

# ScPoEconometrics

## Regression Inference

Florian Oswald, Mylene Feuillade, Gustave Kenedi and Pierre Villedieu  
SciencesPo Paris  
2022-04-07

# Quick "Quiz" on Last Week's Material

1. From your *computer* ↗ connect to [www.wooclap.com/SCPOCIHT](http://www.wooclap.com/SCPOCIHT)

*OR*

2. From your *phone* ↗ flash QR code below



# Today - Statistical inference in the regression framework

- Fully understand a *regression table*
- Compare *theory-based* and *simulation-based* inference
- *Classical Regression Model* assumptions
- Empirical applications:
  - Class size and student performance
  - Returns to education by gender



# Back to class size and student performance

- Let's go back the **STAR** experiment data, and focus on:
  - *small and regular* classes,
  - *Kindergarten* grade.



# Back to class size and student performance

- Let's go back the **STAR** experiment data, and focus on:
  - *small* and *regular* classes,
  - *Kindergarten* grade.
- We consider the following regression model and estimate it by OLS:

$$\text{math score}_i = b_0 + b_1 \text{small}_i + e_i$$



# Back to class size and student performance

- Let's go back the **STAR** experiment data, and focus on:
  - *small* and *regular* classes,
  - *Kindergarten* grade.
- We consider the following regression model and estimate it by OLS:

$$\text{math score}_i = b_0 + b_1 \text{small}_i + e_i$$

```
library(tidyverse)

star_df = read.csv("https://www.dropbox.com/s/bf1fog8y
star_df = star_df[complete.cases(star_df), ]
star_df = star_df %>%
  filter(star %in% c("small", "regular") &
         grade == "k") %>%
  mutate(small = (star == "small"))
```



# Back to class size and student performance

- Let's go back the **STAR** experiment data, and focus on:
  - *small* and *regular* classes,
  - *Kindergarten* grade.
- We consider the following regression model and estimate it by OLS:

$$\text{math score}_i = b_0 + b_1 \text{small}_i + e_i$$

```
library(tidyverse)

star_df = read.csv("https://www.dropbox.com/s/bf1fog8y
star_df = star_df[complete.cases(star_df), ]
star_df = star_df %>%
  filter(star %in% c("small", "regular") &
         grade == "k") %>%
  mutate(small = (star == "small"))
```

```
reg_star = lm(math ~ small, star_df)
reg_star

##
## Call:
## lm(formula = math ~ small, data = star_df)
##
## Coefficients:
## (Intercept)    smallTRUE
##           484.446          8.895
```



# Back to class size and student performance

- Let's go back the **STAR** experiment data, and focus on:
  - *small* and *regular* classes,
  - *Kindergarten* grade.
- We consider the following regression model and estimate it by OLS:

$$\text{math score}_i = b_0 + b_1 \text{small}_i + e_i$$

```
library(tidyverse)

star_df = read.csv("https://www.dropbox.com/s/bf1fog8y
star_df = star_df[complete.cases(star_df), ]
star_df = star_df %>%
  filter(star %in% c("small", "regular") &
         grade == "k") %>%
  mutate(small = (star == "small"))
```

```
reg_star = lm(math ~ small, star_df)
reg_star

##
## Call:
## lm(formula = math ~ small, data = star_df)
##
## Coefficients:
## (Intercept)    smallTRUE
##        484.446      8.895
```

- What if we drew another random sample of schools from Tennessee and redid the experiment, would we find a different value for  $b_1$ ?
- We know the answer is yes, but how different is this estimate likely to be?



# Regression Inference: $b_k$ vs $\beta_k$

- $b_0, b_1$  are *point estimates* computed from our sample.
  - Just like the sample proportion  $\hat{p}$  from our pasta example!



# Regression Inference: $b_k$ vs $\beta_k$

- $b_0, b_1$  are *point estimates* computed from our sample.
  - Just like the sample proportion  $\hat{p}$  from our pasta example!
- In fact, our model's prediction...

$$\hat{y} = b_0 + b_1 x_1$$



# Regression Inference: $b_k$ vs $\beta_k$

- $b_0, b_1$  are **point estimates** computed from our sample.
  - Just like the sample proportion  $\hat{p}$  from our pasta example!
- In fact, our model's prediction...  
$$\hat{y} = b_0 + b_1 x_1$$
... is an **estimate** about an unknown, **true population line**  
$$y = \beta_0 + \beta_1 x_1$$

where  $\beta_0, \beta_1$  are the **population parameters** of interest.



# Regression Inference: $b_k$ vs $\beta_k$

- $b_0, b_1$  are **point estimates** computed from our sample.
  - Just like the sample proportion  $\hat{p}$  from our pasta example!
- In fact, our model's prediction...  
$$\hat{y} = b_0 + b_1 x_1$$
... is an **estimate** about an unknown, **true population line**  
$$y = \beta_0 + \beta_1 x_1$$

where  $\beta_0, \beta_1$  are the **population parameters** of interest.

- You will often find  $\hat{\beta}_k$  rather than  $b_k$ , both refer to sample estimate of  $\beta_k$ .



# Regression Inference: $b_k$ vs $\beta_k$

- $b_0, b_1$  are **point estimates** computed from our sample.
    - Just like the sample proportion  $\hat{p}$  from our pasta example!
  - In fact, our model's prediction...  
$$\hat{y} = b_0 + b_1 x_1$$
- ... is an **estimate** about an unknown, **true population line**  
$$y = \beta_0 + \beta_1 x_1$$

where  $\beta_0, \beta_1$  are the **population parameters** of interest.

- You will often find  $\hat{\beta}_k$  rather than  $b_k$ , both refer to sample estimate of  $\beta_k$ .
- Let's bring what we know about **confidence intervals**, **hypothesis testing** and **standard errors** to bear on those  $\hat{\beta}_k$ !



# Understanding Regression Tables

Here is our `tidy` regression:

```
library(broom)
tidy(lm(math ~ small, star_df))

## # A tibble: 2 x 5
##   term      estimate std.error statistic    p.value
##   <chr>      <dbl>     <dbl>     <dbl>      <dbl>
## 1 (Intercept) 484.      1.15     421.     0
## 2 smallTRUE    8.90     1.68      5.30  0.000000123
```

- There are 3 new columns here: `std.error`, `statistic`, `p.value`.



# Understanding Regression Tables

Here is our `tidy` regression:

```
library(broom)
tidy(lm(math ~ small, star_df))

## # A tibble: 2 x 5
##   term      estimate std.error statistic    p.value
##   <chr>     <dbl>     <dbl>     <dbl>      <dbl>
## 1 (Intercept) 484.      1.15     421.     0
## 2 smallTRUE    8.90     1.68      5.30  0.000000123
```

- There are 3 new columns here: `std.error`, `statistic`, `p.value`.

Entry	Meaning
<code>std. error</code>	Standard error of $b_k$
<code>statistic</code>	Observed test statistic associated to $H_0 : \beta_k = 0, H_A : \beta_k \neq 0$
<code>p.value</code>	p-value associated to $H_0 : \beta_k = 0, H_A : \beta_k \neq 0$



# Understanding Regression Tables

Here is our `tidy` regression:

```
library(broom)
tidy(lm(math ~ small, star_df))

## # A tibble: 2 x 5
##   term      estimate std.error statistic    p.value
##   <chr>     <dbl>     <dbl>     <dbl>      <dbl>
## 1 (Intercept) 484.      1.15     421.     0
## 2 smallTRUE    8.90     1.68      5.30  0.000000123
```

- There are 3 new columns here: `std.error`, `statistic`, `p.value`.

Entry	Meaning
<code>std. error</code>	Standard error of $b_k$
<code>statistic</code>	Observed test statistic associated to $H_0 : \beta_k = 0, H_A : \beta_k \neq 0$
<code>p.value</code>	p-value associated to $H_0 : \beta_k = 0, H_A : \beta_k \neq 0$

- Let's focus on the `small` coefficient and make sense of each entry.



# Standard Error of $b_k$

| ***Standard Error of  $b_k$ :*** Standard deviation of the sampling distribution of  $b_k$ .



# Standard Error of $b_k$

| ***Standard Error of  $b_k$ :*** Standard deviation of the sampling distribution of  $b_k$ .

Let's imagine we could redo the experiment 1,000 times on 1,000 different samples:

- We'd run 1,000 regressions and obtain 1,000 estimates of  $\beta_k$ ,  $b_k$ .



# Standard Error of $b_k$

| **Standard Error of  $b_k$ :** Standard deviation of the sampling distribution of  $b_k$ .

Let's imagine we could redo the experiment 1,000 times on 1,000 different samples:

- We'd run 1,000 regressions and obtain 1,000 estimates of  $\beta_k$ ,  $b_k$ .
- The standard error of  $b_k$  quantifies how much variation in  $b_k$  one would expect across (*an infinity of*) samples.



# Standard Error of $b_{\text{small}}$

- From the table, we get  $\hat{SE}(b_{\text{small}}) = 1.68$ 
  - Notice that we write  $\hat{SE}$  and not  $SE$  because 1.68 is an estimate of the real standard error of  $b_{\text{small}}$  we get from our sample.
  - We would love to know the real standard error  $SE$ , but we have only one sample!



# Standard Error of $b_{\text{small}}$

- From the table, we get  $\hat{\text{SE}}(b_{\text{small}}) = 1.68$ 
  - Notice that we write  $\hat{\text{SE}}$  and not  $\text{SE}$  because 1.68 is an estimate of the real standard error of  $b_{\text{small}}$  we get from our sample.
  - We would love to know the real standard error  $\text{SE}$ , but we have only one sample!
- Let's simulate the sampling distribution of  $b_{\text{small}}$  to see where it comes from.



# Task 1

07 : 00

As we did for the sampling distribution of the proportion of *green pasta*, we want to generate the bootstrap distribution of  $b_{\text{small}}$ .

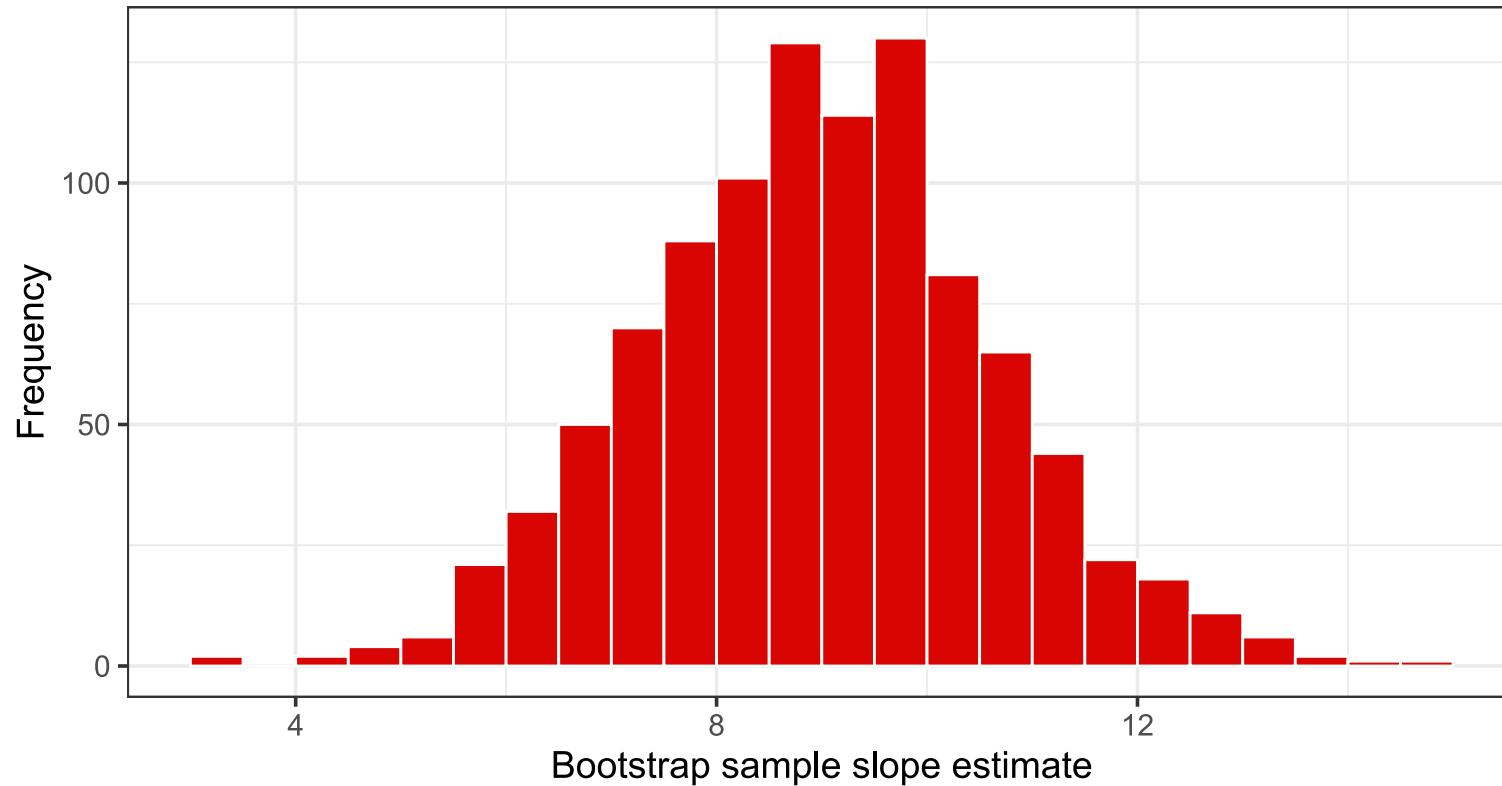
1. Copy the loading and cleaning code from slide 3 and run it.
2. Generate the bootstrap distribution of  $b_{\text{small}}$  based on 1,000 samples drawn from `star_df`.  
You can do so through the following code

```
bootstrap_distrib <- star_df %>%
  mutate(small=as.numeric(small)) %>%
  specify(response = math, explanatory = small) %>%
  generate(reps = 1000, type = "bootstrap") %>%
  calculate(stat = "slope")
```

