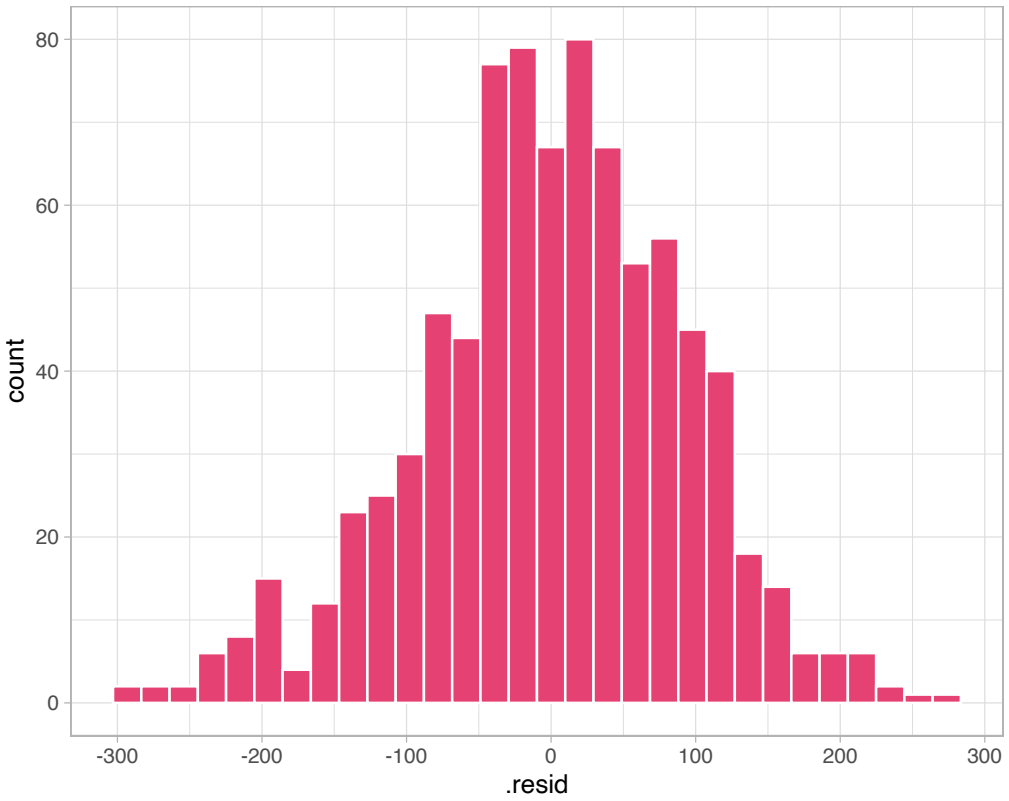
Part H

Now let's start running some diagnostics of the regression. Make a histogram of the residuals. Do they look roughly normal? [Hint: you will need to use the `broom` package's `augment()` command on your saved regression object to add containing the residuals (`.resid`), and save this as a new object - to be your data source for the plot in this question and the next question.]

```r
# here we need broom's augment() command to add residuals to the data

# load broom
library(broom)
# augment the regression, save as reg_aug
reg_aug <- reg %>%
  augment()

# now we use this as the data in our histogram plot in ggplot, (x is .resid)
ggplot(data = reg_aug)+
  aes(x = .resid)+
  geom_histogram(color = "white", fill = "#e64173")+
  theme_light()
```



