

Introduction and Overview

ECO 6973, Set 2

Jonathan Moreno-Medina

Fall 2021

Last Time

Motivation

In our last set of slides, we

1. discussed the **motivation** for studying causal inference and machine learning
2. **introduced R**—why we use it, what it can do
3. **started reviewing** material from basic econometrics course

These notes continue the review, building the foundation for some new topics (soon).

Review

Population vs. sample

Models and notation

We write our (simple) population model

$$y_i = \beta_0 + \beta_1 x_i + u_i$$

and our sample-based estimated regression model as

$$y_i = \hat{\beta}_0 + \hat{\beta}_1 x_i + e_i$$

An estimated regression model produces estimates for each observation:

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$$

which gives us the *best-fit* line through our dataset.

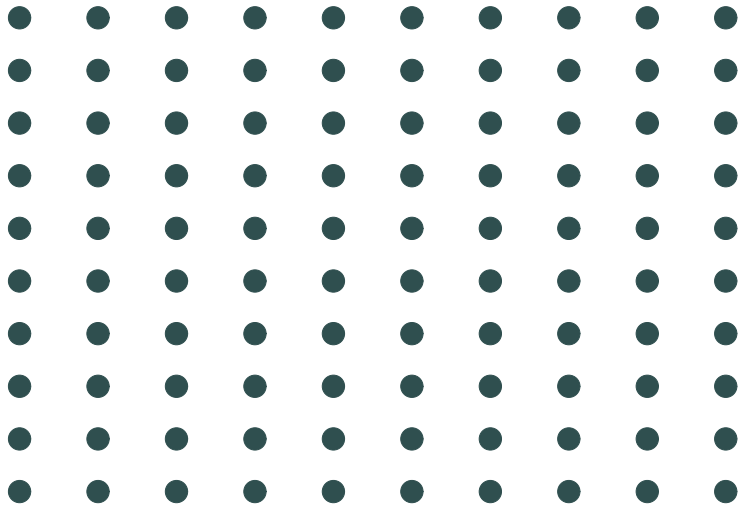
Small note: In the language of Mullainathan and Spiess (2017), \hat{y}_i is the best linear **prediction** [ML], while $\hat{\beta}_1$ is the best estimation of the **causal effect**

Population vs. sample

Question: Why do we care about *population vs. sample*?

Population vs. sample

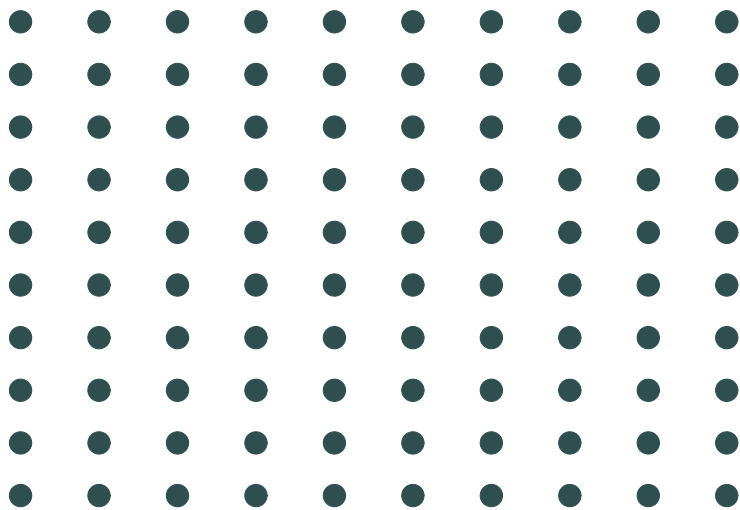
Question: Why do we care about *population vs. sample*?



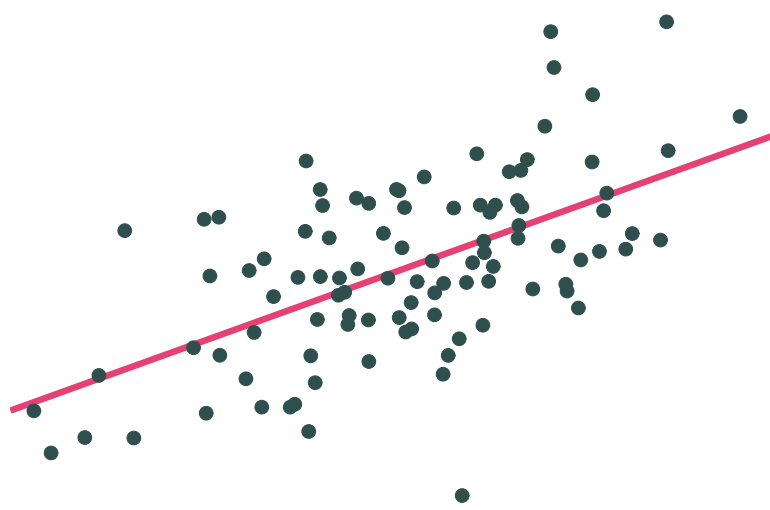
Population

Population vs. sample

Question: Why do we care about *population vs. sample*?