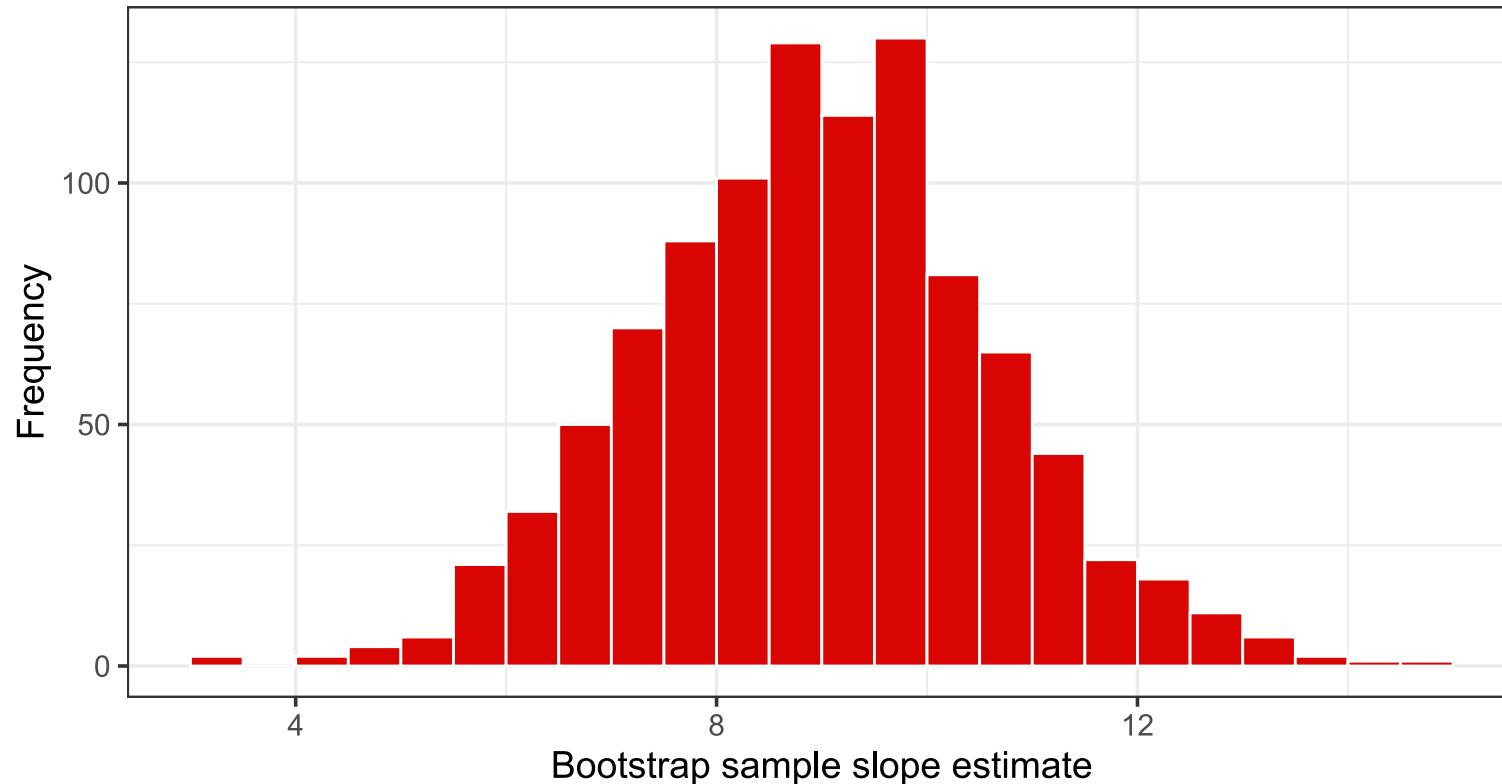
1. Plot this simulated sampling distribution and compute the mean and standard error of  $b_{\text{small}}$ .



# Bootstrap Distribution



# Bootstrap Distribution



**standard error:** 1.66 → very close to the one in the table (1.68)!

Not exactly equal, because we used bootstrapping instead of the theory approach used by R. 10 / 47



# Back to our regression results

```
library(broom)
tidy(lm(math ~ small, star_df))

## # A tibble: 2 x 5
##   term      estimate std.error statistic    p.value
##   <chr>     <dbl>     <dbl>     <dbl>      <dbl>
## 1 (Intercept) 484.      1.15     421.     0
## 2 smallTRUE    8.90      1.68      5.30  0.000000123
```

- We have made sense of the `std.error` column.
- The next two columns in our regression are `statistic` and `p.value`
- We know those terms from our previous class on hypothesis testing
- But which hypothesis test do they correspond to?



# Testing $\beta_k = 0$ vs $\beta_k \neq 0$

By default, the regression output provides the results associated with the following hypothesis test:

$$H_0 : \beta_k = 0$$
$$H_A : \beta_k \neq 0$$



# Testing $\beta_k = 0$ vs $\beta_k \neq 0$

By default, the regression output provides the results associated with the following hypothesis test:

$$H_0 : \beta_k = 0$$
$$H_A : \beta_k \neq 0$$

- It allows to statistically test if there is a true relationship between the outcome and our regressor.



# Testing $\beta_k = 0$ vs $\beta_k \neq 0$

By default, the regression output provides the results associated with the following hypothesis test:

$$H_0 : \beta_k = 0$$
$$H_A : \beta_k \neq 0$$

- It allows to statistically test if there is a true relationship between the outcome and our regressor.
- If  $H_0$  is true, there is **no** relationship between the outcome and our regressor.
  - In that case observing  $b_1 \neq 0$  was just chance.



# Testing $\beta_k = 0$ vs $\beta_k \neq 0$

By default, the regression output provides the results associated with the following hypothesis test:

$$H_0 : \beta_k = 0$$
$$H_A : \beta_k \neq 0$$

- It allows to statistically test if there is a true relationship between the outcome and our regressor.
- If  $H_0$  is true, there is **no** relationship between the outcome and our regressor.
  - In that case observing  $b_1 \neq 0$  was just chance.
- If  $H_0$  is false, then there **is** a true relationship.



# Testing $\beta_k = 0$ vs $\beta_k \neq 0$

By default, the regression output provides the results associated with the following hypothesis test:

$$H_0 : \beta_k = 0$$
$$H_A : \beta_k \neq 0$$

- It allows to statistically test if there is a true relationship between the outcome and our regressor.
- If  $H_0$  is true, there is **no** relationship between the outcome and our regressor.
  - In that case observing  $b_1 \neq 0$  was just chance.
- If  $H_0$  is false, then there **is** a true relationship.
- **Important:** This is a **two-sided** test!



# Test statistic and p-value

- As we saw in the previous lecture, to conduct such a test we need to:



# Test statistic and p-value

- As we saw in the previous lecture, to conduct such a test we need to:
  - Derive the sampling distribution of our **test statistic** (statistic) assuming  $H_0$  is true, i.e. the *null distribution*.



# Test statistic and p-value

- As we saw in the previous lecture, to conduct such a test we need to:
  - Derive the sampling distribution of our **test statistic** (statistic) assuming  $H_0$  is true, i.e. the *null distribution*.
  - Quantify how extreme the **observed test statistic** is in this hypothetic world.



# Test statistic and p-value

- As we saw in the previous lecture, to conduct such a test we need to:
  - Derive the sampling distribution of our **test statistic** (**statistic**) assuming  $H_0$  is true, i.e. the *null distribution*.
  - Quantify how extreme the **observed test statistic** is in this hypothetic world.
- Our *observed test statistic* (**statistic**) equals  $\frac{b}{\hat{SE}(b)}$ .
  - Why not just  $b$ ? We'll come back and explain this formula later.



# Test statistic and p-value

- As we saw in the previous lecture, to conduct such a test we need to:
  - Derive the sampling distribution of our **test statistic** (`statistic`) assuming  $H_0$  is true, i.e. the *null distribution*.
  - Quantify how extreme the **observed test statistic** is in this hypothetic world.
- Our *observed test statistic* (`statistic`) equals  $\frac{b}{\hat{SE}(b)}$ .
  - Why not just  $b$ ? We'll come back and explain this formula later.

```
observed_stat = reg_star$coefficients[2]/sd(bootstrap.  
round(observed_stat,2)
```

```
## smallTRUE  
##      5.36
```



# Test statistic and p-value

- As we saw in the previous lecture, to conduct such a test we need to:
  - Derive the sampling distribution of our **test statistic** (`statistic`) assuming  $H_0$  is true, i.e. the *null distribution*.
  - Quantify how extreme the **observed test statistic** is in this hypothetic world.
- Our *observed test statistic* (`statistic`) equals  $\frac{b}{\hat{SE}(b)}$ .
  - Why not just  $b$ ? We'll come back and explain this formula later.

```
observed_stat = reg_star$coefficients[2]/sd(bootstrap  
round(observed_stat,2)  
  
## smallTRUE  
##      5.36
```

- Quite close to the observed test statistic we got in the table: `statistic` = 5.3.



# Test statistic and p-value

- As we saw in the previous lecture, to conduct such a test we need to:
  - Derive the sampling distribution of our **test statistic** (`statistic`) assuming  $H_0$  is true, i.e. the *null distribution*.
  - Quantify how extreme the **observed test statistic** is in this hypothetic world.
- Our *observed test statistic* (`statistic`) equals  $\frac{b}{\hat{SE}(b)}$ .
  - Why not just  $b$ ? We'll come back and explain this formula later.

```
observed_stat = reg_star$coefficients[2]/sd(bootstrap  
round(observed_stat, 2)  
  
## smallTRUE  
##      5.36
```

- The **p-value** measures the area outside of  $\pm$  *observed test statistic* under the *null distribution*.

- Quite close to the observed test statistic we got in the table: `statistic` = 5.3.



# Test statistic and p-value

- As we saw in the previous lecture, to conduct such a test we need to:
  - Derive the sampling distribution of our **test statistic** (`statistic`) assuming  $H_0$  is true, i.e. the *null distribution*.
  - Quantify how extreme the **observed test statistic** is in this hypothetic world.
- Our *observed test statistic* (`statistic`) equals  $\frac{b}{\hat{SE}(b)}$ .
  - Why not just  $b$ ? We'll come back and explain this formula later.

```
observed_stat = reg_star$coefficients[2]/sd(bootstrap  
round(observed_stat, 2)  
  
## smallTRUE  
##      5.36
```

- The **p-value** measures the area outside of  $\pm$  *observed test statistic* under the *null distribution*.
- Finally, we check if we can reject  $H_0$  at the usual **significance levels**:  $\alpha = 0.1, 0.05, 0.01$ .