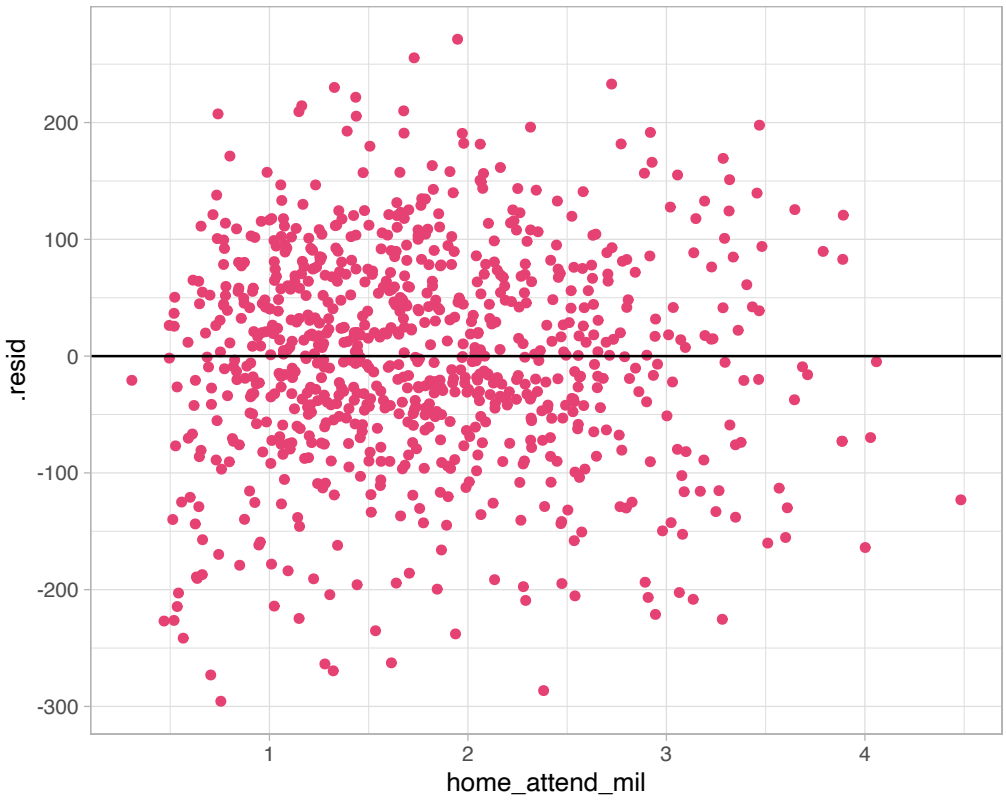
They do look roughly normally distributed.

Part I

Make a residual plot.

```
# this is another plot from reg_aug
# where x is home attendance and y is .fitted

ggplot(data = reg_aug)+
  aes(x = home_attend_mil,
      y = .resid)+
  geom_point(color = "#e64173")+
  geom_hline(yintercept = 0, color = "black")+
  theme_light()
```



Part J

Test the regression for heteroskedasticity. Are the errors homoskedastic or heteroskedastic?

[Hint: use the `lmtest` package's `bptest()` command on your saved regression object.]

Run another regression using robust standard errors. [Hint: use the `estimatr` package's `lm_robust()` command and save the output like the following:

```
reg_robust <-lm_robust(y ~ x, data = the_data, # change y, x, and data names to yours
                              se_type = "stata") # we'll use this method to calculate
```

Now make another regression output table with `huxtable`, with one column using regular standard errors (just use your original saved regression object) and another using robust standard errors (use this new saved object).

```
# First, testing for heteroskedasticity
## this requires the lmtest package for the bptest() command

# install.packages("lmtest")

library(lmtest) # load lmtest

bptest(reg)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  reg
## BP = 0.22515, df = 1, p-value = 0.6351
```

The null hypothesis H_0 is that the errors are homoskedastic. The p-value for this test is very large, so we *cannot* reject the null hypothesis.

