Inference

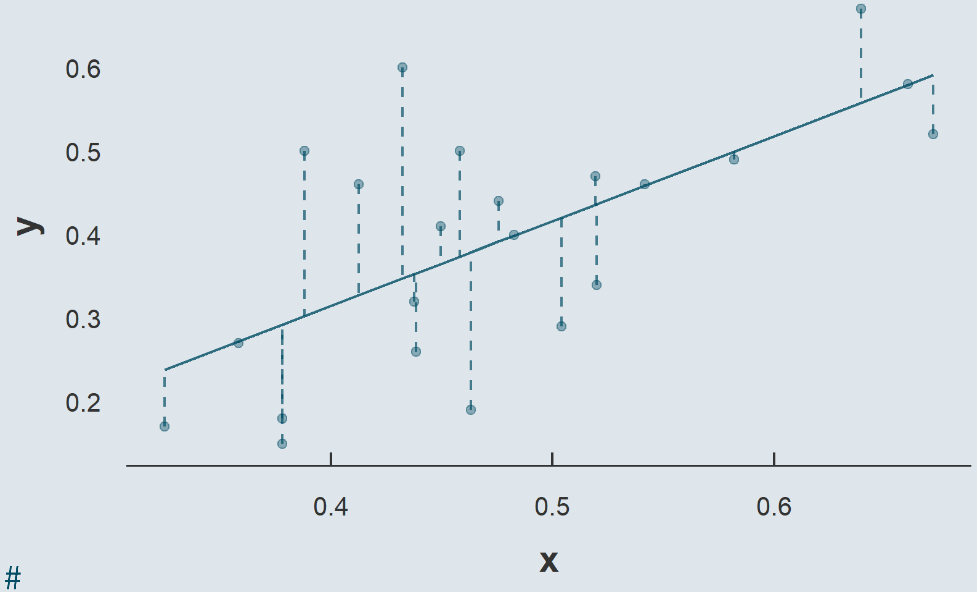
Lecture 10

Louis SIRUGUE

CPES 2 - Fall 2022

Quick reminder

1. Regression




```
##
## Call:
## lm(formula = y ~ x, data = data)
##
## Coefficients:
## (Intercept)            x
##    -0.09129      1.01546
```

- This can be expressed with the **regression equation:**

$$y_i = \hat{\alpha} + \hat{\beta}x_i + \hat{\varepsilon}_i$$

- Where $\hat{\alpha}$ is the **intercept** and $\hat{\beta}$ the **slope** of the **line** $\hat{y}_i = \hat{\alpha} + \hat{\beta}x_i$, and $\hat{\varepsilon}_i$ the **distances** between the points and the line

$$\hat{\beta} = \frac{\text{Cov}(x_i, y_i)}{\text{Var}(x_i)}$$

$$\hat{\alpha} = \bar{y} - \hat{\beta} \times \bar{x}$$

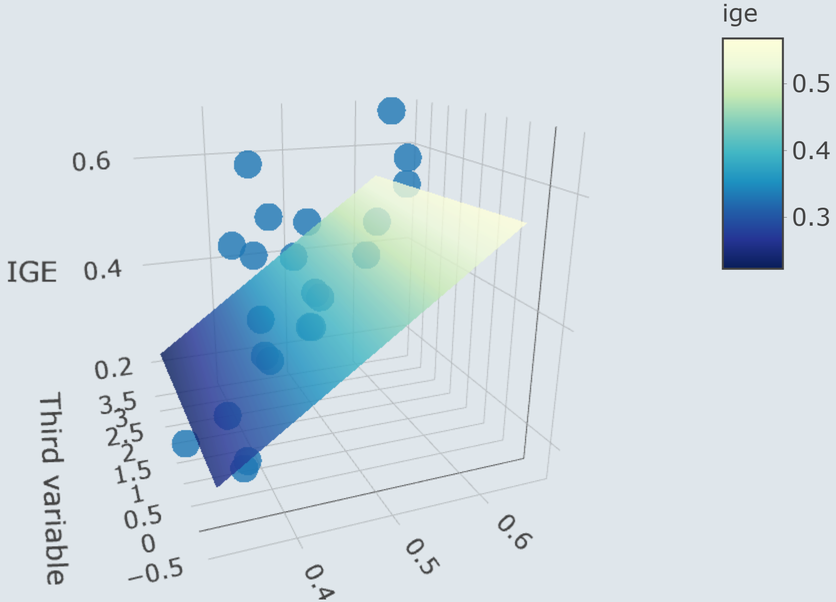
- $\hat{\alpha}$ and $\hat{\beta}$ minimize $\hat{\varepsilon}_i$

Quick reminder

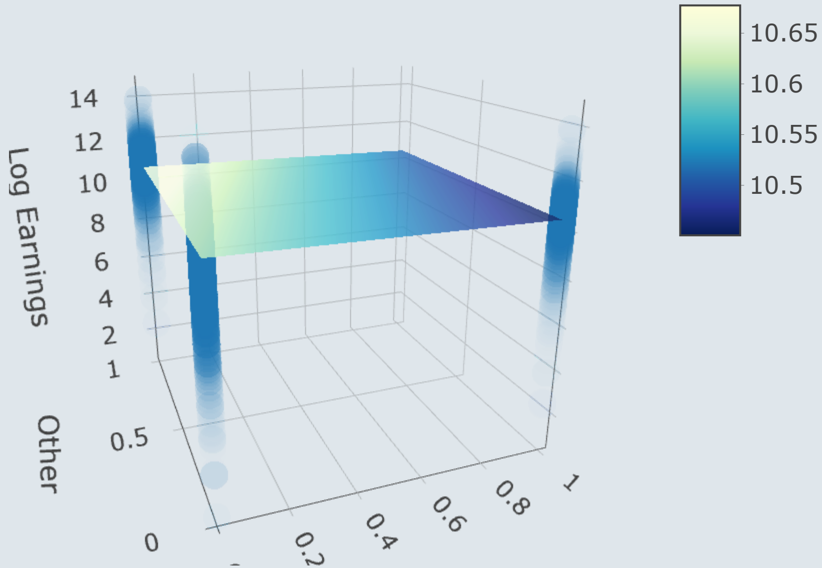
2. Multivariate regressions

- **Adding** a second independent **variable** in the regression amounts to **fitting a plane** instead of a line
  - Adding a third variable would fit an hyperplane of dimension 3 and so on

Adding a continuous variable



Adding a discrete variable

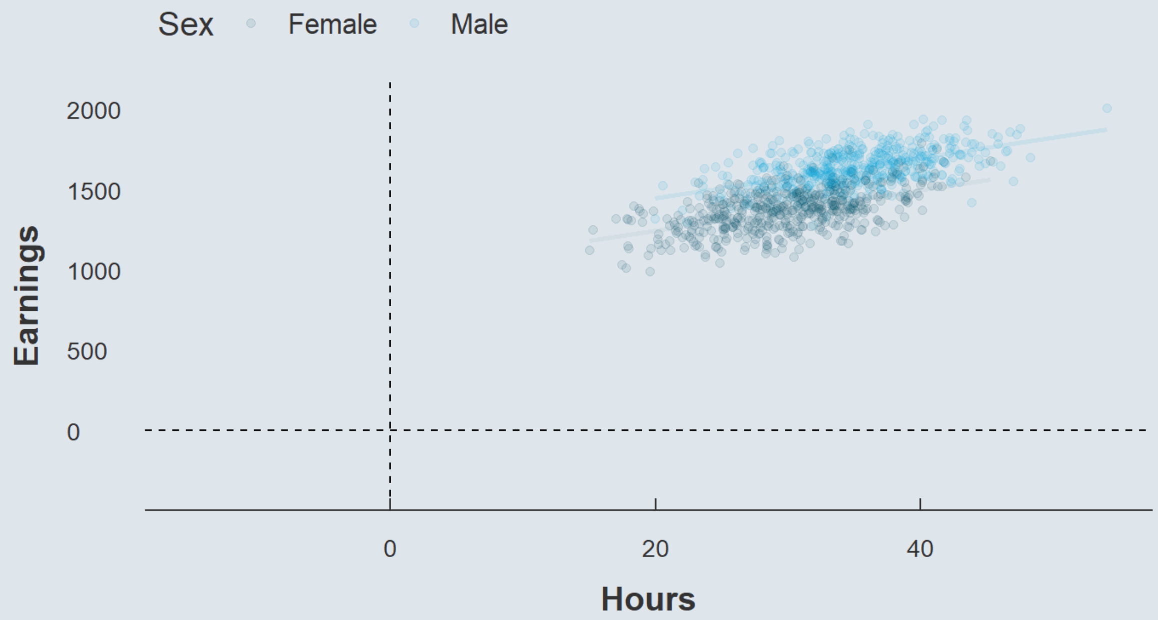


Quick reminder

3. Control variables

- Adding a third variable z **removes** its potential **confounding effect** from the relationship between x and y
  - As we move along the x axis, the **third variable remains constant**

$$\hat{y}_i = \hat{\alpha} + \hat{\beta}_1 x + \hat{\beta}_2 z + \hat{\varepsilon}_i$$

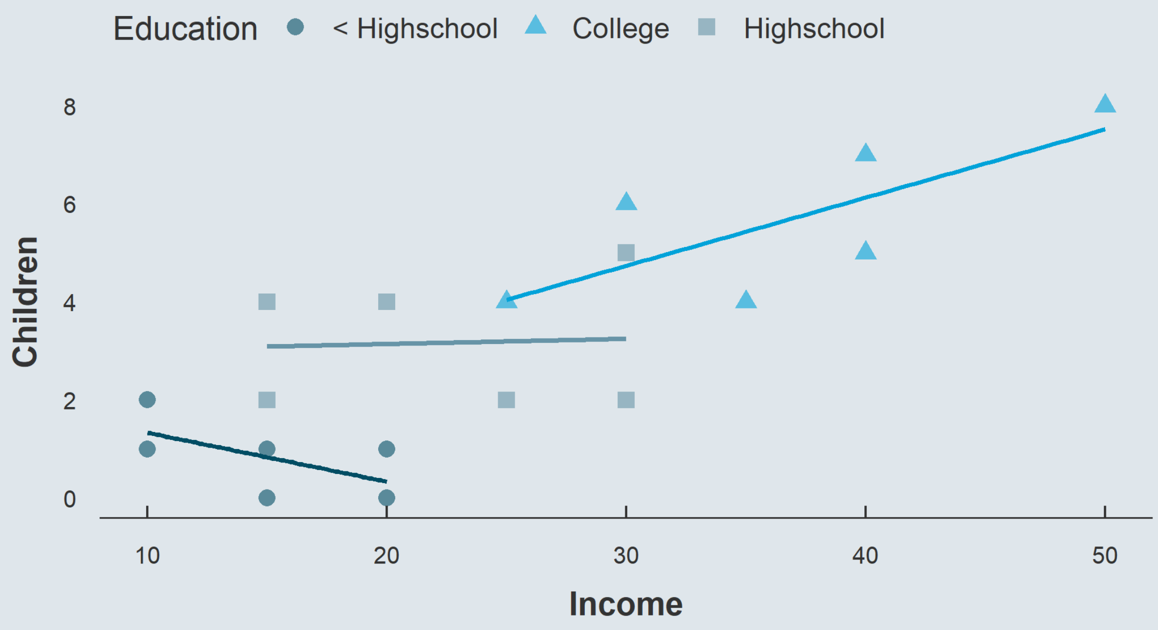


Quick reminder

4. Interactions

- Adding an **interaction** term with z allows to see **how the effect** of x on y **varies** with z
  - If z is **discrete**, it amounts to **regressing** y on x **separately** for each z group

$$\hat{y}_i = \hat{\alpha} + \hat{\beta}_1 x + \hat{\beta}_2 z + \hat{\beta}_3 (x \times z) + \hat{\varepsilon}_i$$



Today: Inference

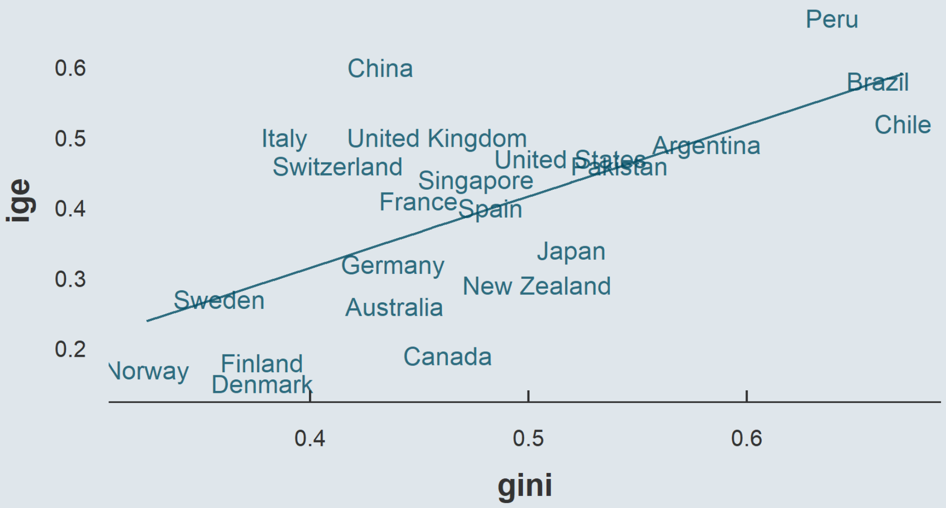
Today: Inference

1. Asymptotic inference

1.1. Data generating process

- In Part I of the course, we distinguished the **empirical moments** from the **theoretical moments**
  - Like the *empirical mean* is a **finite-sample estimation** of the *theoretical expected value*
  - The same principle applies to **regression coefficients**