- Quite close to the observed test statistic we got in the table: `statistic` = 5.3.



# Testing $\beta_{\text{small}} = 0$ vs $\beta_{\text{small}} \neq 0$

- We will approximate the null distribution of  $\frac{b_{\text{small}}}{\hat{SE}(b_{\text{small}})}$  through a simulation exercise.



# Testing $\beta_{\text{small}} = 0$ vs $\beta_{\text{small}} \neq 0$

- We will approximate the null distribution of  $\frac{b_{\text{small}}}{\hat{SE}(b_{\text{small}})}$  through a simulation exercise.
- If there is no relationship between math score and class size, i.e.  $H_0$  is true and  $\beta_{\text{small}} = 0$ , then *reshuffling / permuting* the values of small across students should play no role.



# Testing $\beta_{\text{small}} = 0$ vs $\beta_{\text{small}} \neq 0$

- We will approximate the null distribution of  $\frac{b_{\text{small}}}{\hat{SE}(b_{\text{small}})}$  through a simulation exercise.
- If there is no relationship between math score and class size, i.e.  $H_0$  is true and  $\beta_{\text{small}} = 0$ , then *reshuffling / permuting* the values of `small` across students should play no role.
- Let's generate 1,000 permuted samples and compute  $b_{\text{small}}$  for each.

```
null_distribution <- star_df %>%
  mutate(small=as.numeric(small)) %>%
  specify(formula = math ~ small) %>%
  hypothesize(null = "independence") %>%
  generate(reps = 1000, type = "permute") %>%
  calculate(stat = "slope")
```



# Testing $\beta_{\text{small}} = 0$ vs $\beta_{\text{small}} \neq 0$

- We will approximate the null distribution of  $\frac{b_{\text{small}}}{\hat{SE}(b_{\text{small}})}$  through a simulation exercise.
- If there is no relationship between math score and class size, i.e.  $H_0$  is true and  $\beta_{\text{small}} = 0$ , then *reshuffling / permuting* the values of `small` across students should play no role.
- Let's generate 1,000 permuted samples and compute  $b_{\text{small}}$  for each.

```
null_distribution <- star_df %>%
  mutate(small=as.numeric(small)) %>%
  specify(formula = math ~ small) %>%
  hypothesize(null = "independence") %>%
  generate(reps = 1000, type = "permute") %>%
  calculate(stat = "slope")
```

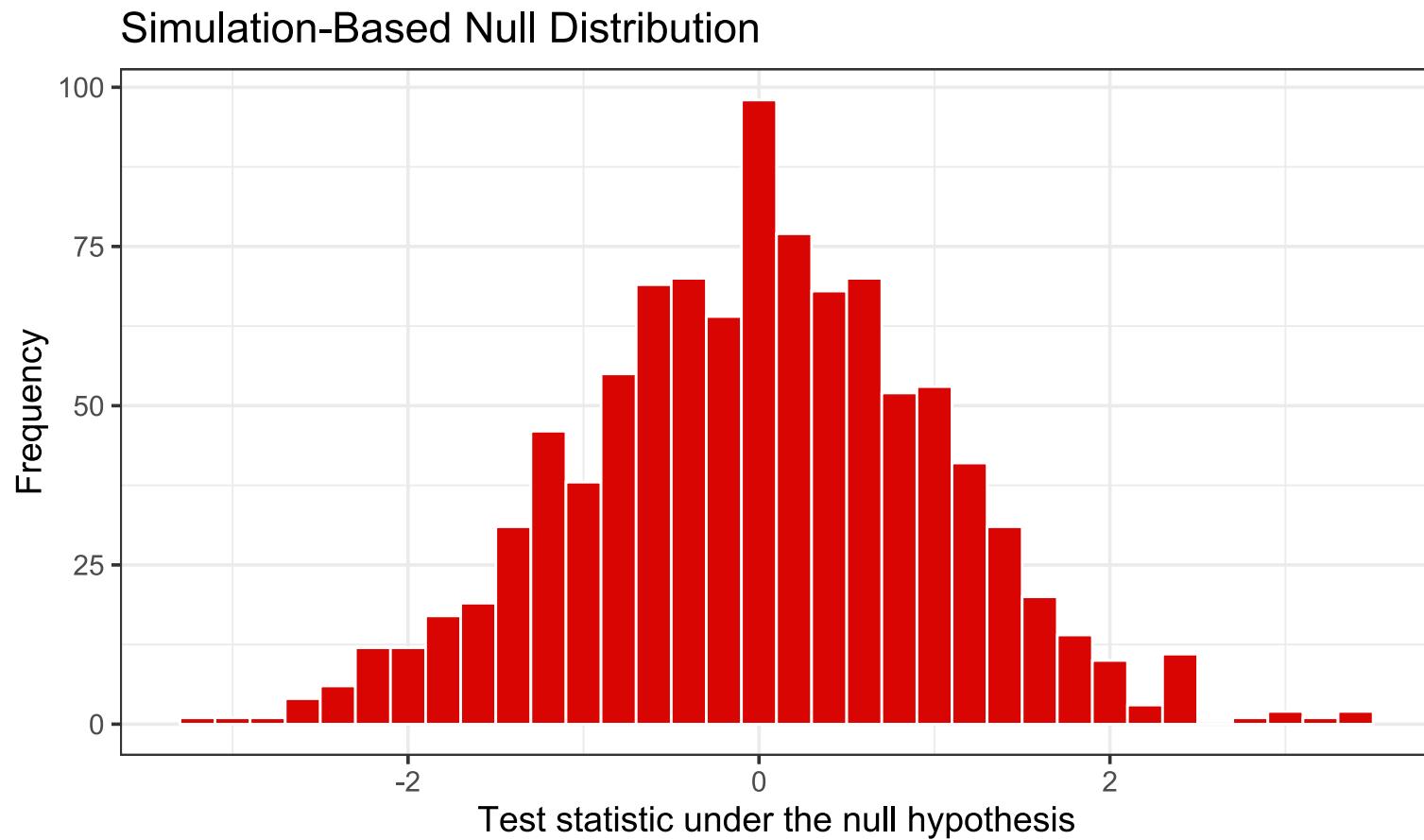
- We can compute the distribution of our test statistic  $\frac{b_{\text{small}}}{\hat{SE}(b_{\text{small}})}$  under the null:

```
null_distribution <- null_distribution %>%
  mutate(test_stat = stat/sd(bootstrap_distrib$stat))
```

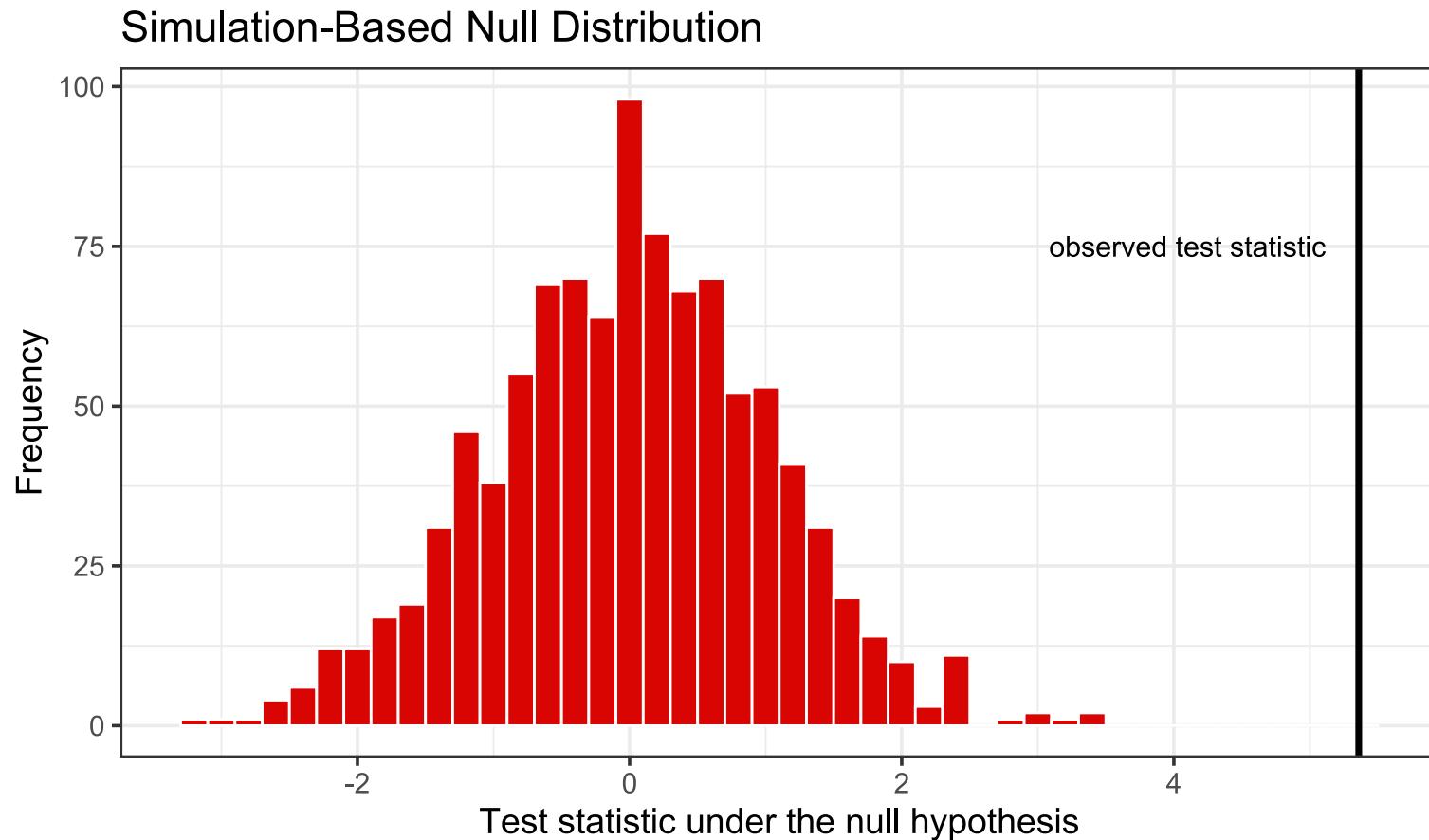
- Remember we got  $\hat{SE}(b_{\text{small}}) = 1.66$  from our bootstrap distribution.



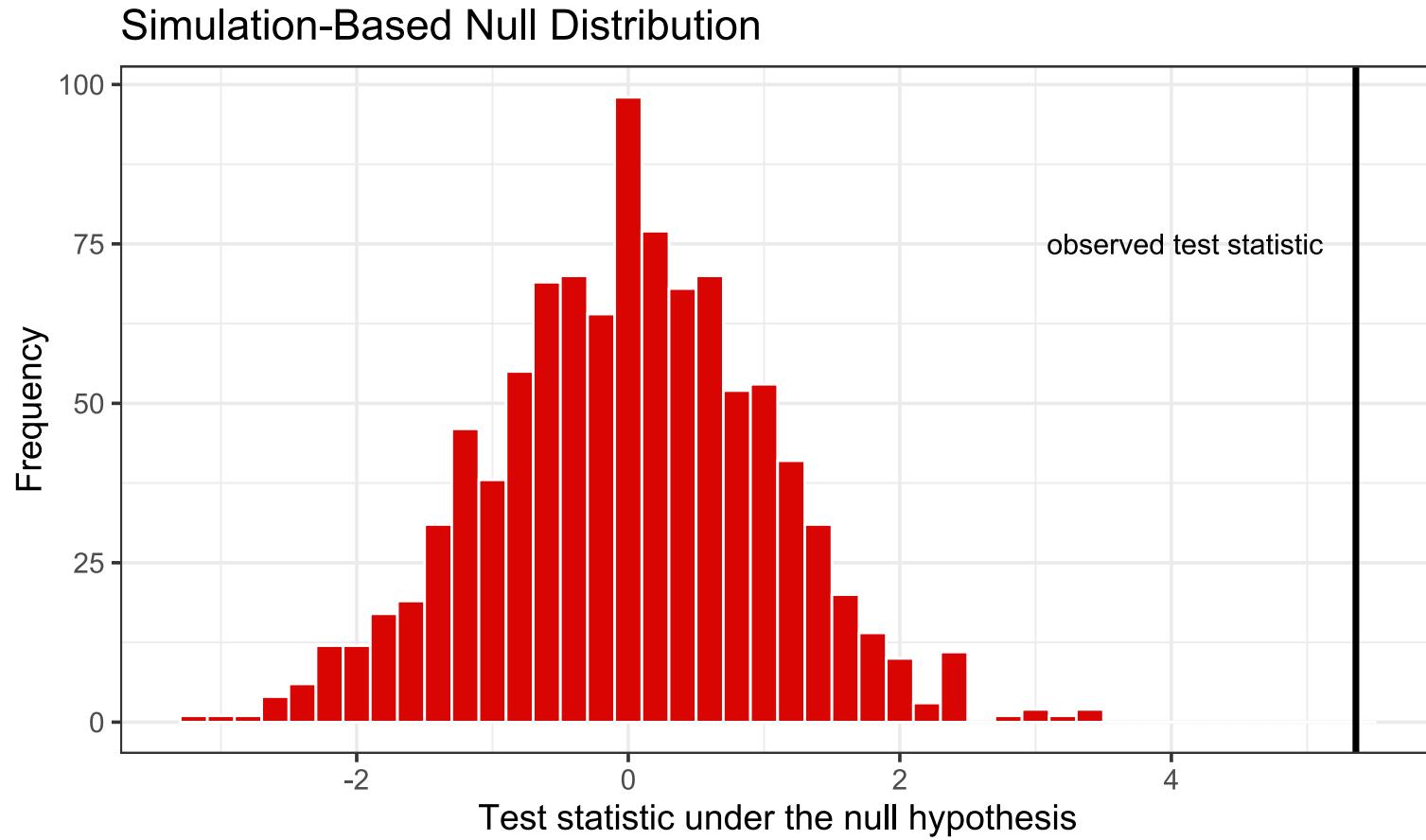
# Testing $\beta_{\text{small}} = 0$ vs $\beta_{\text{small}} \neq 0$