Population



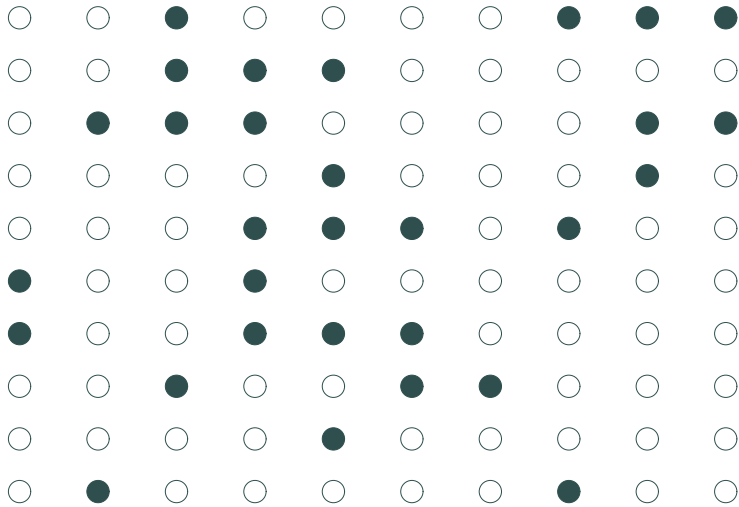
Population relationship

$$y_i = 2.53 + 0.57x_i + u_i$$

$$y_i = \beta_0 + \beta_1 x_i + u_i$$

Population vs. sample

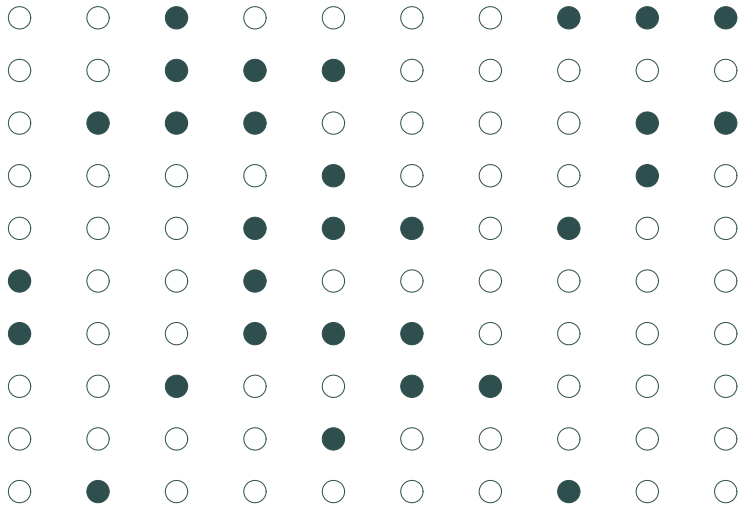
Question: Why do we care about *population vs. sample*?



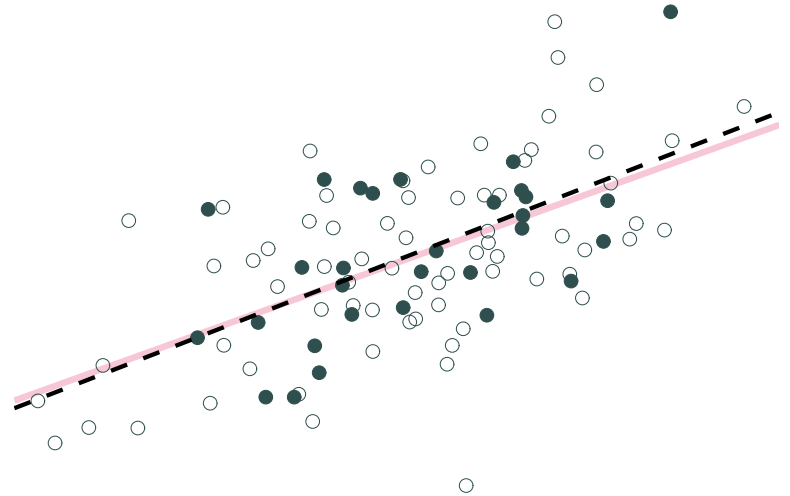
Sample 1: 30 random individuals

Population vs. sample

Question: Why do we care about *population vs. sample*?