This is good, it means the errors are homoskedastic, and our OLS estimators' standard errors are accurate and do not need to be corrected for heteroskedasticity.

```
# Now, creating robust standard errors
## this requires the estimatr package for the lm_robust() command

# install.packages("estimatr")

library(estimatr) # load

reg_robust <-lm_robust(runs_scored ~ home_attend_mil, data = mlb, # change y, x, and data names to your
                       se_type = "stata") # we'll use this method to calculate

summary(reg_robust) # look at it
```

```
##
## Call:
## lm_robust(formula = runs_scored ~ home_attend_mil, data = mlb,
##     se_type = "stata")
##
## Standard error type:  HC1
##
## Coefficients:
##                 Estimate Std. Error t value  Pr(>|t|) CI Lower CI Upper  DF
## (Intercept)        572.6      8.701   65.81 0.000e+00   555.54   589.70 836
## home_attend_mil     68.8      4.600   14.96 5.344e-45    59.77    77.83 836
##
## Multiple R-squared:  0.2445 ,    Adjusted R-squared:  0.2436
## F-statistic: 223.7 on 1 and 836 DF,  p-value: < 2.2e-16
```

```
tidy(reg_robust) # can look in broom as well
```

```
##              term  estimate std.error statistic      p.value  conf.low
## 1     (Intercept) 572.61829  8.700802  65.81213 0.000000e+00 555.54031
## 2 home_attend_mil  68.79777  4.599561  14.95746 5.344131e-45  59.76973
##   conf.high  df     outcome
## 1 589.69628 836 runs_scored
## 2  77.82582 836 runs_scored
```

```
# Now let's compare by making a side-by-side regression table
## the first two lines are sufficient, beyond is just customization
huxreg("Non-Robust SEs" = reg, # 1st column is reg model, title it "Non-Robust SEs"
       "Robust SEs" = reg_robust, # 2nd column is reg_robust model, title it "Robust SEs"
       coefs = c("Constant" = "(Intercept)",
                 "Home Attendance (Millions)" = "home_attend_mil"),
       statistics = c("N" = "nobs",
                      "R-Squared" = "r.squared",
                      "SER" = "sigma"),
       number_format = 3)
```

Observe how neither $\hat{\beta}_0$ nor $\hat{beta}_1$ changes due to using robust standard errors (again, heteroskedasticity does **not** bias our estimates!), but using robust standard errors (correcting for heteroskedasticity) *does* slightly increase the standard errors (making the test statistic *slightly* smaller and p-values *slightly* larger).

	Non-Robust SEs	Robust SEs
Constant	572.618 ***	572.618 ***
	(8.081)	(8.701)
Home Attendance (Millions)	68.798 ***	68.798 ***
	(4.183)	(4.600)
N	838	838
R-Squared	0.244	0.244
SER	91.473	

*** p < 0.001; ** p < 0.01; * p < 0.05.

Part K

Test the data for outliers. If there are any, identify which team(s) and season(s) are outliers. [Hint: use the car package's `outlierTest()` command on your saved regression object.]

```
# this requires the car package for the outlierTest() command

# install.packages("car")

library(car) # load car

outlierTest(reg)
```

```
## No Studentized residuals with Bonferroni p < 0.05
## Largest |rstudent|:
##      rstudent unadjusted p-value Bonferroni p
## 816 -3.255201          0.0011787      0.98779
```

This test detected one outlier, which is observation (row) number `816`. Let's look it up:

```
mlb %>%
  slice(816)
```