# Testing $\beta_{\text{small}} = 0$ vs $\beta_{\text{small}} \neq 0$



# Testing $\beta_{\text{small}} = 0$ vs $\beta_{\text{small}} \neq 0$



Very unlikely to obtain  $b_{\text{small}} = 8.8951932$  when  $H_0$  is true.

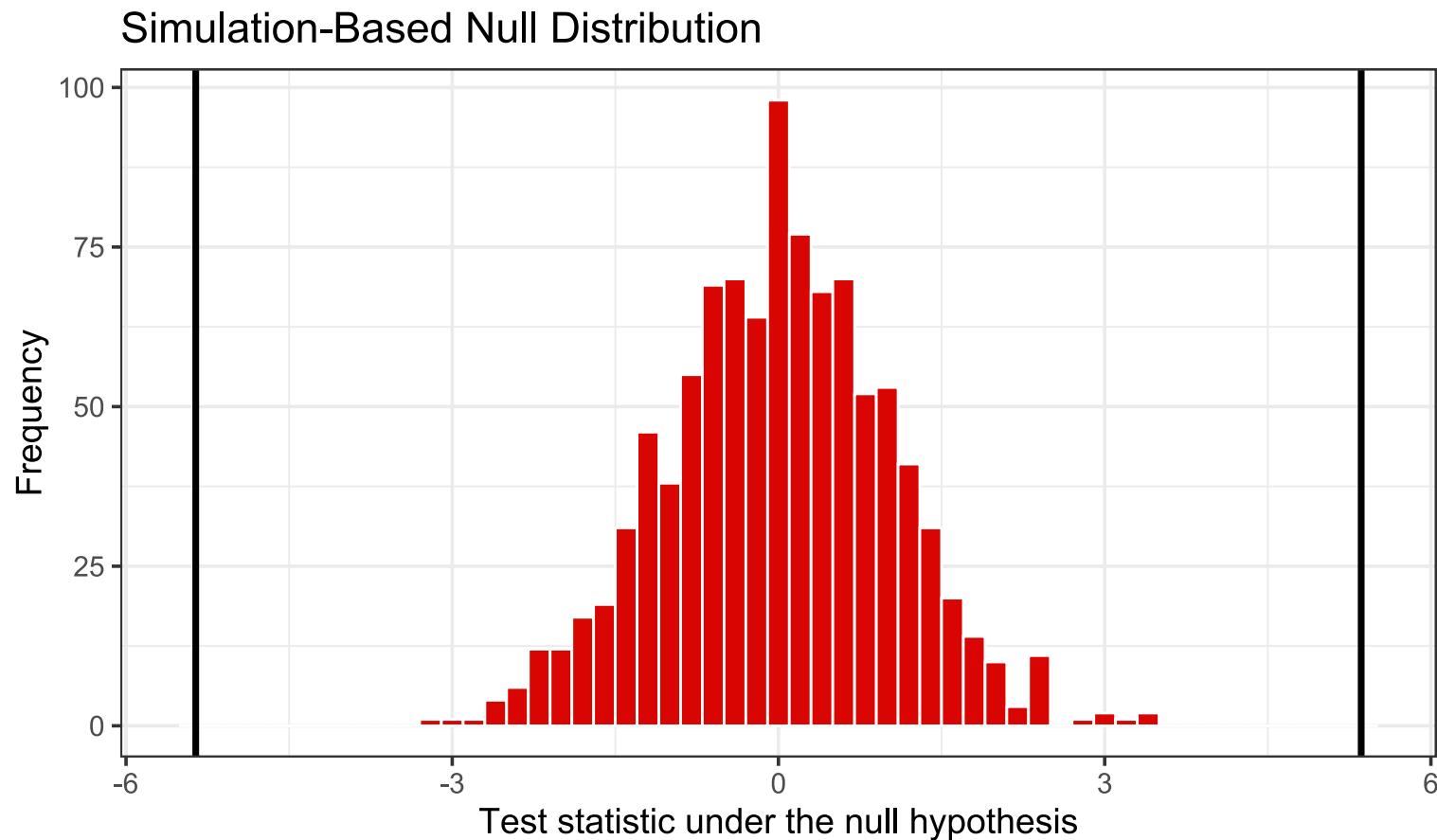


# Testing $\beta_{\text{small}} = 0$ vs $\beta_{\text{small}} \neq 0$

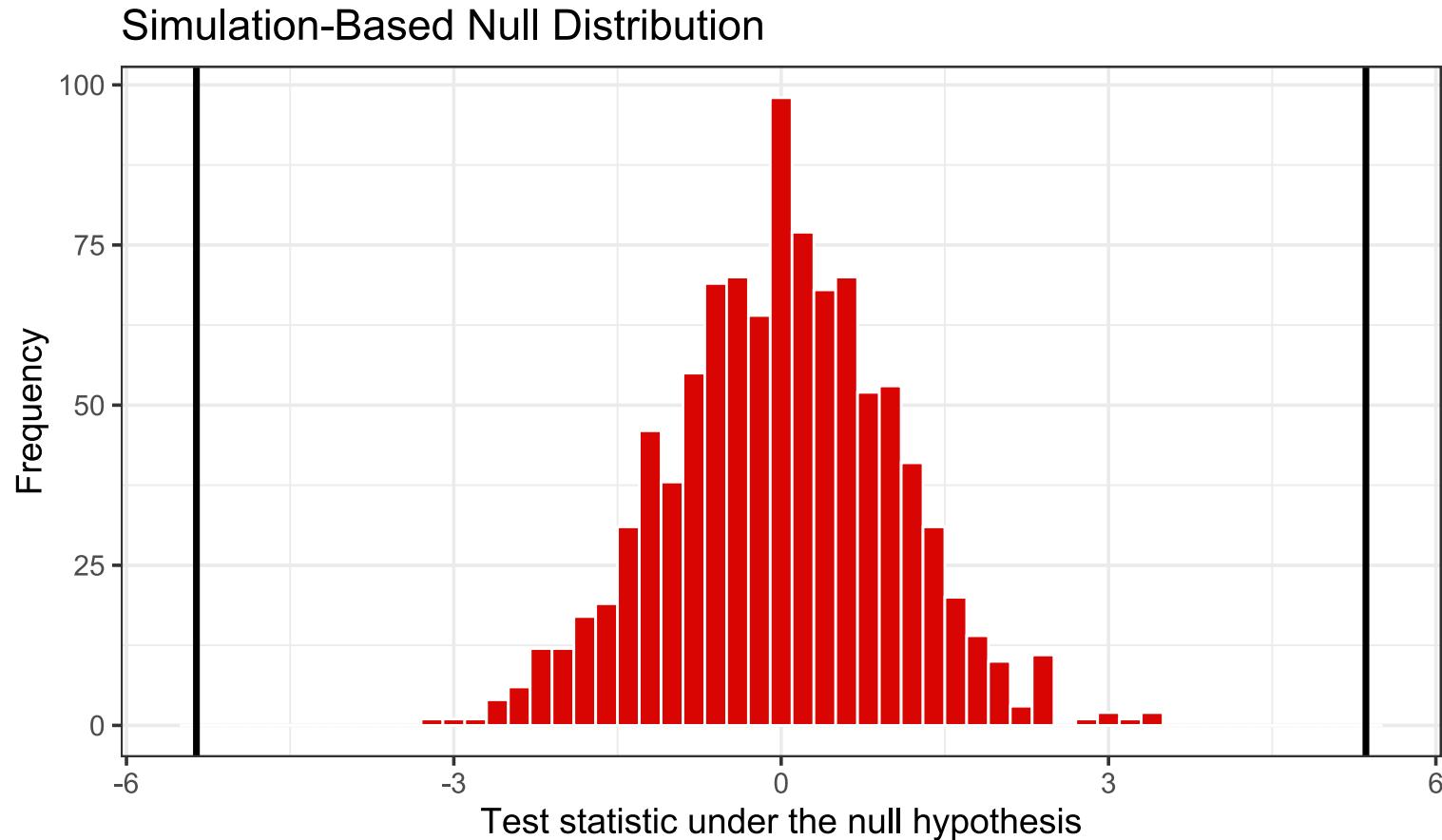
- To decide if we reject  $H_0$ , recall we are considering a **two-sided test** here: *more extreme* means inferior to -5.359 **or** superior to 5.359.



# Testing $\beta_{\text{small}} = 0$ vs $\beta_{\text{small}} \neq 0$



# Testing $\beta_{\text{small}} = 0$ vs $\beta_{\text{small}} \neq 0$



What does the p-value correspond to?

# Testing $\beta_{\text{small}} = 0$ vs $\beta_{\text{small}} \neq 0$

- To decide if we reject  $H_0$ , recall we are considering a **two-sided test** here: *more extreme* means inferior to -5.36 **or** superior to 5.36.
- Computing the *p-value* we get:

```
p_value = mean(abs(null_distribution$test_stat) >= observed_stat)
p_value
## [1] 0
```



# Testing $\beta_{\text{small}} = 0$ vs $\beta_{\text{small}} \neq 0$

- To decide if we reject  $H_0$ , recall we are considering a **two-sided test** here: *more extreme* means inferior to -5.36 **or** superior to 5.36.
- Computing the *p-value* we get:

```
p_value = mean(abs(null_distribution$test_stat) >= observed_stat)  
p_value  
## [1] 0
```

- This is the same value as in the regression table.



# Testing $\beta_{\text{small}} = 0$ vs $\beta_{\text{small}} \neq 0$

- To decide if we reject  $H_0$ , recall we are considering a **two-sided test** here: *more extreme* means inferior to -5.36 **or** superior to 5.36.
- Computing the *p-value* we get:

```
p_value = mean(abs(null_distribution$test_stat) >= observed_stat)
p_value
## [1] 0
```

- This is the same value as in the regression table.
- **Question:** Can we reject the null hypothesis at the 5% level?



# Testing $\beta_{\text{small}} = 0$ vs $\beta_{\text{small}} \neq 0$

- To decide if we reject  $H_0$ , recall we are considering a **two-sided test** here: *more extreme* means inferior to -5.36 **or** superior to 5.36.
- Computing the *p-value* we get:

```
p_value = mean(abs(null_distribution$test_stat) >= observed_stat)
p_value
## [1] 0
```

- This is the same value as in the regression table.
- **Answer:**
  - Since the *p-value* is equal to 0 it means that we would reject  $H_0$  at any significance level: the p-value would always be inferior to  $\alpha$ .
  - In other words, we can say that  $b_{\text{small}}$  is **statistically different from 0** at any significance level.
  - We also say that  $b_{\text{small}}$  is *statistically significant* (at any significance level).