- Take our Great Gatsby Curve for instance
  -
  -

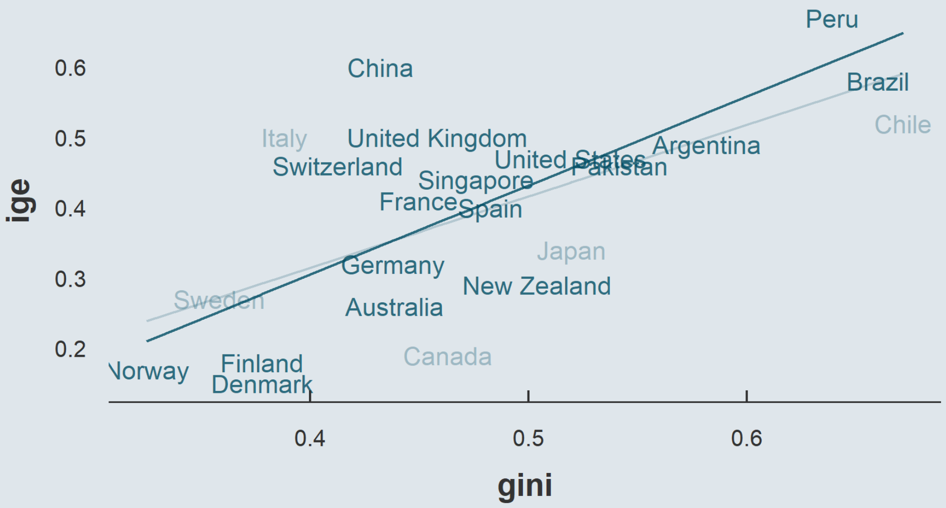


1. Asymptotic inference

1.1. Data generating process

- In Part I of the course, we distinguished the **empirical moments** from the **theoretical moments**
    - Like the *empirical mean* is a **finite-sample estimation** of the *theoretical expected value*
    - The same principle applies to **regression coefficients**

- Take our Great Gatsby Curve for instance
    - Had our **sample** of countries been a bit **different**, our **coefficients** would **not be the same**
    -

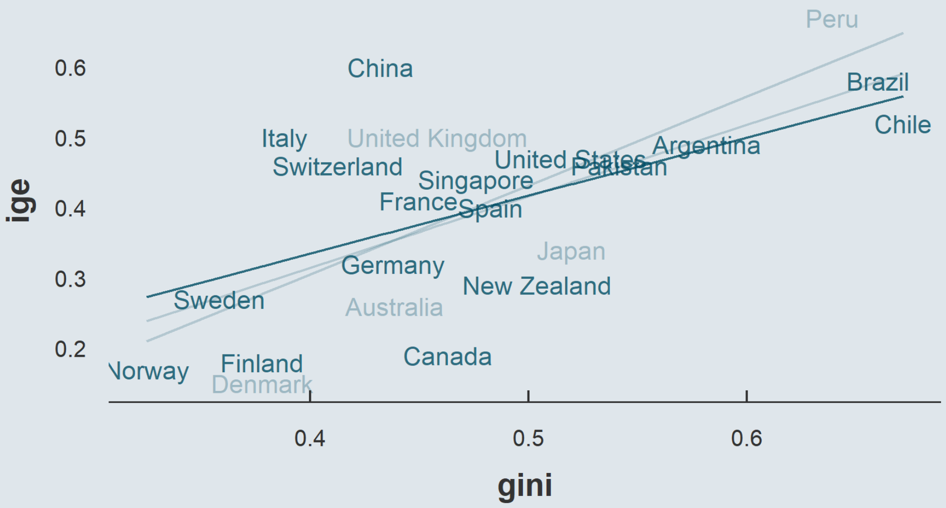


1. Asymptotic inference

1.1. Data generating process

- In Part I of the course, we distinguished the **empirical moments** from the **theoretical moments**
  - Like the *empirical mean* is a **finite-sample estimation** of the *theoretical expected value*
  - The same principle applies to **regression coefficients**

- Take our Great Gatsby Curve for instance
  - Had our **sample** of countries been a bit **different**, our **coefficients** would **not be the same**
  - But they would all be **estimations** of a true relationship whose **data-generating** process is **unobserved**



Then how to asses the reliability of our estimation?

1. Asymptotic inference

1.1. Data generating process

- For **simplicity**, let's work with a relationship whose **DGP is know**
  - Such that we can **understand how estimations** from random samples **behave relative to the DGP**
  - Let's **generate data in R!**

- We can use **functions** that output **random draws from given distributions** whose parameters can be chosen

Normal distribution

→ Sample size, expected value, standard deviation

```
rnorm(n = 10, mean = 100, sd = 5)
```

```
##  [1] 103.48482 102.78332  96.55622
##  [4]  96.46252 101.82291 103.84266
##  [7]  99.43827 104.40554 101.99053
## [10]  96.93987
```

Uniform distribution

→ Sample size, lower bound, upper bound

```
runif(n = 10, min = 4, max = 5)
```

```
##  [1] 4.633493 4.213208 4.129372
##  [4] 4.478118 4.924074 4.598761
##  [7] 4.976171 4.731793 4.356727
## [10] 4.431474
```

1. Asymptotic inference

1.1. Data generating process

- Consider the following **data generating process:**

$$y = -2 + 0.4 \times x + \varepsilon \quad \begin{cases} x \sim \mathcal{N}(4,\, 25) \\ \varepsilon \sim \mathcal{N}(0,\, 1) \end{cases}$$

- We can randomly draw **1,000 observations**

```
dt <- tibble(x = rnorm(1000, 4, 5),
             e = rnorm(1000, 0, 1),
             y = -2 + (.4 * x) + e)
```

Check the empirical moments:

```
c(mean(dt$x), var(dt$x))
```

```
## [1]  4.127842 26.816044
```

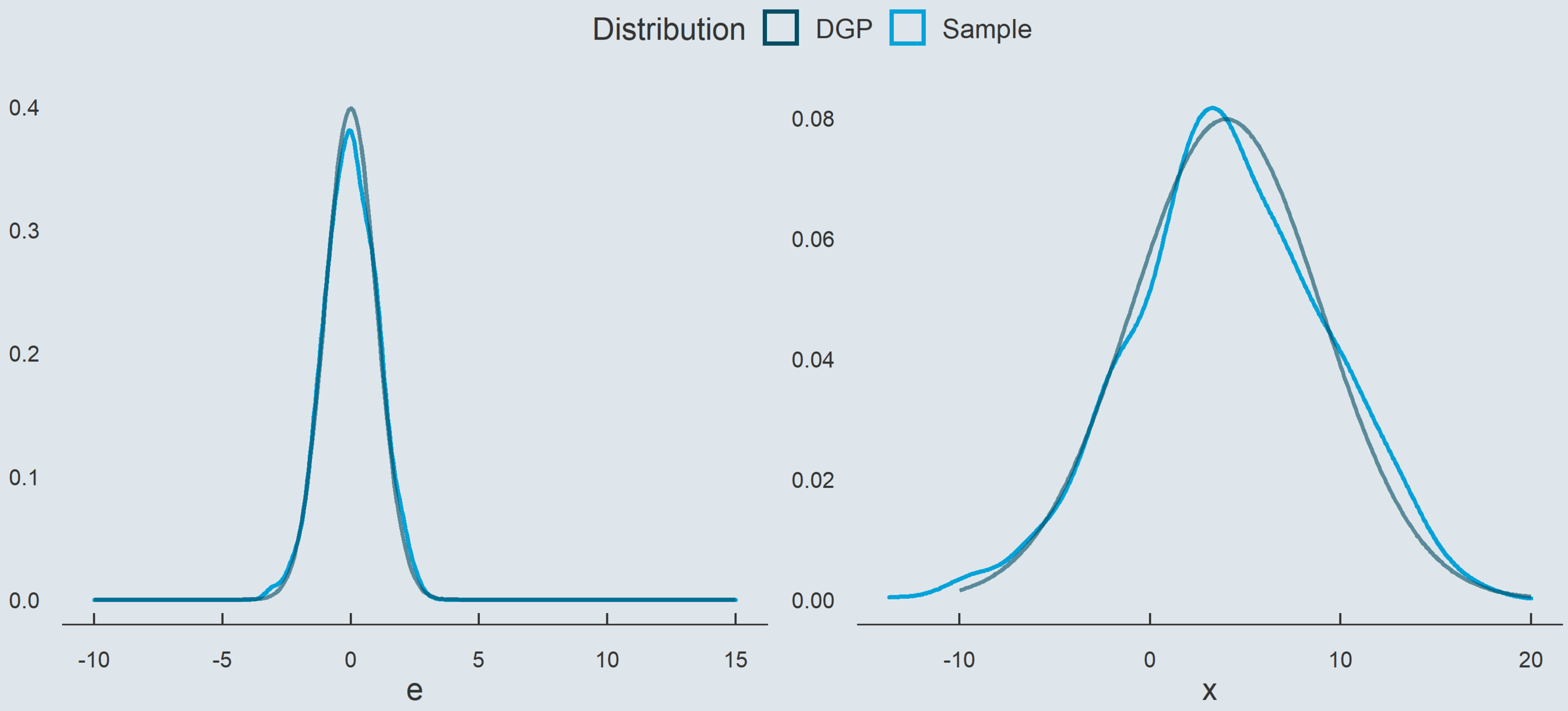
```
c(mean(dt$e), var(dt$e))
```

```
## [1] 0.009459055 1.070444754
```

1. Asymptotic inference

1.1. Data generating process

- Because the randomly drawn **sample is finite**, it **does not match exactly** the features of the DGP:



1. Asymptotic inference

1.1. Data generating process

- Same thing for the **coefficients** of the relationship between x and y:

```
lm(y ~ x, dt)
```

```
##
## Call:
## lm(formula = y ~ x, data = dt)
##
## Coefficients:
## (Intercept)            x
##     -2.0044       0.4033
```

- But what would happen if we were to **redo this operation many times?**
  1. **Draw a random sample** from the DGP
  2. **Compute the slope** of the regression of y on x
  3. Do it many times and **store the coefficients**

1. Asymptotic inference

1.1. Data generating process

- We can **use a loop** to do that:
  -
  -
  -

```
#

for (i in 1:1000) {

#
#
#

#

#

}
```

1. Asymptotic inference

1.1. Data generating process

- We can **use a loop** to do that:
  - First we create an empty vector
  -
  -

```
beta <- c()

for (i in 1:1000) {

#
#
#

#

#

}
```

1. Asymptotic inference

1.1. Data generating process

- We can **use a loop** to do that:
  - First we create an empty vector
  - Then we put the code in a loop
  -

```
beta <- c()

for (i in 1:1000) {

  dt_i <- tibble(x = rnorm(1000, 4, 5),
                 e = rnorm(1000, 0, 1),
                 y = -2 + (.4 * x) + e)

  reg_i <- lm(y ~ x, dt_i)

#

}
```