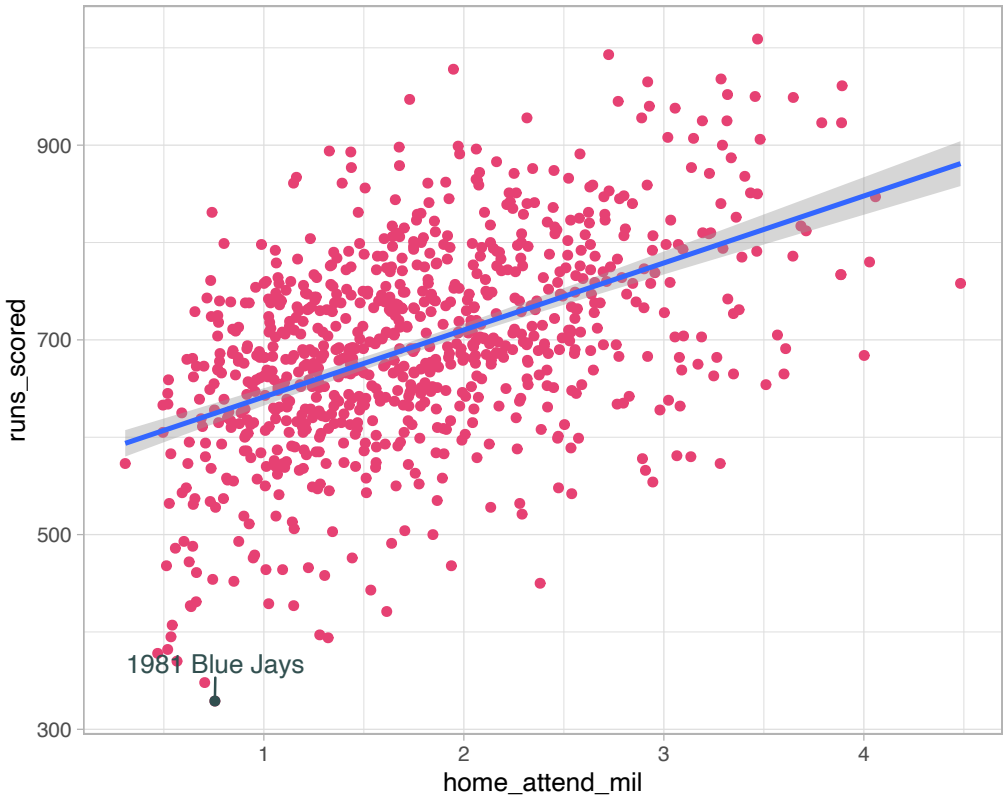
```
## # A tibble: 1 x 13
##   team  city    nickname  league division season home_attend runs_scored
##   <chr> <chr>   <chr>     <chr>  <chr>     <dbl>       <dbl>       <dbl>
## 1 TOR   Toronto Blue Jays AL     East       1981      755083         329
## # ... with 5 more variables: runs_allowed <dbl>, wins <dbl>, losses <dbl>,
## #   games_behind <dbl>, home_attend_mil <dbl>
```

The Toronto Blue Jays' 1981 season is an outlier. Just for kicks, let's point it out on the scatterplot.

```
outlier <- mlb %>%
  slice(816)

library(ggrepel)
scatter+ # our scatterplot saved from part C
  geom_point(data = outlier,
             aes(x = home_attend_mil,
```

```
                y = runs_scored),
            color = "#314f4f")+
geom_text_repel(data = outlier,
            aes(x = home_attend_mil,
                y = runs_scored),
            label = "1981 Blue Jays",
            color = "#314f4f",
            box.padding = 0.75,
            seed = 20)
```



Part L

Look back at your regression results. What is the marginal effect of home attendance on runs scored? Is this statistically significant? Why or why not?

This is an interpretation question, no need to calculate anything. The marginal effect is $\hat{\beta}_1$: for every 1 additional million fans attending home games over a team's season, the team scores 69 more runs.

Looking back at the regression output in part c, the t-score for the hypothesis test $H_0 : \beta_1 = 0$, $H_1 : \beta_1 \neq 0$ is 16.45, yielding a very very small p-value. We have sufficient evidence to reject H_0 in favor of our alternative hypothesis, that there is a relationship between home attendance and runs scored over a season.

Part M

Now we'll try out the `infer` package to understand the p-value for our observed slope in our regression model.