# Regression Inference: Theory

# Regression Inference: Theory

- Up to now we presented simulation-based inference.



# Regression Inference: Theory

- Up to now we presented simulation-based inference.
- The values reported by statistical packages in R are instead obtained from theory.



# Regression Inference: Theory

- Up to now we presented simulation-based inference.
- The values reported by statistical packages in **R** are instead obtained from theory.
- Theoretical inference is based on **large sample approximations**.
  - One can show that sampling distributions *converge* to suitable distributions → ***Central Limit Theorem***



# Regression Inference: Theory

- Up to now we presented simulation-based inference.
- The values reported by statistical packages in R are instead obtained from theory.
- Theoretical inference is based on **large sample approximations**.
  - One can show that sampling distributions *converge* to suitable distributions → **Central Limit Theorem**
- Let's briefly look into the theory-based approach.



# Regression Inference: Theory

- Theory-based approach uses one fundamental result: the sampling distribution of the sample statistic  $\frac{b - \beta}{\hat{\text{SE}}(b)}$  converges to a **standard normal distribution** as the sample size gets larger and larger.
  - $\hat{\text{SE}}(b)$  is the sample estimate of the standard deviation of  $b$ .
  - It is also obtained through a theoretical formula (which you can find in the **book!**) but we'll leave it aside.



# Regression Inference: Theory

- Theory-based approach uses one fundamental result: the sampling distribution of the sample statistic  $\frac{b - \beta}{\hat{\text{SE}}(b)}$  converges to a **standard normal distribution** as the sample size gets larger and larger.
  - $\hat{\text{SE}}(b)$  is the sample estimate of the standard deviation of  $b$ .
  - It is also obtained through a theoretical formula (which you can find in the **book!**) but we'll leave it aside.
- A **standard normal distribution** is a *normal distribution* with *mean 0* and *standard deviation 1*.



# Regression Inference: Theory

- Theory-based approach uses one fundamental result: the sampling distribution of the sample statistic  $\frac{b - \beta}{\hat{\text{SE}}(b)}$  converges to a **standard normal distribution** as the sample size gets larger and larger.
  - $\hat{\text{SE}}(b)$  is the sample estimate of the standard deviation of  $b$ .
  - It is also obtained through a theoretical formula (which you can find in the **book!**) but we'll leave it aside.
- A **standard normal distribution** is a *normal distribution* with *mean 0* and *standard deviation 1*.
- We don't need to simulate any sampling distribution here, we derive it from theory and use it to construct confidence intervals or to conduct hypothesis tests.

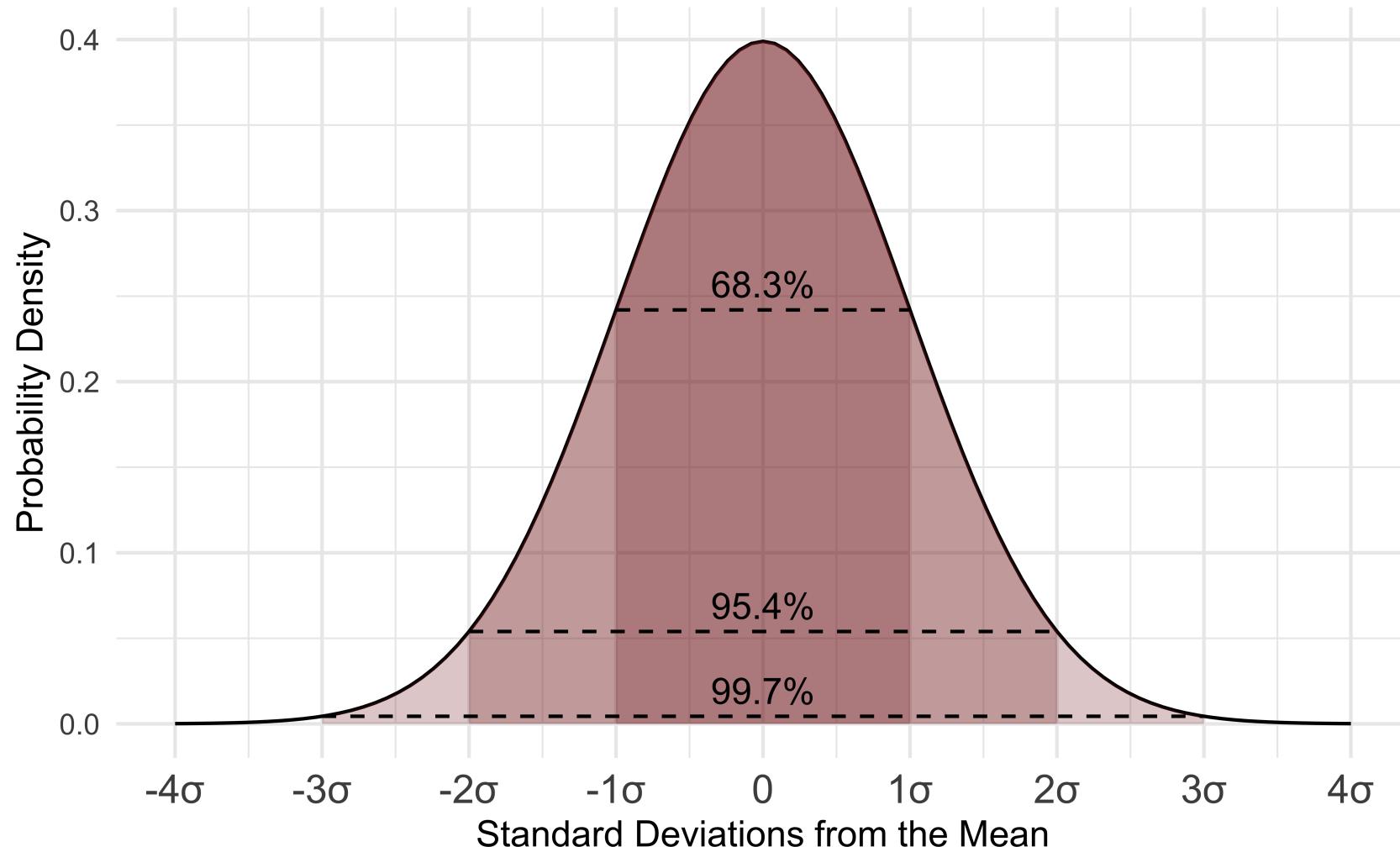


# Regression Inference: Theory

- Theory-based approach uses one fundamental result: the sampling distribution of the sample statistic  $\frac{b-\beta}{\hat{\text{SE}}(b)}$  converges to a **standard normal distribution** as the sample size gets larger and larger.
  - $\hat{\text{SE}}(b)$  is the sample estimate of the standard deviation of  $b$ .
  - It is also obtained through a theoretical formula (which you can find in the **book!**) but we'll leave it aside.
- A **standard normal distribution** is a *normal distribution* with *mean* 0 and *standard deviation* 1.
- We don't need to simulate any sampling distribution here, we derive it from theory and use it to construct confidence intervals or to conduct hypothesis tests.
- Note that if  $\frac{b-\beta}{\hat{\text{SE}}(b)}$  converges to a **standard normal distribution**, then  $b$  converges to a **normal distribution** with mean  $\beta$  and standard deviation  $\hat{\text{SE}}(b)$ .



# Normal Distribution: A Refresher



# Theory-Based Inference: Confidence Interval

- Let's take the example of a 95% confidence interval.



# Theory-Based Inference: Confidence Interval

- Let's take the example of a 95% confidence interval.
- Since the sampling distribution of  $b$  is assumed to be normally shaped, we can use the ***95% rule of thumb*** about normal distributions.



# Theory-Based Inference: Confidence Interval

- Let's take the example of a 95% confidence interval.
- Since the sampling distribution of  $b$  is assumed to be normally shaped, we can use the ***95% rule of thumb*** about normal distributions.
- We know indeed that 95% of the values of a normal distribution lie within approximately 2 standard deviations of the mean (exactly 1.96).



# Theory-Based Inference: Confidence Interval

- Let's take the example of a 95% confidence interval.
- Since the sampling distribution of  $b$  is assumed to be normally shaped, we can use the **95% rule of thumb** about normal distributions.
- We know indeed that 95% of the values of a normal distribution lie within approximately 2 standard deviations of the mean (exactly 1.96).
- So, we can compute a 95% CI for  $\beta$  as:  $\text{CI}_{95\%} = [b \pm 1.96 * \hat{\text{SE}}(b)]$



# Theory-Based Inference: Confidence Interval

- Let's take the example of a 95% confidence interval.
- Since the sampling distribution of  $b$  is assumed to be normally shaped, we can use the **95% rule of thumb** about normal distributions.
- We know indeed that 95% of the values of a normal distribution lie within approximately 2 standard deviations of the mean (exactly 1.96).
- So, we can compute a 95% CI for  $\beta$  as:  $\text{CI}_{95\%} = [b \pm 1.96 * \hat{\text{SE}}(b)]$

```
tidy(lm(math ~ small, star_df),
     conf.int = TRUE, conf.level = 0.95) %>%
  filter(term == "smallTRUE") %>%
  select(term, conf.low, conf.high)

## # A tibble: 1 x 3
##   term    conf.low conf.high
##   <chr>      <dbl>     <dbl>
## 1 smallTRUE    5.60     12.2
```



# Theory-Based Inference: Confidence Interval

- Let's take the example of a 95% confidence interval.
- Since the sampling distribution of  $b$  is assumed to be normally shaped, we can use the **95% rule of thumb** about normal distributions.
- We know indeed that 95% of the values of a normal distribution lie within approximately 2 standard deviations of the mean (exactly 1.96).
- So, we can compute a 95% CI for  $\beta$  as:  $\text{CI}_{95\%} = [b \pm 1.96 * \hat{\text{SE}}(b)]$

```
tidy(lm(math ~ small, star_df),
     conf.int = TRUE, conf.level = 0.95) %>%
  filter(term == "smallTRUE") %>%
  select(term, conf.low, conf.high)

## # A tibble: 1 x 3
##   term    conf.low conf.high
##   <chr>     <dbl>     <dbl>
## 1 smallTRUE  5.60     12.2
```

```
bootstrap_distrib %>%
  summarise(
    lower_bound = 8.895 - 1.96*sd(stat),
    upper_bound = 8.895 + 1.96*sd(stat))

## # A tibble: 1 x 2
##   lower_bound upper_bound
##       <dbl>      <dbl>
## 1      5.64     12.1
```



# Theory-Based Inference: Confidence Interval

- Let's take the example of a 95% confidence interval.
- Since the sampling distribution of  $b$  is assumed to be normally shaped, we can use the **95% rule of thumb** about normal distributions.
- We know indeed that 95% of the values of a normal distribution lie within approximately 2 standard deviations of the mean (exactly 1.96).
- So, we can compute a 95% CI for  $\beta$  as:  $\text{CI}_{95\%} = [b \pm 1.96 * \hat{\text{SE}}(b)]$

```
tidy(lm(math ~ small, star_df),
     conf.int = TRUE, conf.level = 0.95) %>%
  filter(term == "smallTRUE") %>%
  select(term, conf.low, conf.high)

## # A tibble: 1 x 3
##   term    conf.low conf.high
##   <chr>     <dbl>     <dbl>
## 1 smallTRUE  5.60     12.2
```

```
bootstrap_distrib %>%
  summarise(
    lower_bound = 8.895 - 1.96*sd(stat),
    upper_bound = 8.895 + 1.96*sd(stat))

## # A tibble: 1 x 2
##   lower_bound upper_bound
##       <dbl>      <dbl>
## 1      5.64     12.1
```

- This can easily be generalized to any confidence level by taking the appropriate quantile of the normal distribution.