1. Asymptotic inference

1.1. Data generating process

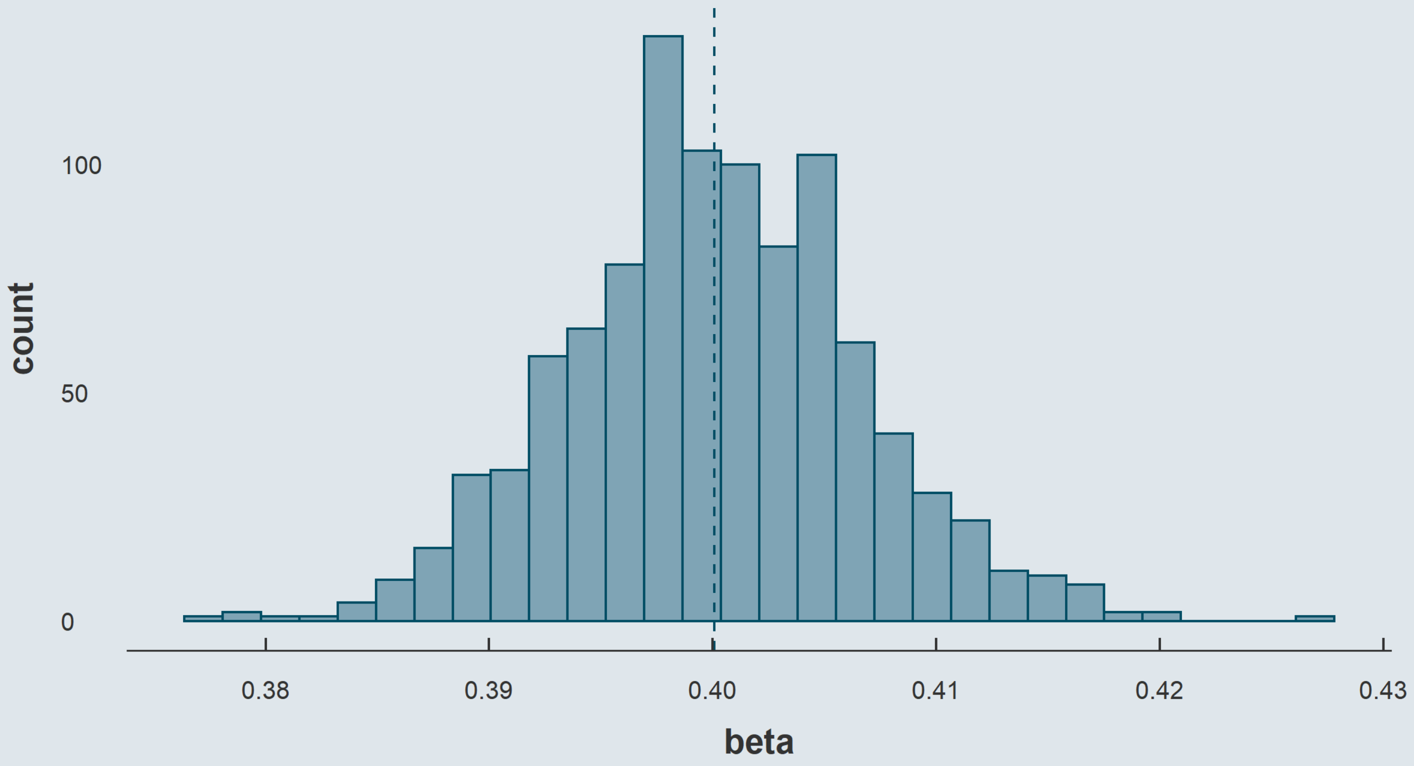
- We can **use a loop** to do that:
  - First we create an empty vector
  - Then we put the code in a loop
  - And we fill the vector at each iteration

```
beta <- c()

for (i in 1:1000) {

  dt_i <- tibble(x = rnorm(1000, 4, 5),
                 e = rnorm(1000, 0, 1),
                 y = -2 + (.4 * x) + e)

  reg_i <- lm(y ~ x, dt_i)

  beta <- c(beta, reg_i$coefficients[2])

}
```

1. Asymptotic inference

1.1. Data generating process

- We now have **1,000 slope coefficients** from 1,000 random samples of the **same DGP**

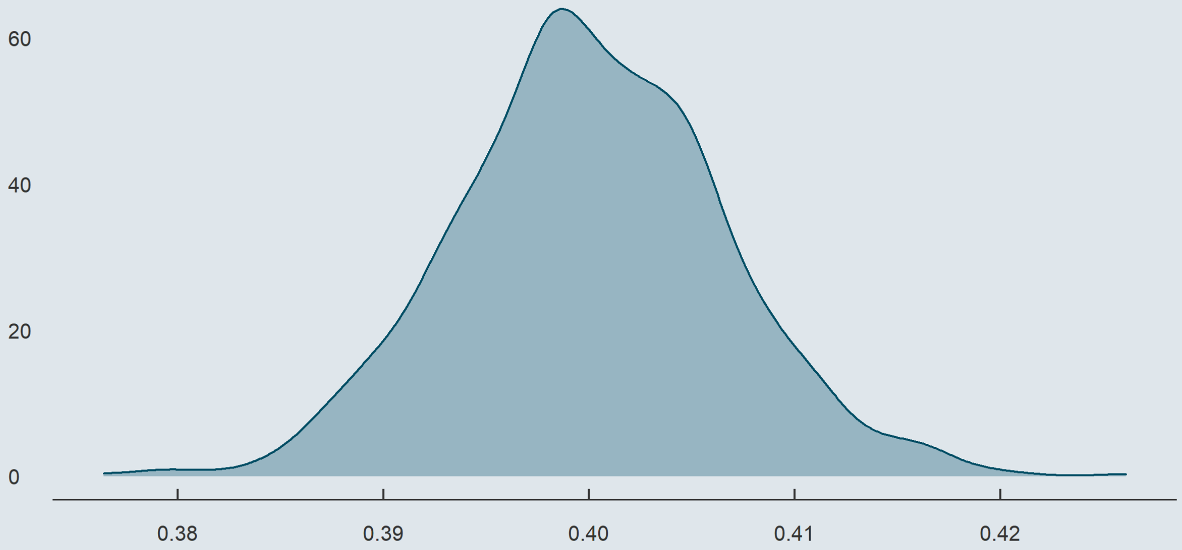


- Some random samples give higher estimates than others
- But **on expectation** we get the **right coefficient!**
- The $\hat{\beta}$s actually follow a **normal distribution**
- And at the limit their mean would **converge towards** β

1. Asymptotic inference

1.2. Standardization

- That is crucial information because it allows to get back to something we know:
  -
  -
  -

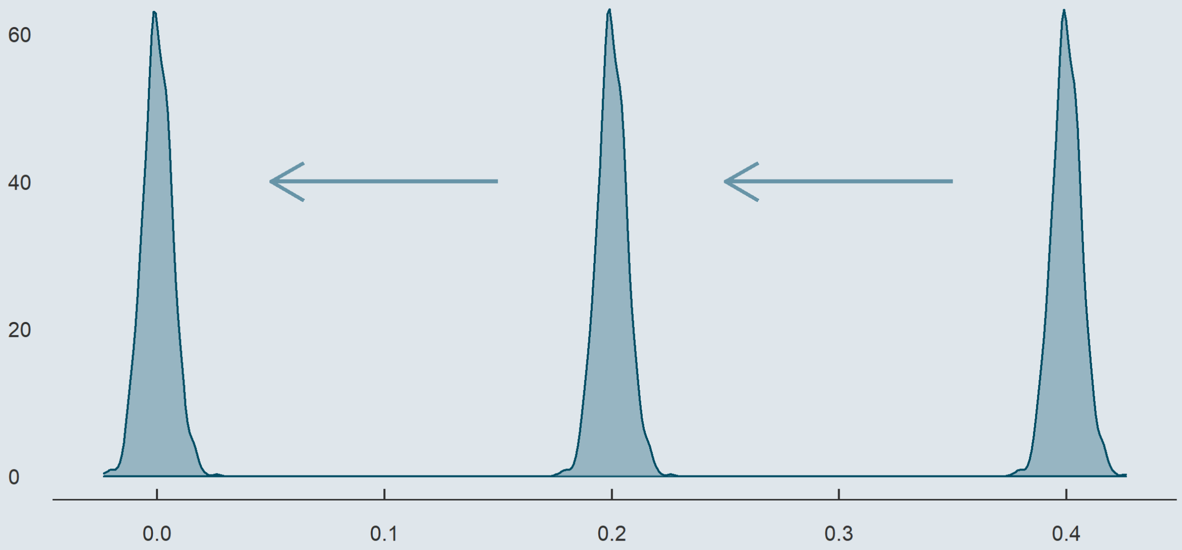


$\hat{\beta}$

1. Asymptotic inference

1.2. Standardization

- That is crucial information because it allows to get back to something we know:
  - By subtracting β from the distribution of $\hat{\beta}$
  -
  -

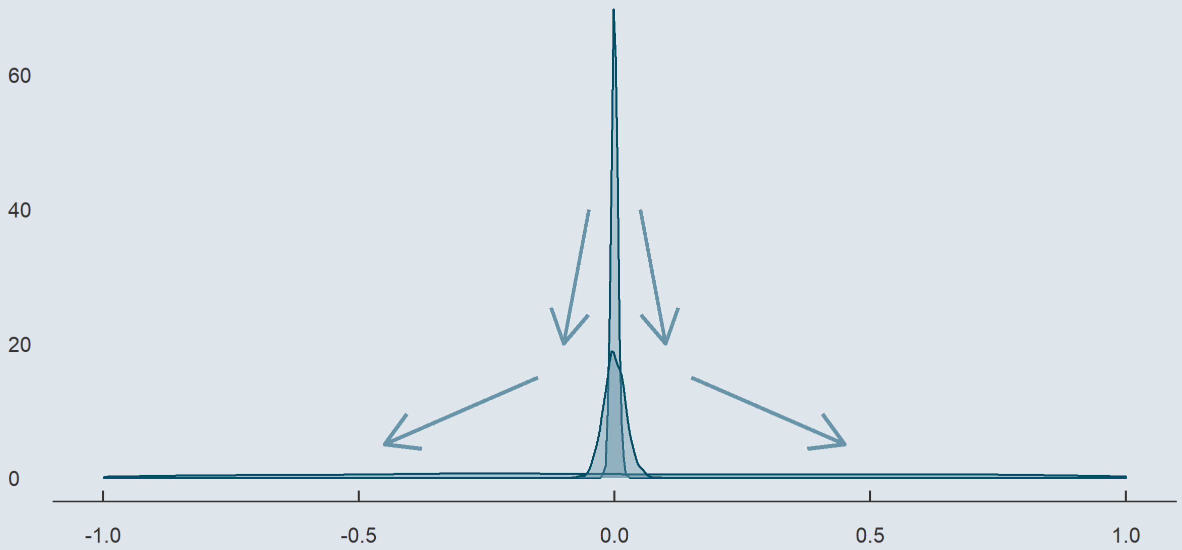


$$\hat{\beta} - \beta$$

1. Asymptotic inference

1.2. Standardization

- That is crucial information because it allows to get back to something we know:
  - By subtracting β from the distribution of $\hat{\beta}$
  - And dividing by the standard deviation of $\hat{\beta}$
  -



$$\frac{\hat{\beta} - \beta}{\text{SD}(\hat{\beta})}$$

1. Asymptotic inference

1.2. Standardization

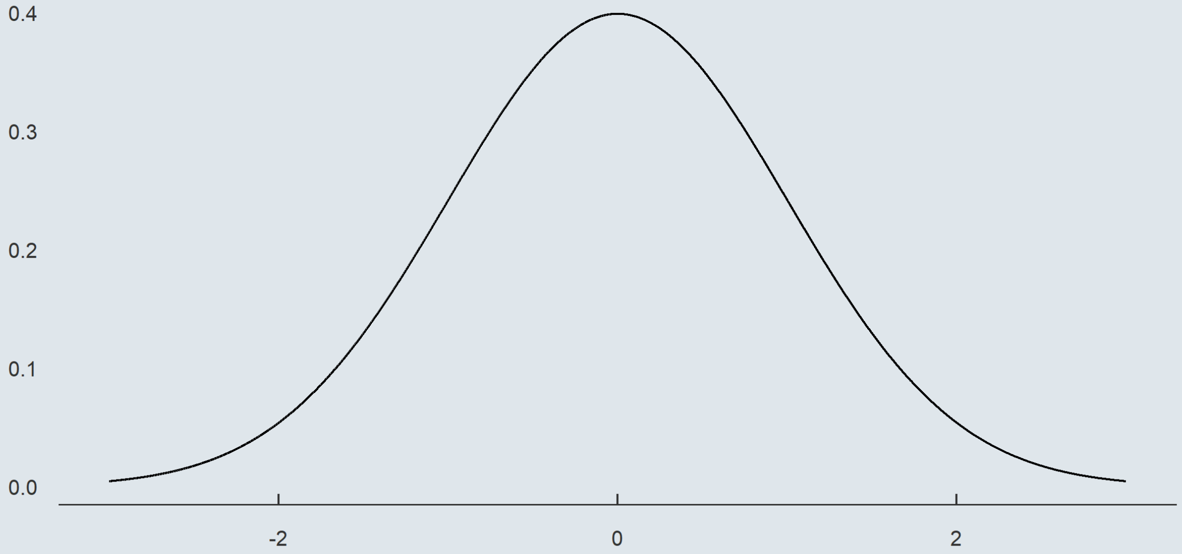
- That is crucial information because it allows to get back to something we know:
  - By subtracting β from the distribution of $\hat{\beta}$
  - And dividing by the standard deviation of $\hat{\beta}$
  - With an infinite sample we would obtain the standard normal distribution

$$\frac{\hat{\beta} - \beta}{\text{SD}(\hat{\beta})} \sim \mathcal{N}(0, 1)$$

1. Asymptotic inference

1.3. Confidence interval

- We can use the fact that we know the standard normal distribution:

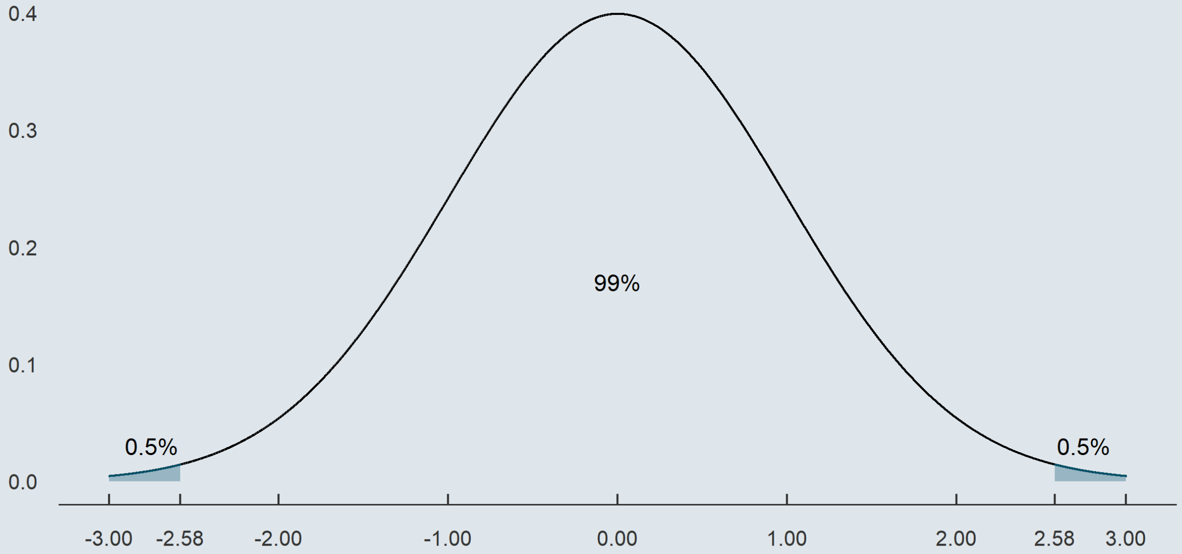


$$\frac{\hat{\beta} - \beta}{\mathrm{SD}(\hat{\beta})} \sim \mathcal{N}(0, 1)$$

1. Asymptotic inference

1.3. Confidence interval

- We can use the fact that we know the standard normal distribution:
  - That 99% of the distribution lie between $\pm$ 2.58
  -
  -



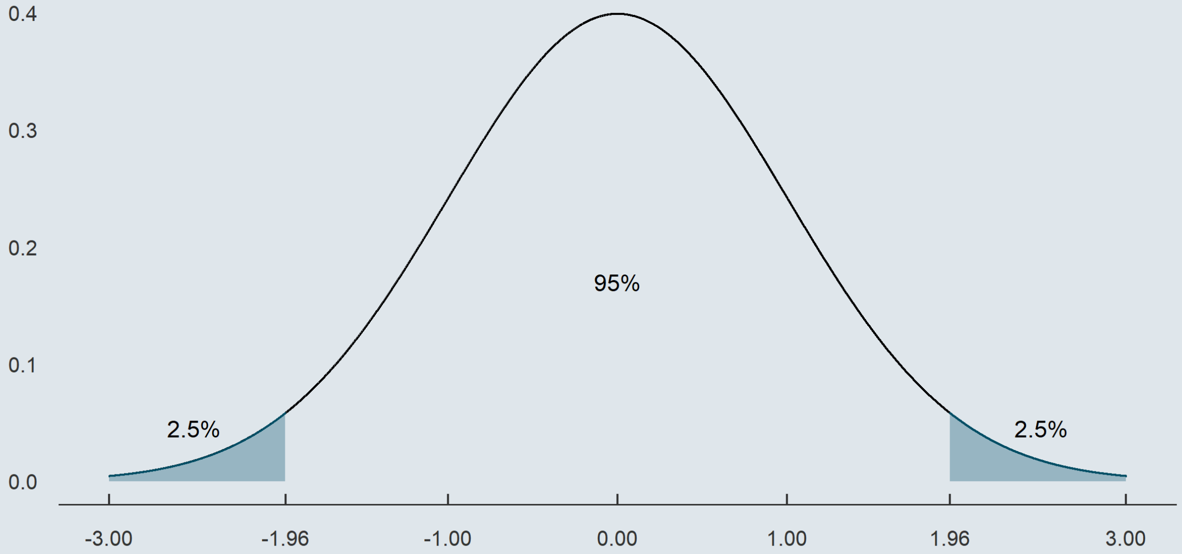
$$\frac{\hat{\beta}-\beta}{\text{SD}(\hat{\beta})} \sim \mathcal{N}(0,1)$$

$$\Pr\left[-2.58 < \frac{\hat{\beta}-\beta}{\text{SD}(\hat{\beta})} < 2.58\right] \approx 99\%$$

1. Asymptotic inference

1.3. Confidence interval

- We can use the fact that we know the standard normal distribution:
    - That 99% of the distribution lie between $\pm$ 2.58
    - That 95% of the distribution lie between $\pm$ 1.96
    -



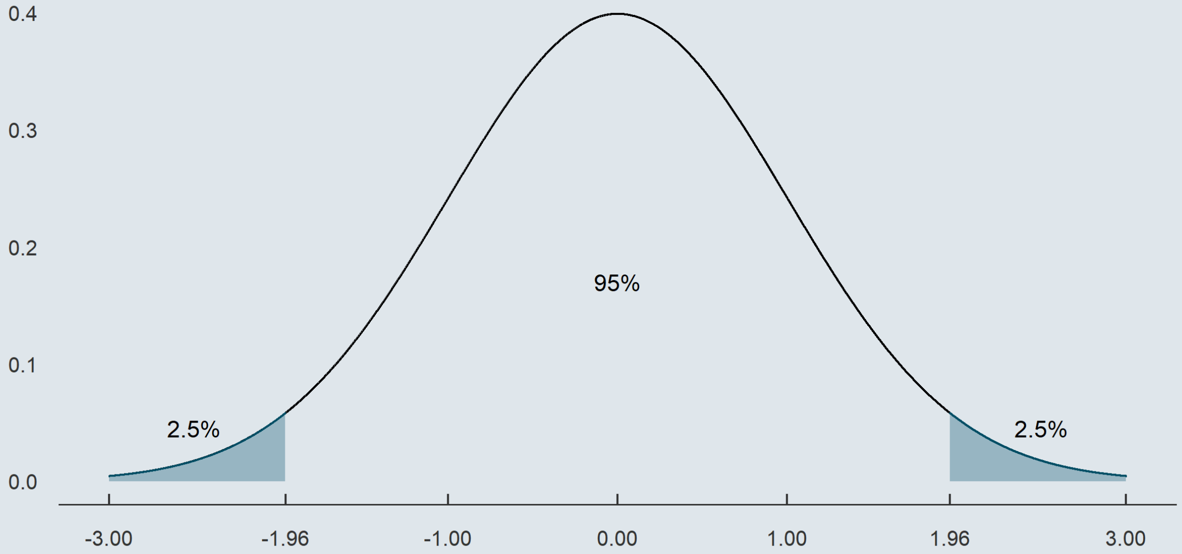
$$\frac{\hat{\beta}-\beta}{\mathrm{SD}(\hat{\beta})} \sim \mathcal{N}(0,1)$$

$$\Pr\left[-1.96 < \frac{\hat{\beta}-\beta}{\mathrm{SD}(\hat{\beta})} < 1.96\right] \approx 95\%$$

1. Asymptotic inference

1.3. Confidence interval

- We can use the fact that we know the standard normal distribution:
  - That 99% of the distribution lie between $\pm$ 2.58
  - That 95% of the distribution lie between $\pm$ 1.96
  - This is what allows to determine confidence intervals



$$\frac{\hat{\beta}-\beta}{\text{SD}(\hat{\beta})} \sim \mathcal{N}(0,1)$$

$$\text{Pr}\left[-1.96 < \frac{\hat{\beta}-\beta}{\text{SD}(\hat{\beta})} < 1.96\right] \approx 95\%$$

$$\Downarrow$$

$$\text{Confidence interval}$$

1. Asymptotic inference

1.3. Confidence interval

$$\Pr\left[-1.96 < \frac{\hat{\beta} - \beta}{\mathrm{SD}(\hat{\beta})} < 1.96\right] \approx 95\%$$

$$\Pr\left[-1.96 \times \mathrm{SD}(\hat{\beta}) < \hat{\beta} - \beta < 1.96 \times \mathrm{SD}(\hat{\beta})\right] \approx 95\%$$

$$\Pr\left[-1.96 \times \mathrm{SD}(\hat{\beta}) - \hat{\beta} < -\beta < 1.96 \times \mathrm{SD}(\hat{\beta}) - \hat{\beta}\right] \approx 95\%$$

$$\Pr\left[+1.96 \times \mathrm{SD}(\hat{\beta}) + \hat{\beta} > \beta > -1.96 \times \mathrm{SD}(\hat{\beta}) + \hat{\beta}\right] \approx 95\%$$

$$\mathrm{CI}_{95\%}:\ \hat{\beta} \pm 1.96 \times \mathrm{SD}(\hat{\beta})$$

Overview

Overview

2. Exact inference

2.1. Standard error

- In the **previous section** I used phrases like *"at the limit"* or ***"with an infinite sample"***
  - But **in practice** this is **not the case**, so things behave slightly differently
  - And this implies to make a few **statistical adjustments** to account for that

$$\hat{\beta} \pm 1.96 \times \text{SD}(\hat{\beta})$$

- First we **cannot measure** directly the **standard deviation** of $\hat{\beta}$
  - Indeed in practice we have **only one observation** of $\hat{\beta}$, not its whole distribution
  - But like for the mean, we can compute a **standard error** instead

Standard deviation	Standard error
➜ *Measures the amount of variability, or dispersion, from the individual data values to the mean*	➜ *Measures how far an estimate from a given sample is likely to be from the true parameter of interest*

2. Exact inference

2.1. Standard error

- We won't go through the theoretical computations together, but let's have a look at the formula:

$$\text{se}(\hat{\beta}) = \sqrt{\widehat{\text{Var}(\hat{\beta})}} = \sqrt{\frac{\sum_{i=1}^{n} \hat{\varepsilon_i}^2}{(n - \#\text{parameters}) \sum_{i=1}^{n} (x_i - \bar{x})^2}}$$

- Notice that the variance, and thus the standard error of our estimate, decreases as:

2. Exact inference

2.1. Standard error

- We won't go through the theoretical computations together, but let's have a look at the formula:

$$\text{se}(\hat{\beta}) = \sqrt{\widehat{\text{Var}(\hat{\beta})}} = \sqrt{\frac{\sum_{i=1}^{n} \hat{\varepsilon_i}^2}{(n - \#\text{parameters}) \sum_{i=1}^{n}(x_i - \bar{x})^2}}$$

- Notice that the variance, and thus the standard error of our estimate, decreases as:
  - The number of of observations gets bigger

2. Exact inference

2.1. Standard error

- We won't go through the theoretical computations together, but let's have a look at the formula:

$$\text{se}(\hat{\beta}) = \sqrt{\widehat{\text{Var}(\hat{\beta})}} = \sqrt{\frac{\sum_{i=1}^{n} \hat{\varepsilon}_i^{\,2}}{(n - \#\text{parameters}) \sum_{i=1}^{n} (x_i - \bar{x})^2}}$$

- Notice that the variance, and thus the standard error of our estimate, decreases as:
  - The number of of observations gets bigger
  - The number of parameters decreases

2. Exact inference

2.1. Standard error

- We won't go through the theoretical computations together, but let's have a look at the formula:

$$\text{se}(\hat{\beta}) = \sqrt{\widehat{\text{Var}(\hat{\beta})}} = \sqrt{\frac{\sum_{i=1}^{n} \hat{\varepsilon}_i^{\,2}}{(n - \#\text{parameters}) \sum_{i=1}^{n} (x_i - \bar{x})^2}}$$

- Notice that the variance, and thus the standard error of our estimate, decreases as:
  - The number of of observations gets bigger
  - The number of parameters decreases
  - The sum of squared errors relative to the variance of x decreases

2. Exact inference

2.1. Standard error

- We won't go through the theoretical computations together, but let's have a look at the formula:

$$\text{se}(\hat{\beta}) = \sqrt{\widehat{\text{Var}(\hat{\beta})}} = \sqrt{\frac{\sum_{i=1}^{n} \hat{\varepsilon_i}^2}{(n - \#\text{parameters}) \sum_{i=1}^{n} (x_i - \bar{x})^2}}$$

- Notice that the variance, and thus the standard error of our estimate, decreases as:
  - The number of of observations gets bigger
  - The number of parameters decreases
  - The sum of squared errors decreases relative to the variance of x

- And as the **standard error** gets **bigger, the confidence interval gets bigger:**