First, save the (value of) our sample $\hat{\beta}_1$ from your regression in Part D as an object, I suggest:

```
our_slope = 123 # replace "123" with whatever number you found for the slope in part D
```

Then, install and load the `infer` package, and then run the following simulation:

```
# save our simulations as an object (I called it "sims")
sims <- data %>% # "data" here is whatever you named your dataframe!
  specify(y ~ x) %>% # replacing y and x with your variable names
  hypothesize(null = "independence") %>% # H_0 is that slope is 0, x and y are independent
  generate(reps = 1000,
           type = "permute") %>% # make 1000 samples assuming H_0 is true
  calculate(stat = "slope") # estimate slope in each sample

# look at it
sims

# calculate p value
sims %>%
  get_p_value(obs_stat = our_slope,
              direction = "both") # a two-sided H_a: slope =/= 0
```

Compare to the p-value in your original regression output in previous parts of this question.

```
# install.packages("infer")
library(infer) # load infer

# recall I already saved our slope as beta_1_hat:
beta_1_hat
```

```
## [1] 68.798
```

```
sims <- mlb %>% # our data
  specify(runs_scored ~ home_attend_mil) %>% # our model
  hypothesize(null = "independence") %>% # H_0 is that slope is 0, x and y are independent
  generate(reps = 1000,
           type = "permute") %>% # make 1000 samples assuming H_0 is true
  calculate(stat = "slope") # estimate slope in each sample

# look at it
sims
```

```
## Response: runs_scored (numeric)
## Explanatory: home_attend_mil (numeric)
## Null Hypothesis: independence
## # A tibble: 1,000 x 2
##    replicate  stat
##        <int> <dbl>
##  1         1 -5.90
##  2         2 -3.60
##  3         3 -3.07
##  4         4 -3.95
##  5         5 -1.86
##  6         6 -4.05
##  7         7 -1.26
##  8         8 -3.68
##  9         9 -2.97
## 10        10  7.33
```

```
## # ... with 990 more rows
```

```
# calculate p value
sims %>%
  get_p_value(obs_stat = beta_1_hat, # our slope
              direction = "both") # a two-sided H_a: slope =/= 0
```

```
## # A tibble: 1 x 1
##   p_value
##     <dbl>
## 1       0
```

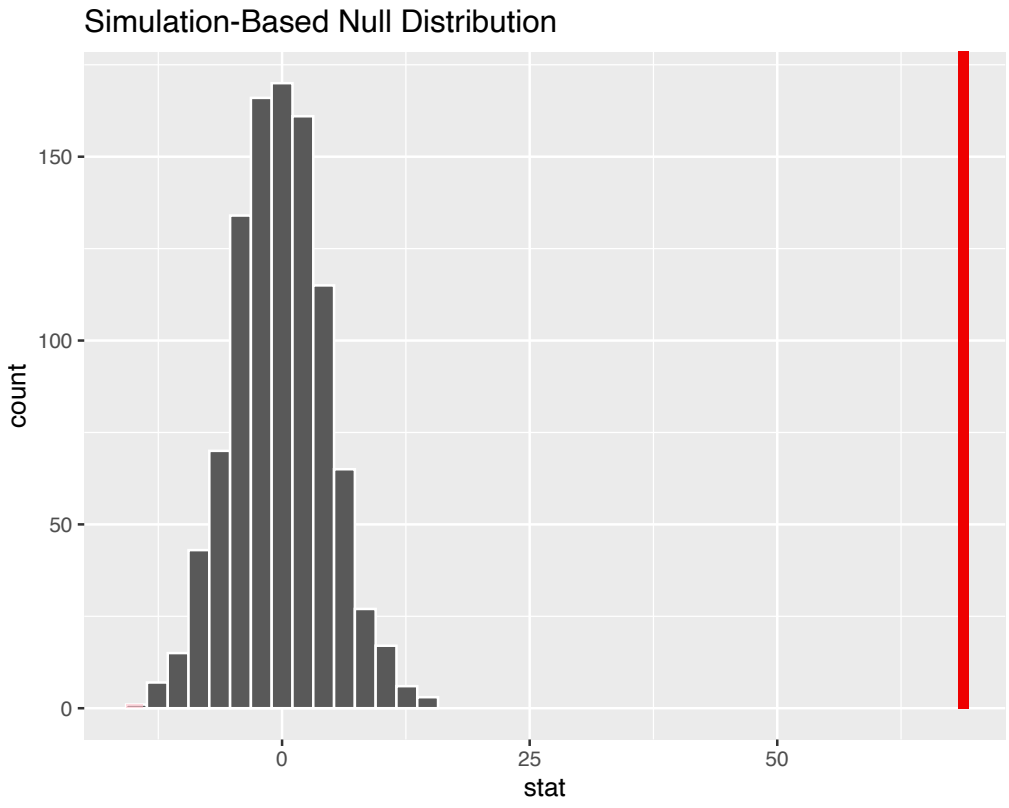
The p-value is basically 0. According to the regression output in part D, it is smaller than 0.00000000000000002.