## Task 2

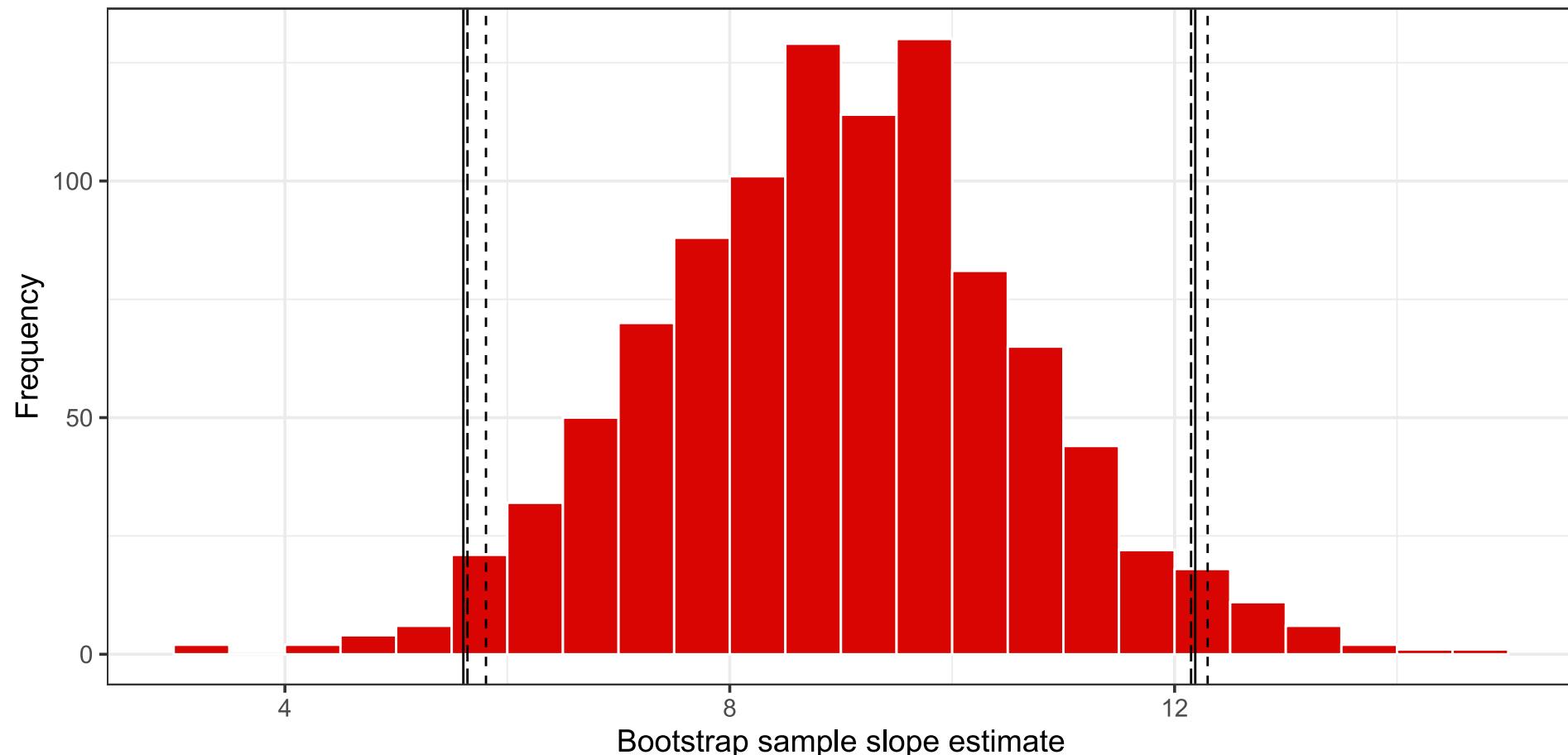
05 : 00

1. Using the bootstrap distribution you generated in Task 1, compute the 95% confidence interval using the *percentile method*.
2. How similar is it to the confidence intervals obtained in the previous slide?



# Confidence Intervals: Visually

95% confidence interval computed with different methods  
percentile (dashed), standard error (longdashed) and theory (solid)



# Theory-Based Inference: Hypothesis Testing

- Theory tells us that  $\frac{b - \beta_k}{\hat{\text{SE}}(b)}$  converges to a standard normal distribution
- As we already mentioned, the default test that is conducted by any statistical software is:

$$H_0 : \beta_k = 0$$

$$H_A : \beta_k \neq 0$$



# Theory-Based Inference: Hypothesis Testing

- Theory tells us that  $\frac{b - \beta_k}{\hat{\text{SE}}(b)}$  converges to a standard normal distribution
- As we already mentioned, the default test that is conducted by any statistical software is:

$$H_0 : \beta_k = 0$$

$$H_A : \beta_k \neq 0$$

- So, **under the null hypothesis**,  $\beta_k = 0$ , and we get from theory that the sampling distribution of  $\frac{b}{\hat{\text{SE}}(b)}$  will be a standard normal distribution.



# Theory-Based Inference: Hypothesis Testing

- Theory tells us that  $\frac{b - \beta_k}{\hat{\text{SE}}(b)}$  converges to a standard normal distribution
- As we already mentioned, the default test that is conducted by any statistical software is:

$$H_0 : \beta_k = 0$$

$$H_A : \beta_k \neq 0$$

- So, **under the null hypothesis**,  $\beta_k = 0$ , and we get from theory that the sampling distribution of  $\frac{b}{\hat{\text{SE}}(b)}$  will be a standard normal distribution.
- So the *standard normal distribution* is the **null distribution** of our test statistic.



# Theory-Based Inference: Hypothesis Testing

- Theory tells us that  $\frac{b - \beta_k}{\hat{\text{SE}}(b)}$  converges to a standard normal distribution
- As we already mentioned, the default test that is conducted by any statistical software is:

$$H_0 : \beta_k = 0$$

$$H_A : \beta_k \neq 0$$

- So, **under the null hypothesis**,  $\beta_k = 0$ , and we get from theory that the sampling distribution of  $\frac{b}{\hat{\text{SE}}(b)}$  will be a standard normal distribution.
- So the *standard normal distribution* is the **null distribution** of our test statistic.
- The **p-value** associated to our test is then equal to the area of the *standard normal distribution* outside  $\pm$  the observed value of  $\frac{b}{\hat{\text{SE}}(b)}$ .



# Theory-Based Inference: Hypothesis Testing

- Theory tells us that  $\frac{b - \beta_k}{\hat{\text{SE}}(b)}$  converges to a standard normal distribution
- As we already mentioned, the default test that is conducted by any statistical software is:

$$H_0 : \beta_k = 0$$

$$H_A : \beta_k \neq 0$$

- So, **under the null hypothesis**,  $\beta_k = 0$ , and we get from theory that the sampling distribution of  $\frac{b}{\hat{\text{SE}}(b)}$  will be a standard normal distribution.
- So the *standard normal distribution* is the **null distribution** of our test statistic.
- The **p-value** associated to our test is then equal to the area of the *standard normal distribution* outside  $\pm$  the observed value of  $\frac{b}{\hat{\text{SE}}(b)}$ .
- Common rule of thumb: if the *estimate* is **twice the size of the standard error**, then it is significant at the 5% level. Why?



# Formatting a regression table

- Now that we have learned about all components of a regression table, let's finally learn how to create and read one!

```
reg_simple_math <- lm(math ~ small, data=star_df)
reg_gender_math <- lm(math ~ small + gender , data=star_df)
reg_simple_read <- lm(read ~ small, data=star_df)
reg_gender_read <- lm(read ~ small + gender , data=star_df)

export_summs(reg_simple_math, reg_gender_math, reg_simple_read, reg_gender_read,
             model.names = c("Math score", "Math Score",
                            "Reading score", "Reading score"),
             coefs=c("Intercept" = "(Intercept)",
                    "Small class" = "smallTRUE",
                    "Male gender" = "gendermale"))
```



# Formatting a regression table

	Math score	Math Score	Reading score	Reading score
Intercept	484.45 *** (1.15)	488.85 *** (1.43)	435.76 *** (0.75)	439.62 *** (0.93)
Small class	8.90 *** (1.68)	8.94 *** (1.67)	5.37 *** (1.09)	5.41 *** (1.09)
Male gender		-8.56 *** (1.67)		-7.49 *** (1.09)
N	3359	3359	3359	3359
R2	0.01	0.02	0.01	0.02

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



# Reading a regression table

	Math score	Math Score	Reading score	Reading score
Intercept	484.45 *** (1.15)	488.85 *** (1.43)	435.76 *** (0.75)	439.62 *** (0.93)
Small class	8.90 *** (1.68)	8.94 *** (1.67)	5.37 *** (1.09)	5.41 *** (1.09)
Male gender		-8.56 *** (1.67)		-7.49 *** (1.09)
N	3359	3359	3359	3359
R2	0.01	0.02	0.01	0.02

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

- Each column corresponds to a regression. For the first regression we have:
  - the **name of the outcome variable** in blue
  - the **estimated coefficient associated to being in a small class**  $\hat{\beta}_{\text{small}}$  in green
  - its **estimated standard error** in yellow
  - the **number of observations** in purple
  - the **R-squared** in red
  - interpretation of the stars at the bottom



# Classical Regression Model

# Classical Regression Model

- Whether the inference is made from theory or simulations, some assumptions have to be met for this inference to be valid.
- The set of assumptions needed defines the *Classical Regression Model* (CRM).

# Classical Regression Model

- Whether the inference is made from theory or simulations, some assumptions have to be met for this inference to be valid.
- The set of assumptions needed defines the *Classical Regression Model* (CRM).
- Before delving into these assumptions, let's see the small but important modifications we apply to our model (back to *lecture SLR*):

# Classical Regression Model

- Whether the inference is made from theory or simulations, some assumptions have to be met for this inference to be valid.
- The set of assumptions needed defines the *Classical Regression Model* (CRM).
- Before delving into these assumptions, let's see the small but important modifications we apply to our model (back to *lecture SLR*):
  - We already mentioned the distinction between the sample estimate  $b_k$  (or  $\hat{\beta}_k$ ) and the population parameter  $\beta_k$ .

# Classical Regression Model

- Whether the inference is made from theory or simulations, some assumptions have to be met for this inference to be valid.
- The set of assumptions needed defines the *Classical Regression Model* (CRM).
- Before delving into these assumptions, let's see the small but important modifications we apply to our model (back to *lecture SLR*):
  - We already mentioned the distinction between the sample estimate  $b_k$  (or  $\hat{\beta}_k$ ) and the population parameter  $\beta_k$ .
  - In the same way, we distinguish  $e$ , the sample error (*residual*), from  $\varepsilon$ , the error term from the true population model:

$$y_i = \beta_0 + \beta_1 x_{1,i} + \dots + \beta_k x_{k,i} + \varepsilon_i$$

# Classical Regression Model

- Whether the inference is made from theory or simulations, some assumptions have to be met for this inference to be valid.
- The set of assumptions needed defines the *Classical Regression Model* (CRM).
- Before delving into these assumptions, let's see the small but important modifications we apply to our model (back to *lecture SLR*):
  - We already mentioned the distinction between the sample estimate  $b_k$  (or  $\hat{\beta}_k$ ) and the population parameter  $\beta_k$ .
  - In the same way, we distinguish  $e$ , the sample error (*residual*), from  $\varepsilon$ , the error term from the true population model:

$$y_i = \beta_0 + \beta_1 x_{1,i} + \dots + \beta_k x_{k,i} + \varepsilon_i$$

- The classical regression model applies to **correctly specified linear regressions**: the model needs to be linear in parameters, include all relevant variables, and variables cannot be collinear.