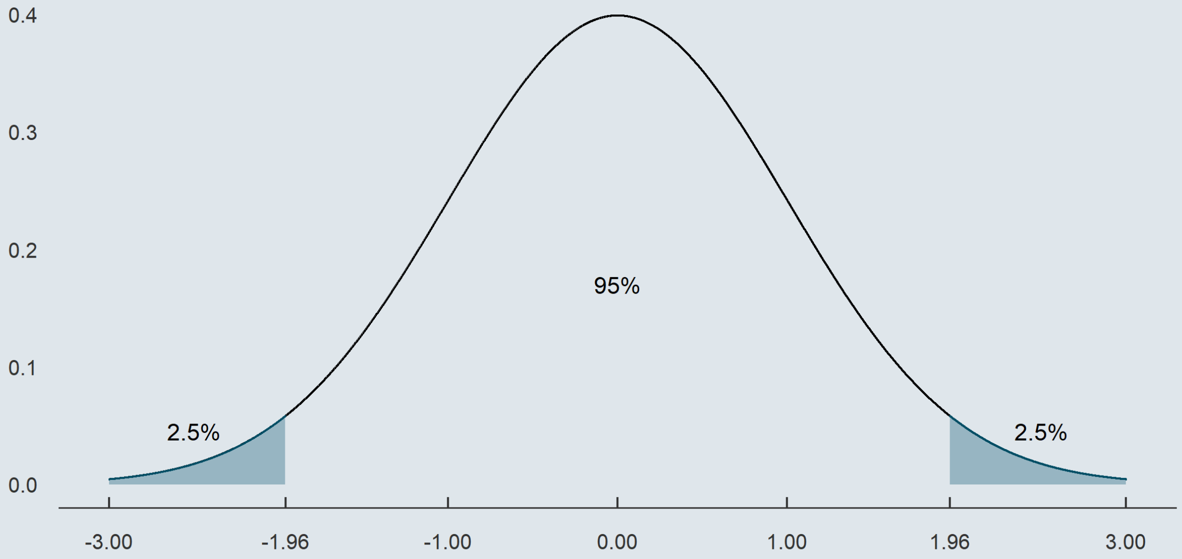
$$\hat{\beta} \pm 1.96 \times \text{se}(\hat{\beta})$$

2. Exact inference

2.2. Student-t distribution

- But that's not it, remember that we took the value **1.96** from the **normal distribution**

$$\hat{\beta} \pm 1.96 \times \text{se}(\hat{\beta})$$

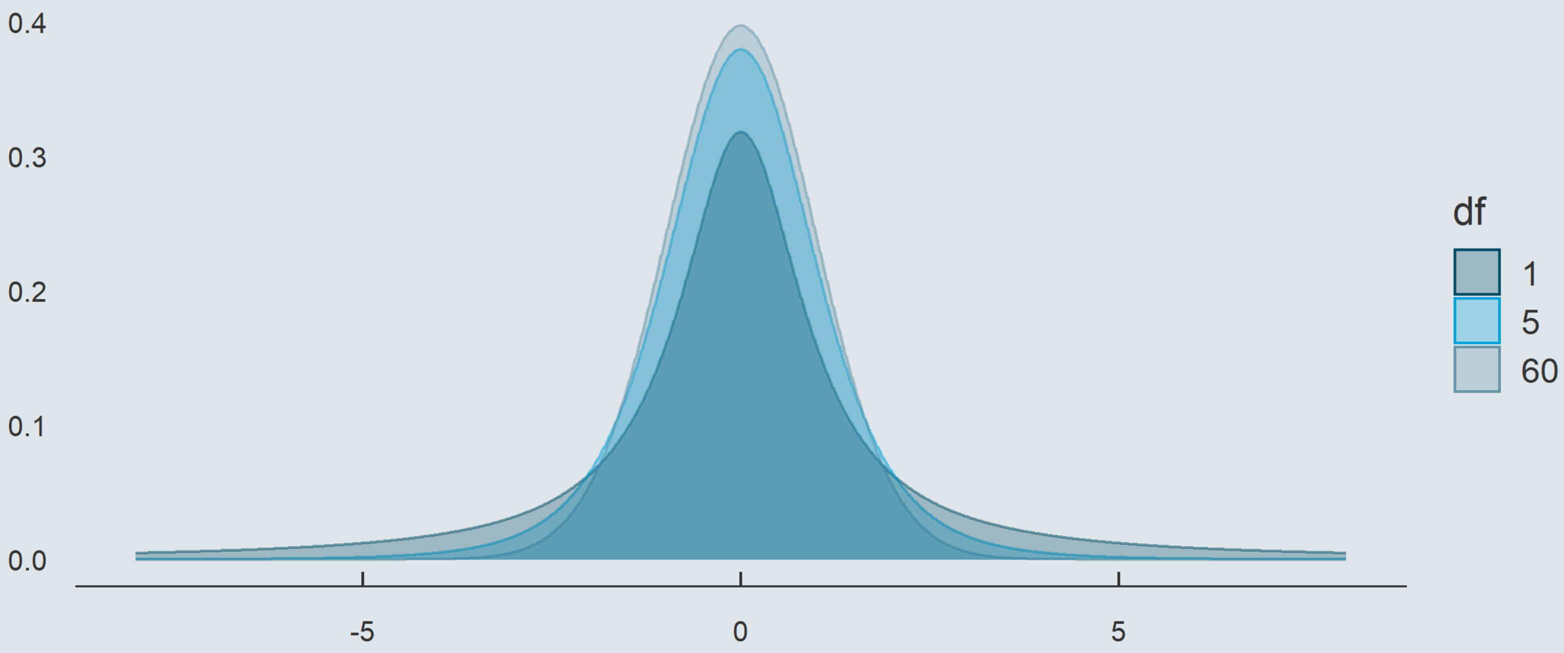


- But the **normal distribution** is what $\frac{\hat{\beta}-\beta}{\text{SD}(\hat{\beta})}$ converges to ***at the limit***
- In the **finite** world, $\frac{\hat{\beta}-\beta}{\text{se}(\hat{\beta})}$ follows a slightly flatter distribution
- The **Student t distribution**, whose precise shape depends on the number of observations we have and parameters we estimate

2. Exact inference

2.2. Student-t distribution

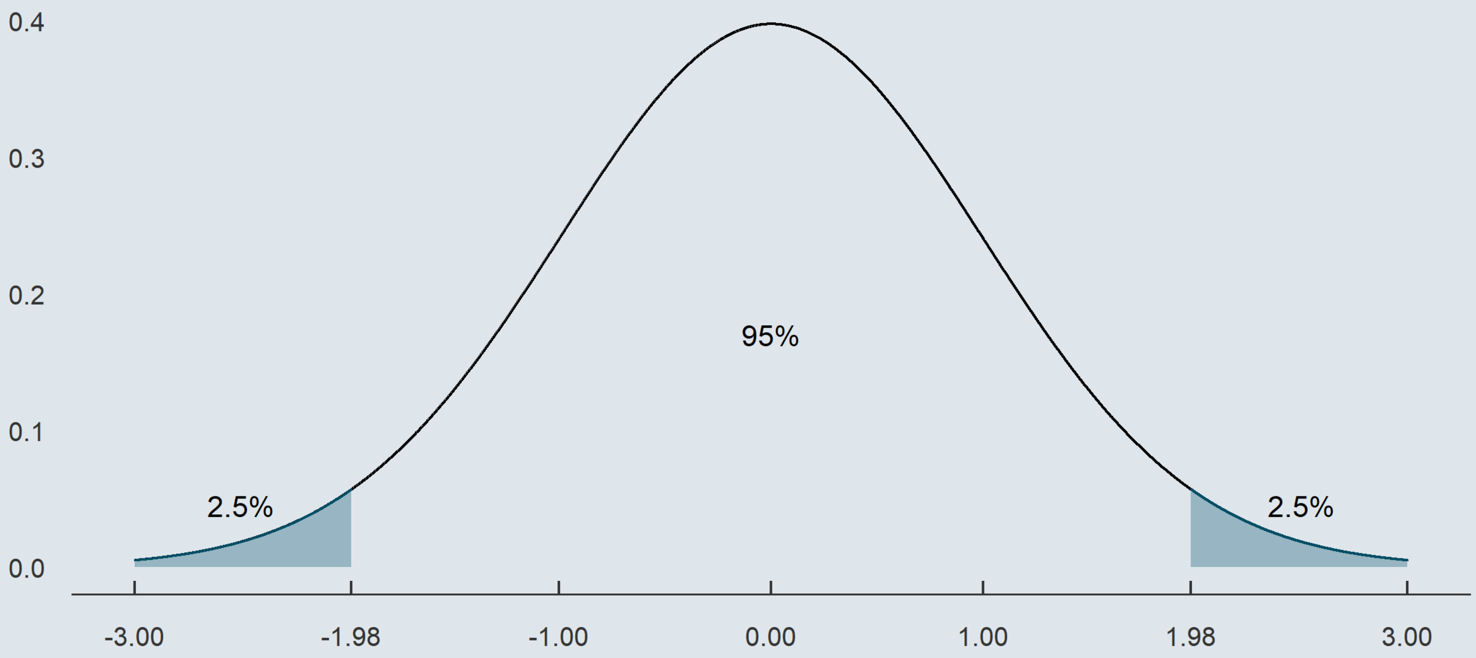
- The Student t distribution **accounts for** the fact that the **sample is finite**
  - The lower the number of **degrees of freedom** (#observations - #parameters) the flatter
  - And it **tends to a normal** distribution as the number of degrees of freedom $\rightarrow\ \infty$



2. Exact inference

2.2. Student-t distribution

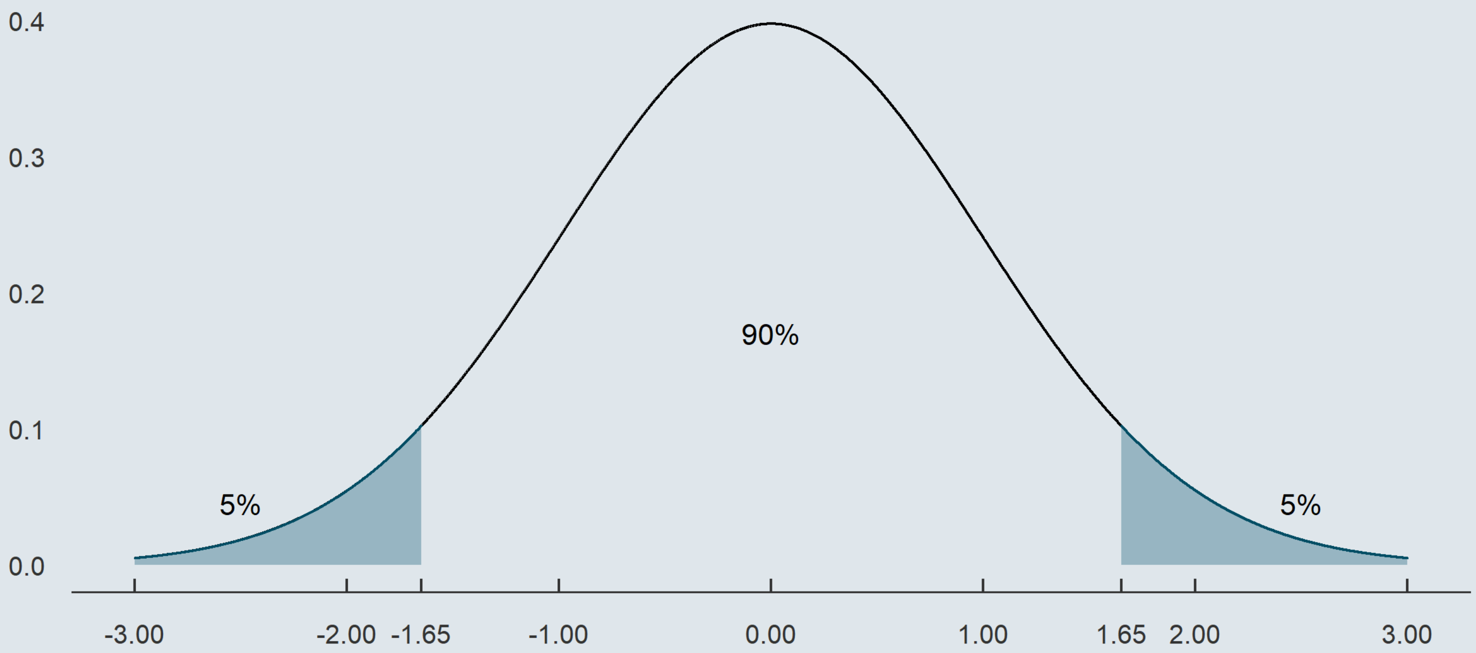
- But **we know the Student t distributions** just as well as the standard normal distribution
  - With 100 degrees of freedom, 95% of the distribution lie between $\pm$ 1.98
  -



2. Exact inference

2.2. Student-t distribution

- But **we know the Student t distributions** just as well as the standard normal distribution
  - With 100 degrees of freedom, 95% of the distribution lie between $\pm$ 1.98
  - With 3,000 degrees of freedom, 90% of the distribution lie between $\pm$ 1.65



2. Exact inference

2.2. Student-t distribution

- So **instead of 1.96**, we must use the **value such that:**
  - The **desired percentage** of the distribution is comprised within $\pm$ that value...
  - For a Student t distribution with the relevant number of **degrees of freedom**

- We can get these values easily with the **qt()** function, indicating:
  -
  -

```
qt( , )
```

2. Exact inference

2.2. Student-t distribution

- So **instead of 1.96**, we must use the **value such that:**
  - The **desired percentage** of the distribution is comprised within $\pm$ that value...
  - For a Student t distribution with the relevant number of **degrees of freedom**

- We can get these values easily with the **qt()** function, indicating:
  - The share of the distribution below the value we're looking for (e.g., 0.975 for a 95% CI)
  -

```
qt(.975, )
```