Part N

Make a histogram of the simulated slopes, and plot our sample slope on that histogram, shading the p-value.

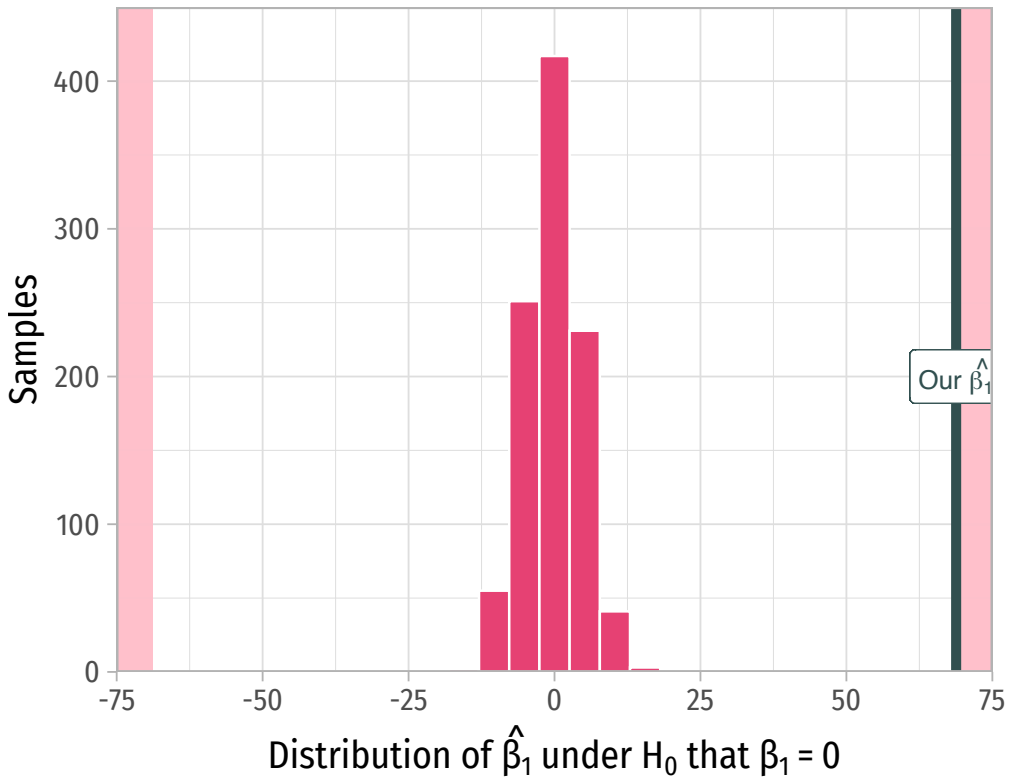
[You can pipe `sims` into `visualize(obs_stat = our_slope)`, or use `ggplot2` to plot a histogram in the normal way, using `sims` as the data source and add a `geom_vline(xintercept = our_slope)` to show our finding on the distribution.]

```
sims %>%
  visualize()+
  shade_p_value(obs_stat = beta_1_hat, # our slope
                direction = "both") # two-sided test, shade both sides
```