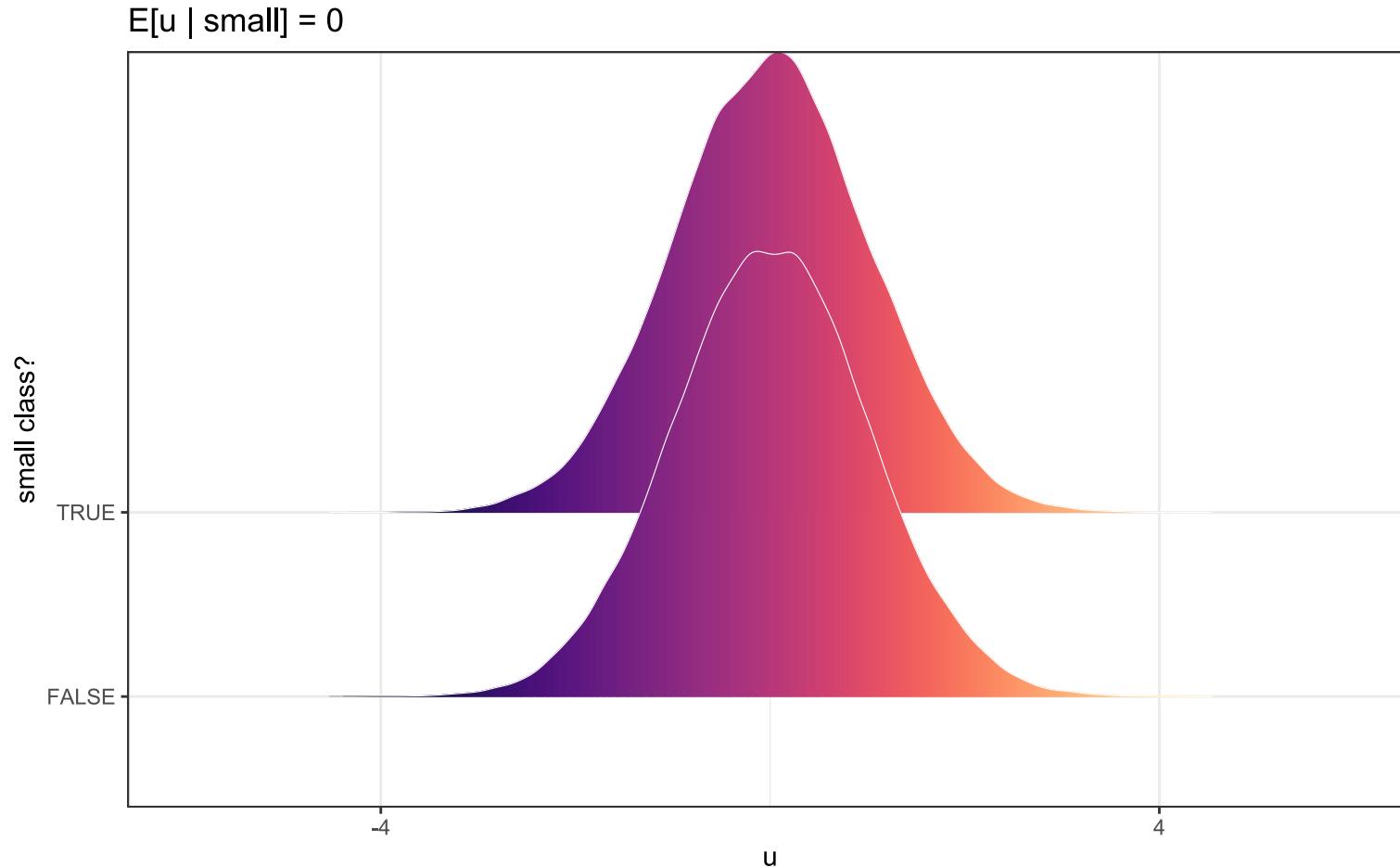
# CRM Assumptions

1. **Mean Independence:** the mean of the residuals conditional on  $x$  should be zero,  $E[\varepsilon|x] = 0$ . Notice that this also means that  $Cov(\varepsilon, x) = 0$ , i.e. that the errors and our explanatory variable(s) should be *uncorrelated*.

# CRM Assumptions

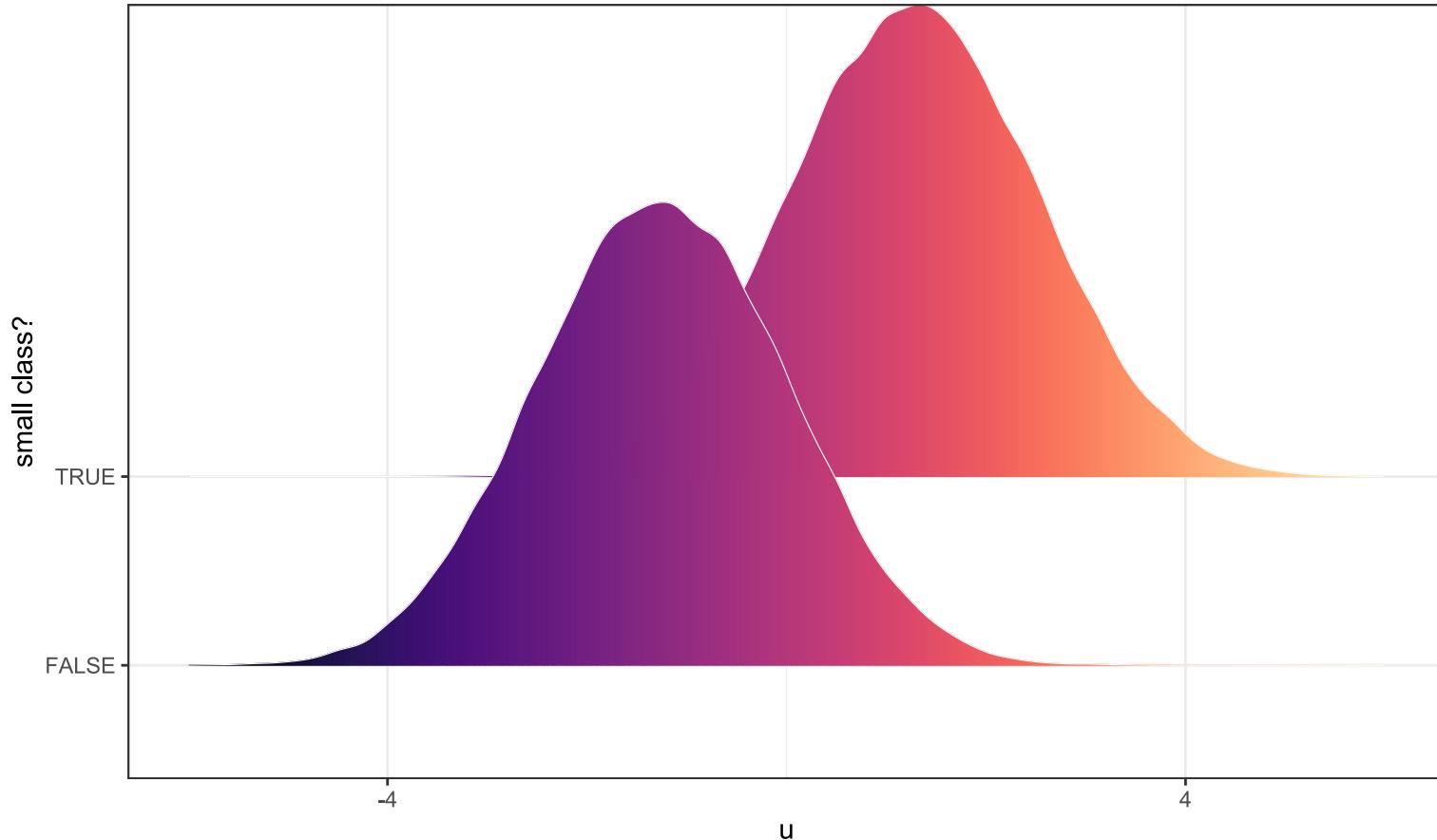
1. **Mean Independence:** the mean of the residuals conditional on  $x$  should be zero,  $E[\varepsilon|x] = 0$ . Notice that this also means that  $Cov(\varepsilon, x) = 0$ , i.e. that the errors and our explanatory variable(s) should be *uncorrelated*.
  - Violating this assumption will lead to **biased** estimates of  $\beta_k$ .

# Mean Independence of Error: $E[u | \text{small}] = ?$



# Mean Independence of Error: $E[u | \text{small}] = ?$

$E[u | \text{small}] \neq E[u | \text{not small}] \neq 0$



# Exogeneity Assumption

The CRM assumption #1 is also known as the (strict) **exogeneity assumption**.

- When this assumption is violated our estimate  $b$  will be a **biased** estimate of  $\beta$ , i.e.  
 $\mathbb{E}[b] \neq \beta$

# Exogeneity Assumption

The CRM assumption #1 is also known as the (strict) **exogeneity assumption**.

- When this assumption is violated our estimate  $b$  will be a **biased** estimate of  $\beta$ , i.e.  
 $\mathbb{E}[b] \neq \beta$
- For example, imagine you are interested in the effect of education on wage

$$\text{wage}_i = \beta_0 + \beta_1 \text{education}_i + \varepsilon_i$$

- Under the exogeneity assumption  $\beta_1$  denotes the causal effect of education in the population.

# Exogeneity Assumption

The CRM assumption #1 is also known as the (strict) **exogeneity assumption**.

- When this assumption is violated our estimate  $b$  will be a **biased** estimate of  $\beta$ , i.e.  
 $\mathbb{E}[b] \neq \beta$
- For example, imagine you are interested in the effect of education on wage

$$\text{wage}_i = \beta_0 + \beta_1 \text{education}_i + \varepsilon_i$$

- Under the exogeneity assumption  $\beta_1$  denotes the causal effect of education in the population.
- Suppose there is *unobserved* ability  $a_i$ .
  - High ability means higher wage.
  - It *also* means school is easier, and so  $i$  selects into more schooling.

# Exogeneity Assumption

- Given ability is *unobserved*,  $a_i$  goes into the error  $\varepsilon_i$

# Exogeneity Assumption

- Given ability is *unobserved*,  $a_i$  goes into the error  $\varepsilon_i$
- Our *ceteris paribus* assumption (all else equal) does not hold.

# Exogeneity Assumption

- Given ability is *unobserved*,  $a_i$  goes into the error  $\varepsilon_i$
- Our *ceteris paribus* assumption (all else equal) does not hold.
- Then regressing the wage on education we will attribute to **education** part of the effect on wages that is actually *caused* by ability  $a_i$ !

# Exogeneity Assumption

- Given ability is *unobserved*,  $a_i$  goes into the error  $\varepsilon_i$
- Our *ceteris paribus* assumption (all else equal) does not hold.
- Then regressing the wage on education we will attribute to **education** part of the effect on wages that is actually *caused* by ability  $a_i$ !
  - Remember the formula of the **omitted variable bias**:

$$\text{OVB} = \text{multiple regression coefficient on omitted variable} \times \frac{\text{Cov}(x, z)}{\text{Var}(x)}$$

# Exogeneity Assumption

- Given ability is *unobserved*,  $a_i$  goes into the error  $\varepsilon_i$
- Our *ceteris paribus* assumption (all else equal) does not hold.
- Then regressing the wage on education we will attribute to **education** part of the effect on wages that is actually *caused* by ability  $a_i$ !
  - Remember the formula of the **omitted variable bias**:

$$\text{OVB} = \text{multiple regression coefficient on omitted variable} \times \frac{\text{Cov}(x, z)}{\text{Var}(x)}$$

- Thus, we have:

$$\mathbb{E}(b_1) = \beta_1 + OVB > \beta_1$$

- *Interpretation*: taking repeated sample from the population and computing  $b_1$  each time, we would **systematically overestimate** the effect of education on wage.

# CRM Assumptions

1. **Mean Independence:** the mean of the residuals conditional on  $x$  should be zero,  $E[\varepsilon|x] = 0$ . Notice that this also means that  $\text{Cov}(\varepsilon, x) = 0$ , i.e. that the errors and our explanatory variable(s) should be *uncorrelated*.

- Violating this assumption will lead to **biased** estimates of  $\beta_k$ .

# CRM Assumptions

1. **Mean Independence:** the mean of the residuals conditional on  $x$  should be zero,  $E[\varepsilon|x] = 0$ . Notice that this also means that  $\text{Cov}(\varepsilon, x) = 0$ , i.e. that the errors and our explanatory variable(s) should be *uncorrelated*.

- Violating this assumption will lead to **biased** estimates of  $\beta_k$ .

2. **Independently and identically distributed:** the data are drawn from a **random sample** of size  $n$ : observation  $(x_i, y_i)$  comes from the exact same distribution, and is **independent** of observation  $(x_j, y_j)$ , for all  $i \neq j$ .

# CRM Assumptions

1. **Mean Independence:** the mean of the residuals conditional on  $x$  should be zero,  $E[\varepsilon|x] = 0$ . Notice that this also means that  $\text{Cov}(\varepsilon, x) = 0$ , i.e. that the errors and our explanatory variable(s) should be *uncorrelated*.

- Violating this assumption will lead to **biased** estimates of  $\beta_k$ .

2. **Independently and identically distributed:** the data are drawn from a **random sample** of size  $n$ : observation  $(x_i, y_i)$  comes from the exact same distribution, and is **independent** of observation  $(x_j, y_j)$ , for all  $i \neq j$ .