2. Exact inference

2.2. Student-t distribution

- So **instead of 1.96**, we must use the **value such that:**
  - The **desired percentage** of the distribution is comprised within $\pm$ that value...
  - For a Student t distribution with the relevant number of **degrees of freedom**

- We can get these values easily with the **qt()** function, indicating:
  - The share of the distribution below the value we're looking for (e.g., 0.975 for a 95% CI)
  - The number of degrees of freedom of the Student t distribution (e.g., 88 observations - 2 parameters)

```
qt(.975, 86)
```

```
## [1] 1.987934
```

- Denote this value $t(\text{df})_{1-\frac{\alpha}{2}}$
  - With α equal to $1-$ the confidence level
  - And df the number of degrees of freedom

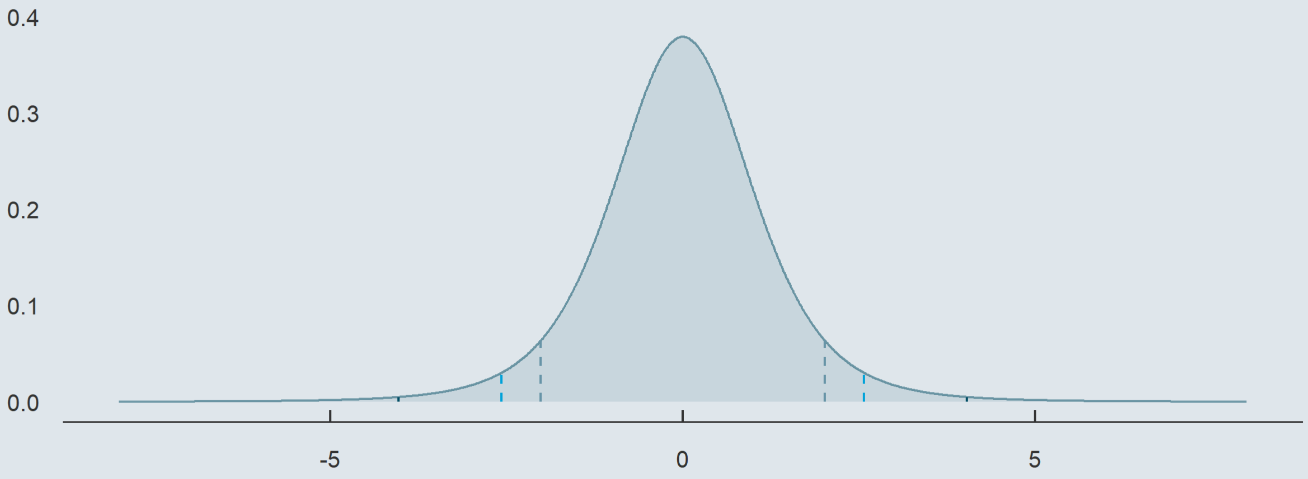
2. Exact inference

2.3. Confidence interval

- The formula for the confidence interval in finite sample hence writes:

$$\hat{\beta} \pm t(\text{df})_{1-\frac{\alpha}{2}} \times \text{se}(\hat{\beta})$$

- The confidence interval increases as:
    - The confidence level increases
    -



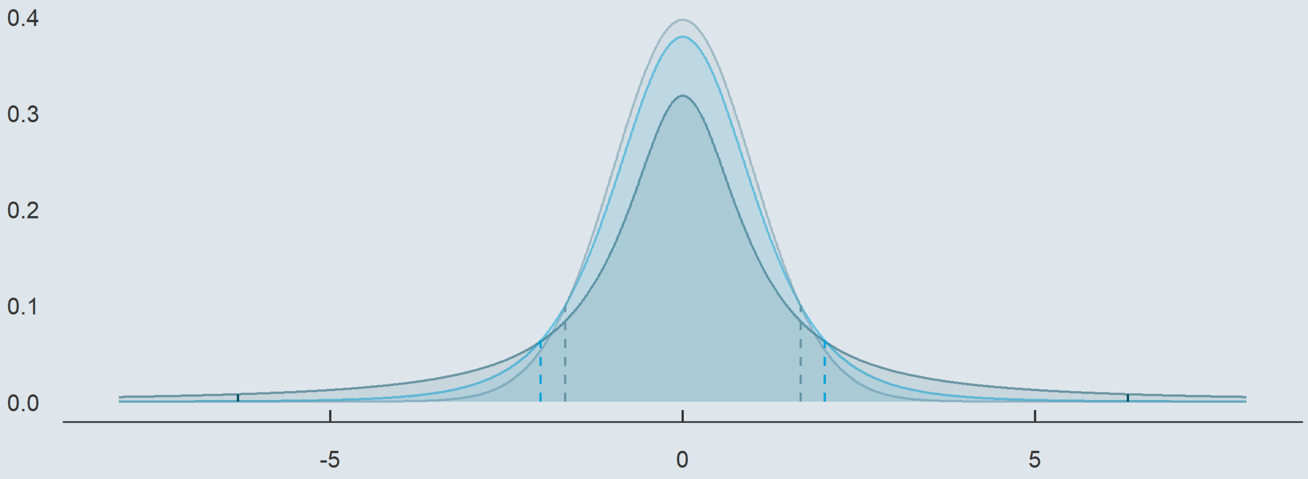
2. Exact inference

2.3. Confidence interval

- The formula for the confidence interval in finite sample hence writes:

$$\hat{\beta} \pm t(\text{df})_{1-\frac{\alpha}{2}} \times \text{se}(\hat{\beta})$$

- The confidence interval increases as:
  - The confidence level increases
  - The number of degrees of freedom decreases



Practice

10:00

1) Import the `ggcurve.csv` dataset

2) Regress the IGE on the Gini coefficient and store the estimated regression parameters

3) Compute the 95% confidence interval of the regression slope

$$\hat{\beta} \pm t(\text{df})_{1-\frac{\alpha}{2}} \times \text{se}(\hat{\beta})$$

$$\text{se}(\hat{\beta}) = \sqrt{\frac{\sum_{i=1}^{n} \hat{\varepsilon}_i^{\,2}}{(n - \#\text{parameters}) \sum_{i=1}^{n} (x_i - \bar{x})^2}}$$

You've got 10 minutes!

Solution

1) Import the `ggcurve.csv` dataset

```
ggcurve <- read.csv("C:/User/Documents/ggcurve.csv")
```

2) Regress the IGE on the Gini coefficient and store the regression slope

```
model <- lm(ige ~ gini, ggcurve)
model
```

```
## 
## Call:
## lm(formula = ige ~ gini, data = ggcurve)
## 
## Coefficients:
## (Intercept)         gini  
##    -0.09129      1.01546
```

```
alpha <- model$coefficients[1]
beta <- model$coefficients[2]
```

Solution

3) Compute the 95% confidence interval of the regression slope

```
se_dat <- ggcurve %>%
  mutate(fit = alpha + gini * beta, e = ige - fit) %>%
  summarise(se = sqrt(sum(e^2)/((n()-2)*sum((gini-mean(gini))^2))))

se_dat$se
```

```
## [1] 0.2642477
```

```
beta - se_dat$se * qt(.975, nrow(ggcurve) - 2)
```

```
##      gini
## 0.4642511
```

```
beta + se_dat$se * qt(.975, nrow(ggcurve) - 2)
```

```
##     gini
## 1.566673
```

Overview

Overview

3. Hypothesis testing

3.1. P-value

- We now have the **95% confidence interval** for our estimate:
  - Our estimate of β is 1.02
  - And we are 95% sure that β lies between 0.46 and 1.57

$$\overset{0.46}{|}\underbrace{-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-}_{t(\text{df})_{1-\frac{\alpha}{2}}\times\text{se}(\hat{\beta})}\overset{1.02}{\cdot}\underbrace{-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-}_{t(\text{df})_{1-\frac{\alpha}{2}}\times\text{se}(\hat{\beta})}\overset{1.57}{|}$$

- Note that in our **confidence interval** formula:
  - The **standard error** and the relevant Student t **distribution** are **given**
  - But the **confidence level** $1-\alpha$ was **chosen arbitrarily**

→ Setting a **higher confidence** level would **widen the confidence interval**
→ Allowing for a **lower confidence** level would **narrow the confidence interval**

3. Hypothesis testing

3.1. P-value

- So far we framed the problem as:

"What are the values β is likely to take under a given confidence level?"

- But we could also think of it as:

"Under which confidence level is β is likely to take a given value?"

- And this is actually a very **practical way of framing the question:**
  - To (in)validate the predictions from a theoretical model
  - To know under which confidence level β is likely to be $\neq 0$ at all

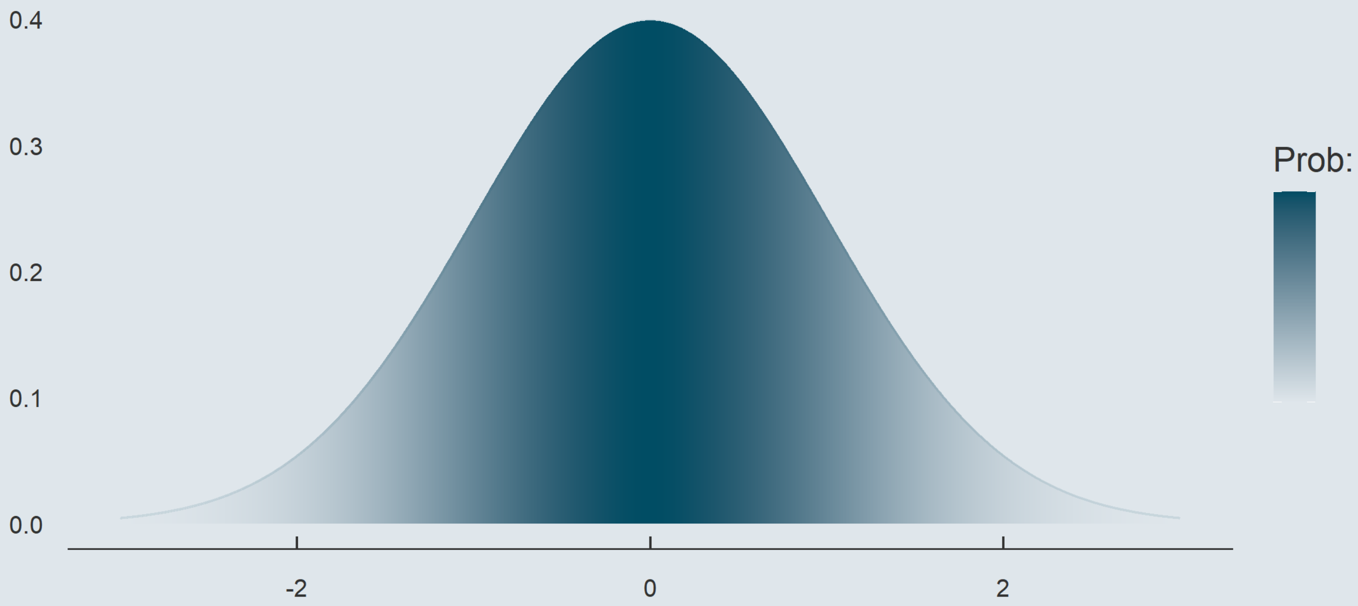
➜ But how to answer such questions in practice?