Again, since `visualize()` really is just using `ggplot()`, we can use `sims` as our data to make our own plot:

```r
ggplot(data = sims)+
  aes(x = stat)+ # slope from simulations
  geom_histogram(color = "white",
                 fill = "#e64173")+
  # add "shading" for p-value's two sides
  # right side
  geom_rect(xmin=beta_1_hat,
            xmax=Inf,
            ymin=0,
            ymax=Inf,
            fill = "pink",
            alpha=0.4)+
  # left side
  geom_rect(xmin=-beta_1_hat,
            xmax=-Inf,
            ymin=0,
            ymax=Inf,
            fill = "pink",
            alpha=0.4)+
  # add vertical line for our sample slope
  geom_vline(xintercept = beta_1_hat,
             color = "#314f4f",
             size = 2)+
  # add label
  geom_label(x = beta_1_hat,
             y = 200,
             color = "#314f4f",
             label = expression(paste("Our ", hat(beta[1]))))+
  scale_x_continuous(breaks=c(-75,-50,-25,0,25,50,75),
                     limits=c(-75,75),
                     expand=c(0,0))+
  scale_y_continuous(limits=c(0,450),
                       expand=c(0,0))+
  labs(x = expression(paste("Distribution of ", hat(beta[1]), " under ", H[0], " that ", beta[1]==0)),
       y = "Samples")+
    theme_light(base_family = "Fira Sans Condensed",
           base_size=16)
```



Note this is the sampling distribution of $\hat{\beta}_1$ under the null hypothesis (the true $\beta_1 = 0$). Values on the horizontal axis are values of $\hat{\beta}_1$, *not* the number of standard deviations away from the null hypothesis.

What the test-statistic t does is standardize this distribution similar to how we would standardize a distribution to the standard normal distribution via calculating Z-scores.

Let me visualize what would happen if we tried to standardize our simulated null distribution:

```
sims %>%
  summarize(mean = mean(stat), # get the mean slope of the simulated distribution of slopes
            se = (sd(stat))) # get the standard error of the slope
```

```
## # A tibble: 1 x 2
##     mean    se
##    <dbl> <dbl>
## 1 -0.209  4.71
```

```
# the mean isn't exactly 0, but close

# get OUR standard error from our original regression
se_beta_1_hat<- reg_tidy %>%
  filter(term == "home_attend_mil") %>%
  pull(std.error)

# standardize slopes to t-statistics (like Z-scores)
##  t = estimate - null hypothesis value / standard error of estimate

t_statistics <- sims %>%
  mutate(t_scores = ((stat - 0)/se_beta_1_hat))

our_t <- ((beta_1_hat - 0) / se_beta_1_hat)
```

```
our_t
```

```
## [1] 16.44736
```