- Violating this assumption would make your sample less representative of the underlying population. It will lead to **biased** estimates of  $\beta_k$ .

# CRM Assumptions

3. ***Homoskedasticity***: the variance of the error term  $\varepsilon$  is the same for each value of  $x$ :

$$Var(\varepsilon|x) = \sigma^2.$$

# CRM Assumptions

3. **Homoskedasticity:** the variance of the error term  $\varepsilon$  is the same for each value of  $x$ :

$$Var(\varepsilon|x) = \sigma^2.$$

- If this assumption is violated, you can still obtain unbiased estimates of  $\beta_k$ . However your estimate of  $\hat{SE}(b_k)$  will be biased, which will affect your test statistic and p-value.

# CRM Assumptions

3. **Homoskedasticity:** the variance of the error term  $\varepsilon$  is the same for each value of  $x$ :

$$Var(\varepsilon|x) = \sigma^2.$$

- If this assumption is violated, you can still obtain unbiased estimates of  $\beta_k$ . However your estimate of  $\hat{SE}(b_k)$  will be biased, which will affect your test statistic and p-value.

4. **Normally distributed errors:** the error term is normally distributed, i.e.  $\varepsilon \sim \mathcal{N}(0, \sigma^2)$

# CRM Assumptions

3. **Homoskedasticity:** the variance of the error term  $\varepsilon$  is the same for each value of  $x$ :

$$Var(\varepsilon|x) = \sigma^2.$$

- If this assumption is violated, you can still obtain unbiased estimates of  $\beta_k$ . However your estimate of  $\hat{SE}(b_k)$  will be biased, which will affect your test statistic and p-value.

4. **Normally distributed errors:** the error term is normally distributed, i.e.  $\varepsilon \sim \mathcal{N}(0, \sigma^2)$

- Not strictly necessary, but makes inference possible even with small sample sizes.

# CRM Assumptions

3. **Homoskedasticity:** the variance of the error term  $\varepsilon$  is the same for each value of  $x$ :

$$Var(\varepsilon|x) = \sigma^2.$$

- If this assumption is violated, you can still obtain unbiased estimates of  $\beta_k$ . However your estimate of  $\hat{SE}(b_k)$  will be biased, which will affect your test statistic and p-value.

4. **Normally distributed errors:** the error term is normally distributed, i.e.  $\varepsilon \sim \mathcal{N}(0, \sigma^2)$

- Not strictly necessary, but makes inference possible even with small sample sizes.

☞ Takeaway: **if assumptions violated, inference is invalid!**

# Task 3.1

10 : 00

Let's go back to our question of returns to education and gender.

1. Load the data `CPS1985` from the `AER` package and look back at the `help` to get the definition of each variable: `?CPS1985`
2. Create the `log_wage` variable equal to the log of `wage`.
3. Regress `log_wage` on `gender` and `education`, and save it as `reg1`.
  - Interpret each coefficient.
  - Are the coefficients statistically significant? At which significance level?
4. Regress the `log_wage` on `gender`, `education` and their interaction `gender*education`, save it as `reg2`.
  - How do you interpret the coefficient associated to *female \* education*?
  - Can we reject the nullity of this coefficient at the 5% level? At 10%?

## Task 3.2

10:00

1. Produce a scatterplot of the relationship between the log wage and the level of education.
2. Add the *regression line* with `geom_smooth`. What does this line represents?
3. Let's illustrate what the shaded area stands for.
  1. Draw one bootstrap sample from our `cps` data.
  2. Regress the `log_wage` on `gender`, `education` and their interaction `gender*education`, save it as `reg_bootstrap`.
  3. From `reg_bootstrap` extract and save the value of the intercept for men as `intercept_men_bootstrap` and the value of the slope for men as `slope_men_bootstrap`. Do the same for women.
  4. Add both predicted lines from this bootstrap sample to the previous plot (*Hint:* use `geom_abline (x2)`)

# Illustrating Uncertainty

Let's repeat the procedure you just made  
100 times!

```
library(AER)
data("CPS1985")
cps = CPS1985 %>% mutate(log_wage = log(wage))

set.seed(1)
bootstrap_sample = cps %>%
  rep_sample_n(size = nrow(cps), reps = 100, replace = TRUE)

ggplot(data=cps,aes(y = log_wage, x = education, colour = gender))
  geom_point(size = 1, alpha = 0.7) +
  geom_smooth(method = "lm", alpha = 2) +
  geom_smooth(data=bootstrap_sample,
              size = 0.2,
              aes(y = log_wage, x = education, group = rep),
              method = "lm", se = FALSE) +
  facet_wrap(~gender) +
  scale_colour_manual(values = c("darkblue", "darkred"))
  labs(x = "Education", y = "Log wage") +
  guides(colour=FALSE) +
  theme_bw(base_size = 20)
```

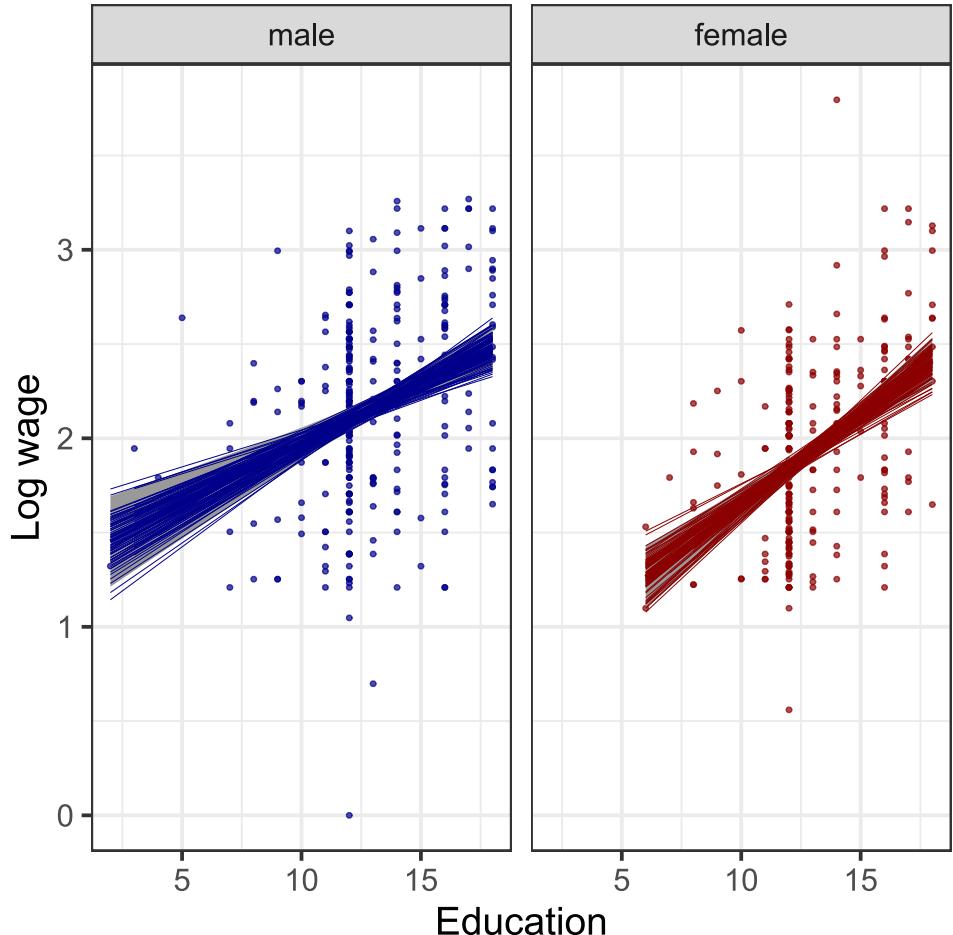
# Illustrating Uncertainty

Let's repeat the procedure you just made 100 times!

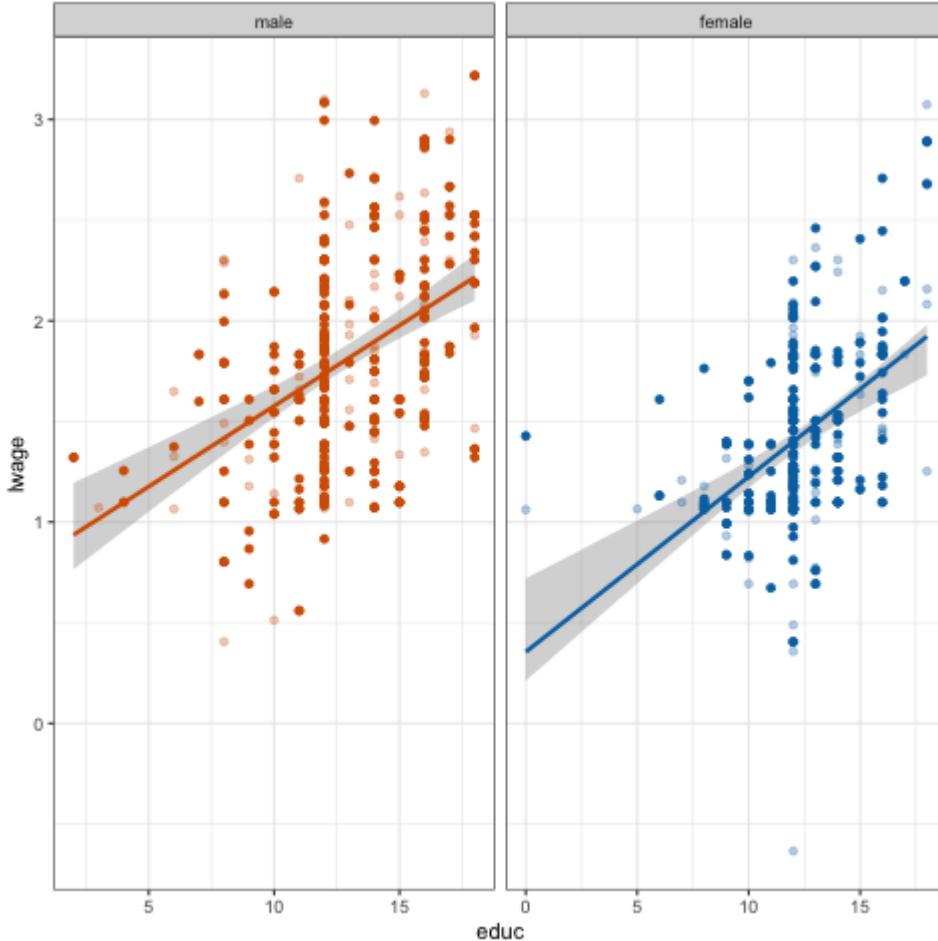
```
library(AER)
data("CPS1985")
cps = CPS1985 %>% mutate(log_wage = log(wage))

set.seed(1)
bootstrap_sample = cps %>%
  rep_sample_n(size = nrow(cps), reps = 100, replace = TRUE)

ggplot(data=cps,aes(y = log_wage, x = education, colour = gender)) +
  geom_point(size = 1, alpha = 0.7) +
  geom_smooth(method = "lm", alpha = 2) +
  geom_smooth(data=bootstrap_sample,
              size = 0.2,
              aes(y = log_wage, x = education, group = rep),
              method = "lm", se = FALSE) +
  facet_wrap(~gender) +
  scale_colour_manual(values = c("darkblue", "darkred"))
  labs(x = "Education", y = "Log wage") +
  guides(colour=FALSE) +
  theme_bw(base_size = 20)
```



# Illustrating Uncertainty



Even better : `ungeviz` and `ganimate` bring you moving lines!

- We took 20 bootstrap samples from our data
- You can see how different data points are included in each bootstrap sample.
- Those different points imply different regression lines.
- On average, 95% of these lines should fall into the shaded area.
- You should remember those moving lines when looking at the shaded area!

# On the way to causality

- How to manage data? Read it, tidy it, visualise it!
- How to summarise relationships between variables? Simple and multiple linear regression, non-linear regressions, interactions...
- What is causality?
- What if we don't observe an entire population? Sampling!
- Are our findings just due to randomness?** Confidence intervals and hypothesis testing, regression inference.
- How to find exogeneity in practice?

# THANKS

To the amazing **moderndive** team!

Big Thanks  to **ungeviz** and  **ganimate** for their awesome packages!

**SEE YOU NEXT WEEK!**

---

 florian.oswald@sciencespo.fr

 Slides

 Book

 @ScPoEcon

 @ScPoEcon

---