3. Hypothesis testing

3.1. P-value

- We can start from the fact that even though we do not know β, we know that:

$$\frac{\hat{\beta} - \beta}{\text{se}(\hat{\beta})} \sim t(\text{df})$$

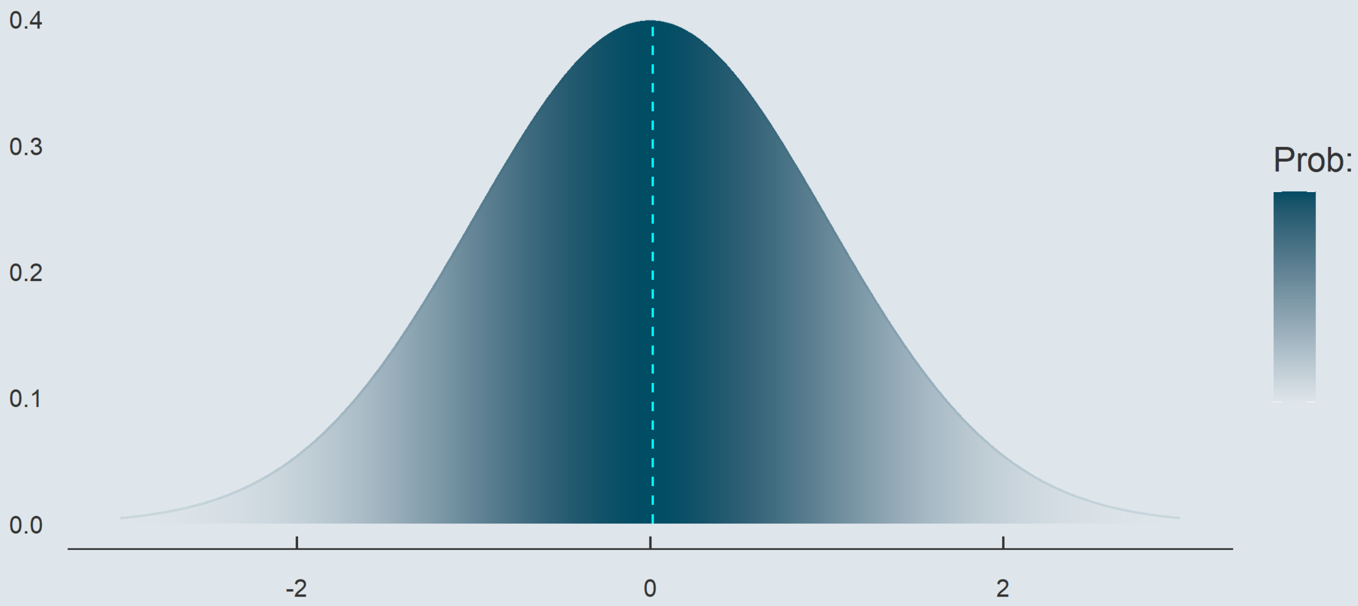


3. Hypothesis testing

3.1. P-value

- And that in this distribution some values are quite plausible:

$$\frac{\hat{\beta} - 1}{\text{se}(\hat{\beta})}$$

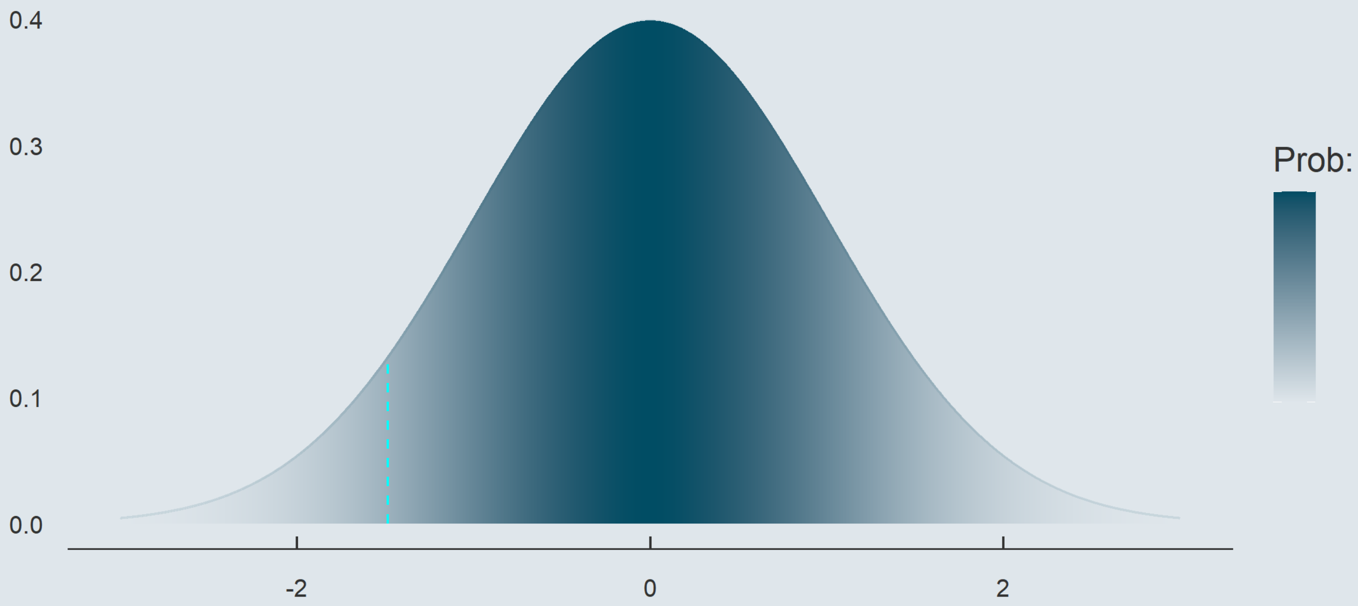


3. Hypothesis testing

3.1. P-value

- And some are way less plausible:

$$\frac{\hat{\beta} - 2.5}{\text{se}(\hat{\beta})}$$



3. Hypothesis testing

3.1. P-value

- But because the distribution is **continuous:**
  -
  -

3. Hypothesis testing

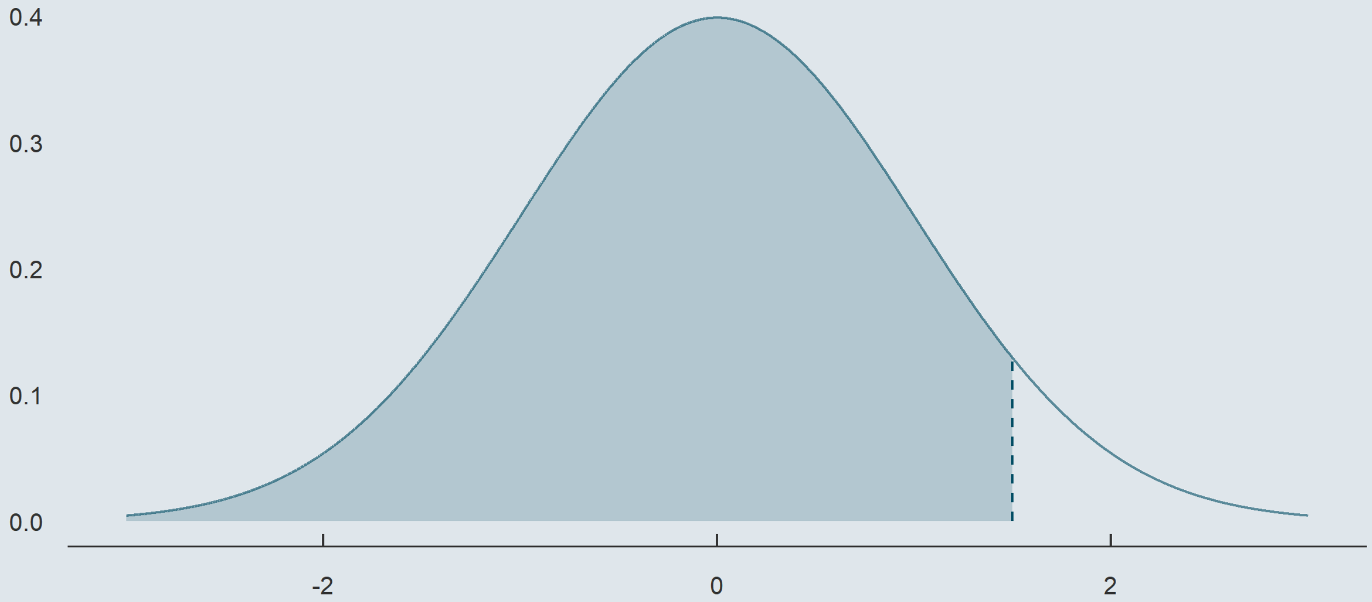
3.1. P-value

- But because the distribution is **continuous:**
  - The **probability** to draw any **exact value** would be **0**
  -

3. Hypothesis testing

3.1. P-value

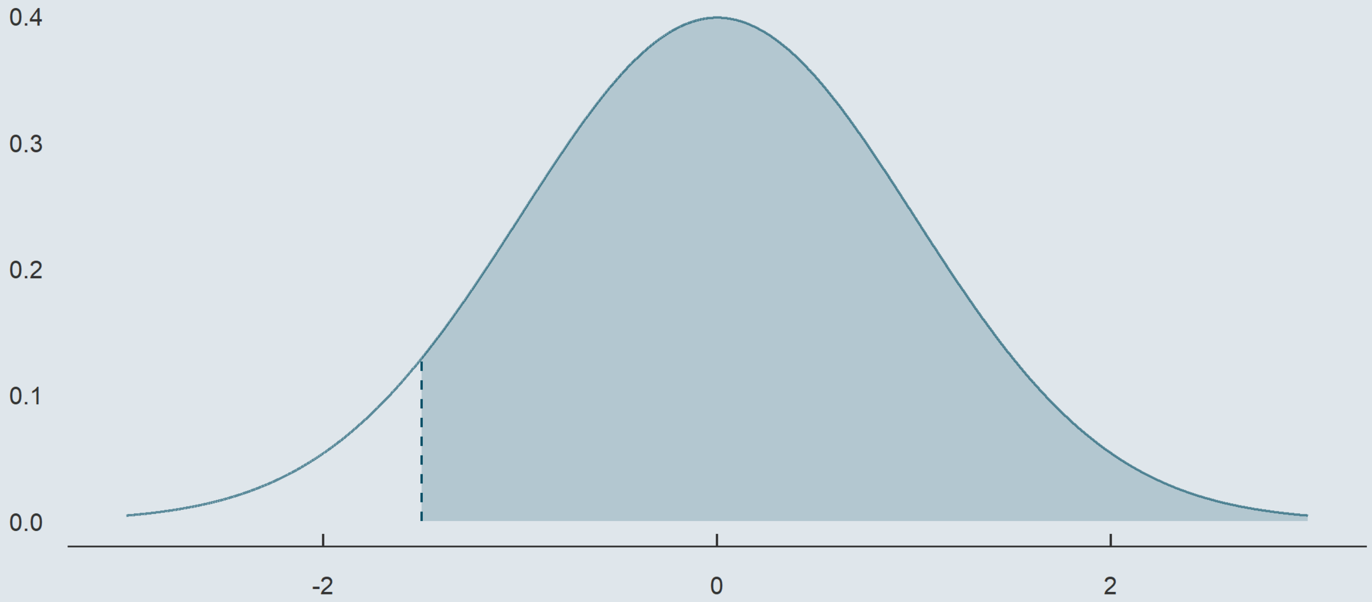
- But because the distribution is **continuous:**
  - The **probability** to draw any **exact value** would be **0**
  - We can only compute the **probability to fall below** that value



3. Hypothesis testing

3.1. P-value

- But because the distribution is **continuous:**
  - The **probability** to draw any **exact value** would be **0**
  - **Or to fall above** that value **if** it is **negative**



3. Hypothesis testing

3.1. P-value

- But **generally** what makes sense is to know what are the **chances to fall *that far* from 0:**
    -
    -

3. Hypothesis testing

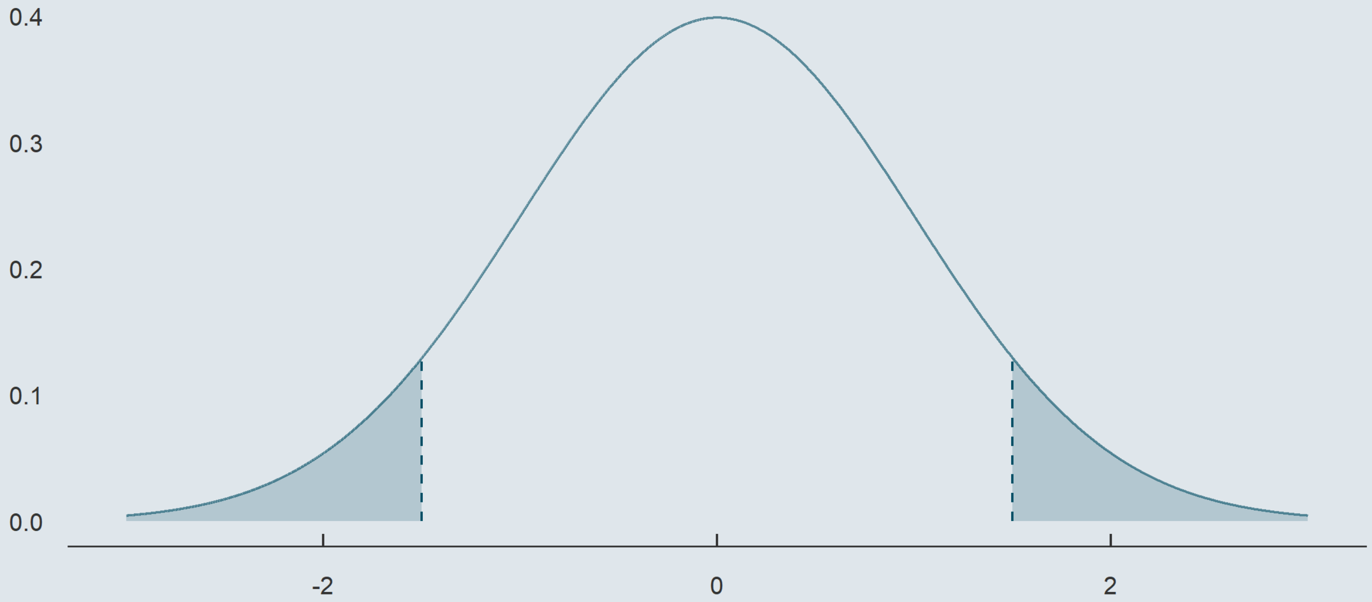
3.1. P-value

- But **generally** what makes sense is to know what are the **chances to fall *that far* from 0:**
  - So we take 1 - the probability to fall below the absolute value
  -

3. Hypothesis testing

3.1. P-value

- But **generally** what makes sense is to know what are the **chances to fall *that far* from 0:**
  - So we take 1 - the probability to fall below the absolute value
  - And we multiply it by 2