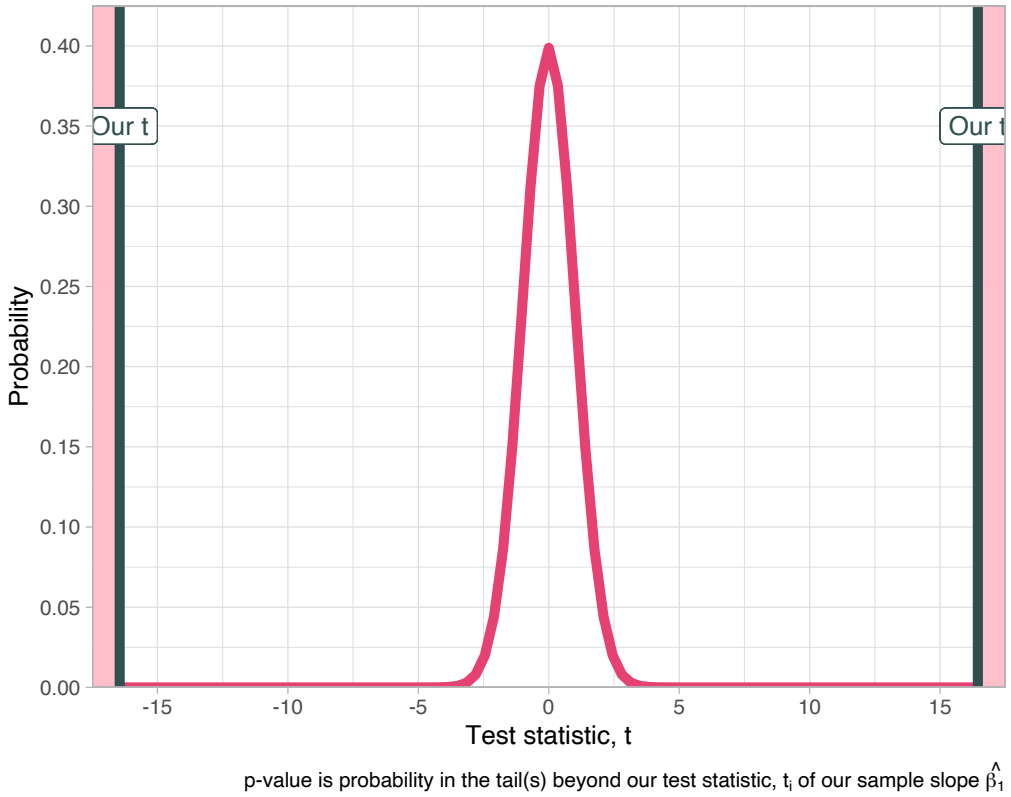
This is the `t` statistic that `R` calculates as part of the regression, and calculates the `p`-value for the above hypothesis test as the probability beyond this value in both tails of a *theoretical* t-distribution with $n-k-1$ (in this case, 836) degrees of freedom:

We can visualize the t-distribution

```
ggplot(data = tibble(x=-4:4))+
  aes(x = x)+
  stat_function(fun = dt, args = list(df = 836), size=2, color="#e64173")+
      # add "shading" for p-value's two sides
  # right side
  geom_rect(xmin=our_t,
            xmax=Inf,
            ymin=0,
            ymax=Inf,
            fill = "pink",
            alpha=0.4)+
  # left side
  geom_rect(xmin=-1* our_t,
            xmax=-Inf,
            ymin=0,
            ymax=Inf,
            fill = "pink",
            alpha=0.4)+
  # add vertical line for our sample slope's t-statistic
  geom_vline(xintercept = our_t,
             color = "#314f4f",
             size = 2)+
  #add label
  geom_label(x = our_t,
             y = 0.35,
             color = "#314f4f",
             label = "Our t")+
  geom_vline(xintercept = -1 * our_t,
             color = "#314f4f",
             size = 2)+
  #add label
  geom_label(x = -1 * our_t,
             y = 0.35,
             color = "#314f4f",
             label = "Our t")+

  scale_x_continuous(breaks = seq(-20,20,5),
                     limits = c(-17.5,17.5),
                     expand = c(0,0))+
  scale_y_continuous(breaks = seq(0,0.4,0.05),
                     limits = c(0,0.425),
                     expand = c(0,0))+
  labs(x = "Test statistic, t",
       y = "Probability",
       caption = expression(paste("p-value is probability in the tail(s) beyond our test statistic, ",
```

```
  theme_light()
```



We can calculate this probability (as R does in the regression) by finding the probability in the tails of the t_{836}-distribution beyond ± 16.447. The t distribution has 53,938 degrees of freedom — $(n - k - 1)$ where $n = 53940$ and $k = 1$.

```
2 * pt(16.447, # our t-statistic
       df = 836, # the df number
       lower.tail = F) # we'll use the right tail
```

```
## [1] 7.160046e-53
```

If the null hypothesis were true ($\beta_1 = 0$), the probability that we get a test-statistic at least as extreme as 16.447 (essentially, 16.447 standard deviations away!!) is virtually 0.