3. Hypothesis testing

3.1. P-value

- The **resulting area** is what we call a **p-value**
  - It is the probability that β falls at least as far from $\hat{\beta}$ as the hypothesized value

- Consider finding $\hat{\beta} = 4$ and a hypothesizing value of 3 for β
  - A p-value of 5% indicates that there is only a 5% chance to find a $\hat{\beta} = 4$ if $\beta = 3$
  - Below that threshold we would reject the hypothesis that $\beta = 3$ at the 95% confidence level

- Notice that in this example, the 95% confidence interval of $\hat{\beta}$ would not include the value 3
  - With a hypothesized value equal to the bound of a confidence interval the p-value would equal 1 - the corresponding confidence level
  - So a p-value lower than α means that the hypothesized value is outside the $(1 - \alpha)$% confidence interval

→ *Let's go through a formal example with our data*

3. Hypothesis testing

3.1. P-value

- Can we **reject** at the 95% confidence level that $\beta = 0$?

```
beta
```

```
##     gini
## 1.015462
```

- We should start by hypothesizing that $\beta = 0$
  - This is what we call the ***"null hypothesis"*** H_0

$$H_0 : \beta = 0$$

Under H_0:

$$\frac{\hat{\beta} - 0}{\text{se}(\hat{\beta})} \sim t(\text{df})$$

3. Hypothesis testing

3.1. P-value

- We should find the **area below** $(\hat{\beta} - 0)/\text{se}(\hat{\beta})$ in a Student t distribution we the right number of df
  - $(\hat{\beta} - 0)/\text{se}(\hat{\beta})$ is what we call **the t-stat**

```
(beta - 0) / se_dat$se
```

```
##     gini
## 3.842842
```

- While **qt()** gave us the **value** for a certain probability, **pt()** gives the the **probability** for a given value:
  -
  -

```
pt( , )
```

3. Hypothesis testing

3.1. P-value

- We should find the **area below** $(\hat{\beta} - 0)/\text{se}(\hat{\beta})$ in a Student t distribution we the right number of df
  - $(\hat{\beta} - 0)/\text{se}(\hat{\beta})$ is what we call **the t-stat**

```
(beta - 0) / se_dat$se
```

```
##     gini
## 3.842842
```

- While **qt()** gave us the **value** for a certain probability, **pt()** gives the the **probability** for a given value:
  - Put in the **t-stat**
  -

```
pt((beta - 0) / se_dat$se, )
```

3. Hypothesis testing

3.1. P-value

- We should find the **area below** $(\hat{\beta} - 0)/\text{se}(\hat{\beta})$ in a Student t distribution we the right number of df
  - $(\hat{\beta} - 0)/\text{se}(\hat{\beta})$ is what we call **the t-stat**

```
(beta - 0) / se_dat$se
```

```
##     gini
## 3.842842
```

- While **qt()** gave us the **value** for a certain probability, **pt()** gives the the **probability** for a given value:
  - Put in the **t-stat**
  - And the **degrees of freedom**

```
pt((beta - 0) / se_dat$se, nrow(ggcurve) - 2)
```

```
##      gini
## 0.9994921
```

3. Hypothesis testing

3.1. P-value

- We must then:
  - Take **1 - this probability** (area above the t-stat)
  -

```
1 - pt((beta - 0) / se_dat$se, nrow(ggcurve) - 2)
```

```
##         gini
## 0.0005078528
```

3. Hypothesis testing

3.1. P-value

- We must then:
  - Take **1 - this probability** (area above the t-stat)
  - And **multiply it by 2** (consider the absolute distance and not the signed distance)

```
2 * (1 - pt((beta - 0) / se_dat$se, nrow(ggcurve) - 2))
```

```
##        gini
## 0.001015706
```

- The **p-value** is **lower than 1%:**
  - We can **reject at the 99% confidence level** that $\beta = 0$
  - In that case we say that $\hat{\beta}$ is **significantly different from 0** at the 1% significance level

- But the **p-value** is **greater than 0.1%:**
  - We **cannot reject at the 99.9% confidence level** that $\beta = 0$
  - In that case we say that $\hat{\beta}$ is **not significantly different from 0** at the 0.1% significance level

3. Hypothesis testing

3.1. P-value

- By default, the **summary()** function **tests** whether or not each coefficient is significantly **different from 0**
    -

```r
summary(lm(ige ~ gini, ggcurve))
```

3. Hypothesis testing

3.1. P-value

- By default, the **summary()** function **tests** whether or not each coefficient is significantly **different from 0**
  - You can **extract** the information from the **$coefficient** attribute of the output

```
summary(lm(ige ~ gini, ggcurve))$coefficients
```

```
##                Estimate Std. Error    t value    Pr(>|t|)
## (Intercept) -0.09129311  0.1287045 -0.7093234 0.486311455
## gini         1.01546204  0.2642477  3.8428420 0.001015706
```

- For each coefficient it indicates:
  - The standard error
  - The t-stat $(H_0 : \beta = 0)$
  - The p-value $(H_0 : \beta = 0)$

- The output of the **summary()** function is great to have a **quick overview** of the model:

```
summary(lm(ige ~ gini, ggcurve))
```

3. Hypothesis testing

3.1. P-value

```
##
## Call:
## lm(formula = ige ~ gini, data = ggcurve)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.188991 -0.088238 -0.000855  0.047284  0.252310
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.09129    0.12870  -0.709  0.48631
## gini         1.01546    0.26425   3.843  0.00102 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1159 on 20 degrees of freedom
## Multiple R-squared:  0.4247,    Adjusted R-squared:  0.396
## F-statistic: 14.77 on 1 and 20 DF,  p-value: 0.001016
```

3. Hypothesis testing

3.1. P-value

```
##
## Call:
## lm(formula = ige ~ gini, data = ggcurve)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.188991 -0.088238 -0.000855  0.047284  0.252310
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.09129    0.12870  -0.709  0.48631
## gini         1.01546    0.26425   3.843  0.00102 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1159 on 20 degrees of freedom
## Multiple R-squared:  0.4247,    Adjusted R-squared:  0.396
## F-statistic: 14.77 on 1 and 20 DF,  p-value: 0.001016
```

← Command

3. Hypothesis testing

3.1. P-value

```
##
## Call:
## lm(formula = ige ~ gini, data = ggcurve)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.188991 -0.088238 -0.000855  0.047284  0.252310
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.09129    0.12870  -0.709  0.48631
## gini         1.01546    0.26425   3.843  0.00102 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1159 on 20 degrees of freedom
## Multiple R-squared:  0.4247,    Adjusted R-squared:  0.396
## F-statistic: 14.77 on 1 and 20 DF,  p-value: 0.001016
```

← Command

← Residuals distribution

3. Hypothesis testing

3.1. P-value

```
##
## Call:
## lm(formula = ige ~ gini, data = ggcurve)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.188991 -0.088238 -0.000855  0.047284  0.252310
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.09129    0.12870  -0.709  0.48631
## gini         1.01546    0.26425   3.843  0.00102 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1159 on 20 degrees of freedom
## Multiple R-squared:  0.4247,    Adjusted R-squared:  0.396
## F-statistic: 14.77 on 1 and 20 DF,  p-value: 0.001016
```

← Command

← Residuals distribution

← Coefs, s.e., t-/p-values

3. Hypothesis testing

3.1. P-value

```
##
## Call:
## lm(formula = ige ~ gini, data = ggcurve)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.188991 -0.088238 -0.000855  0.047284  0.252310
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.09129    0.12870  -0.709  0.48631
## gini         1.01546    0.26425   3.843  0.00102 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1159 on 20 degrees of freedom
## Multiple R-squared:  0.4247,    Adjusted R-squared:  0.396
## F-statistic: 14.77 on 1 and 20 DF,  p-value: 0.001016
```

← Command

← Residuals distribution

← Coefs, s.e., t-/p-values

← Significance

3. Hypothesis testing

3.1. P-value

```
##
## Call:
## lm(formula = ige ~ gini, data = ggcurve)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.188991 -0.088238 -0.000855  0.047284  0.252310
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.09129    0.12870  -0.709  0.48631
## gini         1.01546    0.26425   3.843  0.00102 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1159 on 20 degrees of freedom
## Multiple R-squared:  0.4247,    Adjusted R-squared:  0.396
## F-statistic: 14.77 on 1 and 20 DF,  p-value: 0.001016
```

← Command

← Residuals distribution

← Coefs, s.e., t-/p-values

← Significance

← df and advanced stats

3. Hypothesis testing

3.2. linearHypothesis()

- But the **linearHypothesis()** function from the **car** package allows to **easily test** other **hypotheses:**
  -
  -

```
linearHypothesis( , )
```

3. Hypothesis testing

3.2. linearHypothesis()

- But the **linearHypothesis()** function from the **car** package allows to **easily test** other **hypotheses:**
  - You must provide the **model**
  -

```
linearHypothesis(lm(ige ~ gini, ggcurve), )
```

3. Hypothesis testing

3.2. linearHypothesis()

- But the **linearHypothesis()** function from the **car** package allows to **easily test** other **hypotheses:**
  - You must provide the **model**
  - And the **hypothesis** (referring to coefficients as in the summary)

```
linearHypothesis(lm(ige ~ gini, ggcurve), "gini = 0")
```

```
## Linear hypothesis test
##
## Hypothesis:
## gini = 0
##
## Model 1: restricted model
## Model 2: ige ~ gini
##
##   Res.Df     RSS Df Sum of Sq      F   Pr(>F)
## 1     21 0.46733
## 2     20 0.26883  1    0.1985 14.767 0.001016 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

3. Hypothesis testing

3.2. linearHypothesis()

- You can also test **more complex hypotheses**
  - Like equality between coefficients

```
linearHypothesis(lm(ige ~ gini, ggcurve), "gini = (Intercept)")
```

```
## Linear hypothesis test
##
## Hypothesis:
## - (Intercept)  + gini = 0
##
## Model 1: restricted model
## Model 2: ige ~ gini
##
##   Res.Df     RSS Df Sum of Sq      F  Pr(>F)
## 1     21 0.37634
## 2     20 0.26883  1   0.10751 7.9983 0.01039 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

3. Hypothesis testing

3.2. linearHypothesis()

- You can also test **more complex hypotheses**
  - Like equality between coefficients, or joint hypotheses (relying a generalization of the t-test called *F-test*)

```
linearHypothesis(lm(ige ~ gini, ggcurve), c("gini = 0", "(Intercept) = 0"))
```

```
## Linear hypothesis test
##
## Hypothesis:
## gini = 0
## (Intercept) = 0
##
## Model 1: restricted model
## Model 2: ige ~ gini
##
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1     22 3.8841
## 2     20 0.2688  2    3.6153 134.48 2.523e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Overview

1. Asymptotic inference ✓
- 1.1. Data generating process
- 1.2. Standardization
- 1.3. Confidence interval

2. Exact inference ✓
- 2.1. Standard error
- 2.2. Student-t distribution
- 2.3. Confidence interval

3. Hypothesis testing ✓
- 3.1. P-value
- 3.2. linearHypothesis()

4. Wrap up!

4. Wrap up!

Data generating process

- In practice we estimate coefficients on a **given realization of a data generating process**
  - So the **true coefficient** is **unobserved**
  - But our **estimation** is **informative** on the values the true coefficient is likely to take

$$\frac{\hat{\beta} - \beta}{\text{SD}(\hat{\beta})} \sim \mathcal{N}(0, 1)$$

4. Wrap up!

Confidence interval

- This allows to infer a **confidence interval:**

$$\hat{\beta} \pm t(\mathrm{df})_{1-\frac{\alpha}{2}} \times \mathrm{se}(\hat{\beta})$$

- Where $t(\mathrm{df})_{1-\frac{\alpha}{2}}$ is the value from a **Student t distribution**
  - With the relevant number of **degrees of freedom** df (n - #parameters)
  - And the desired **confidence level** $1-\alpha$

- And where $\mathrm{se}(\hat{\beta})$ denotes the **standard error** of $\hat{\beta}$:

$$\mathrm{se}(\hat{\beta}) = \sqrt{\widehat{\mathrm{Var}(\hat{\beta})}} = \sqrt{\frac{\sum_{i=1}^{n} \hat{\varepsilon_i}^2}{(n - \#\text{parameters}) \sum_{i=1}^{n} (x_i - \bar{x})^2}}$$

4. Wrap up!

P-value

- It also allows to **test** how likely is β to be **different from a given value:**
  - If the **p-value** < 5%, we can **reject** that β equals the **hypothesized value** at the 95% confidence level
  - This threshold, very common in Economics, implies that we have 1 chance out of 20 to be wrong

```r
linearHypothesis(lm(ige ~ gini, ggcurve), "gini = 0")
```

```
## Linear hypothesis test
##
## Hypothesis:
## gini = 0
##
## Model 1: restricted model
## Model 2: ige ~ gini
##
##   Res.Df     RSS Df Sum of Sq      F   Pr(>F)
## 1     21 0.46733
## 2     20 0.26883  1    0.1985 14.767 0.001016 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```