Sample 1: 30 random individuals



Population relationship

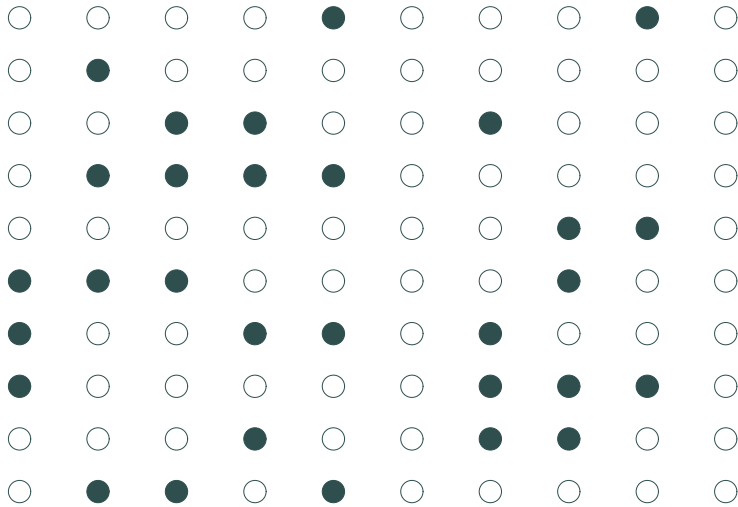
$$y_i = 2.53 + 0.57x_i + u_i$$

Sample relationship

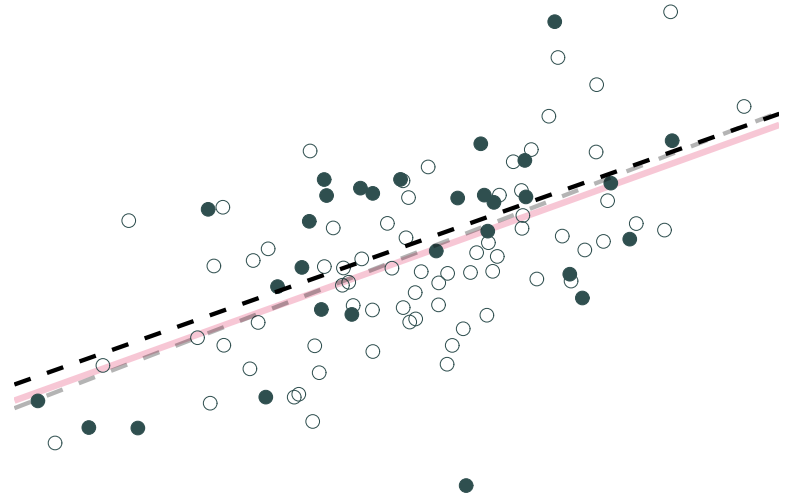
$$\hat{y}_i = 2.36 + 0.61x_i$$

Population vs. sample

Question: Why do we care about *population vs. sample*?



Sample 2: 30 random individuals



Population relationship

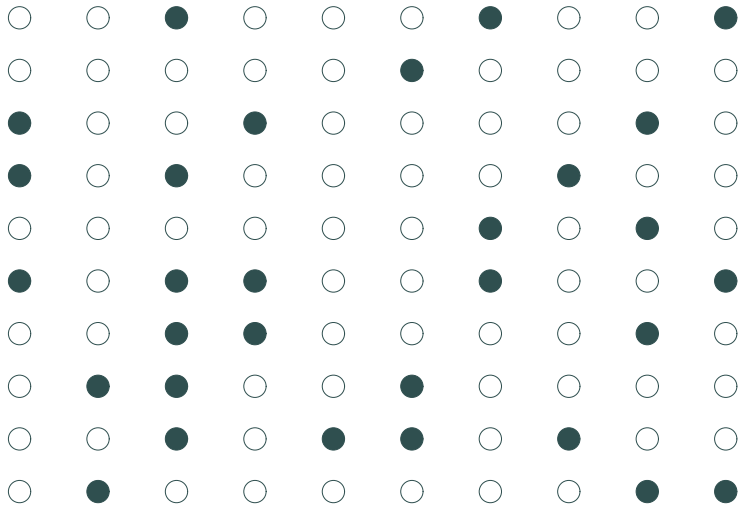
$$y_i = 2.53 + 0.57x_i + u_i$$

Sample relationship

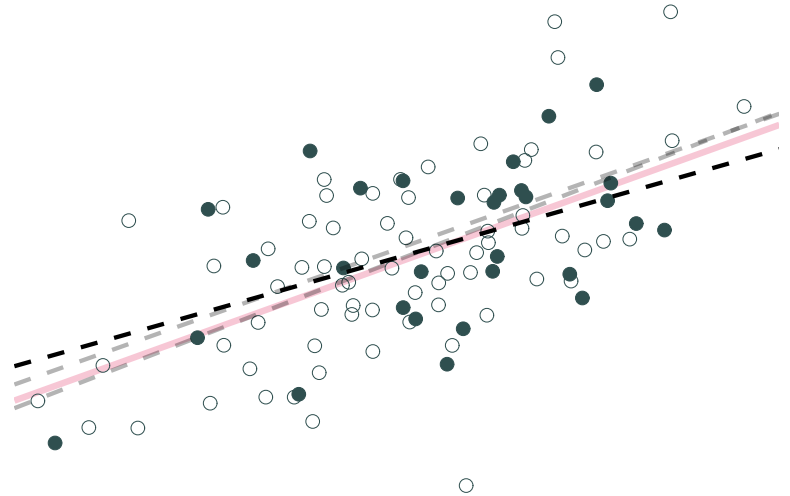
$$\hat{y}_i = 2.79 + 0.56x_i$$

Population vs. sample

Question: Why do we care about *population vs. sample*?



Sample 3: 30 random individuals



Population relationship

$$y_i = 2.53 + 0.57x_i + u_i$$

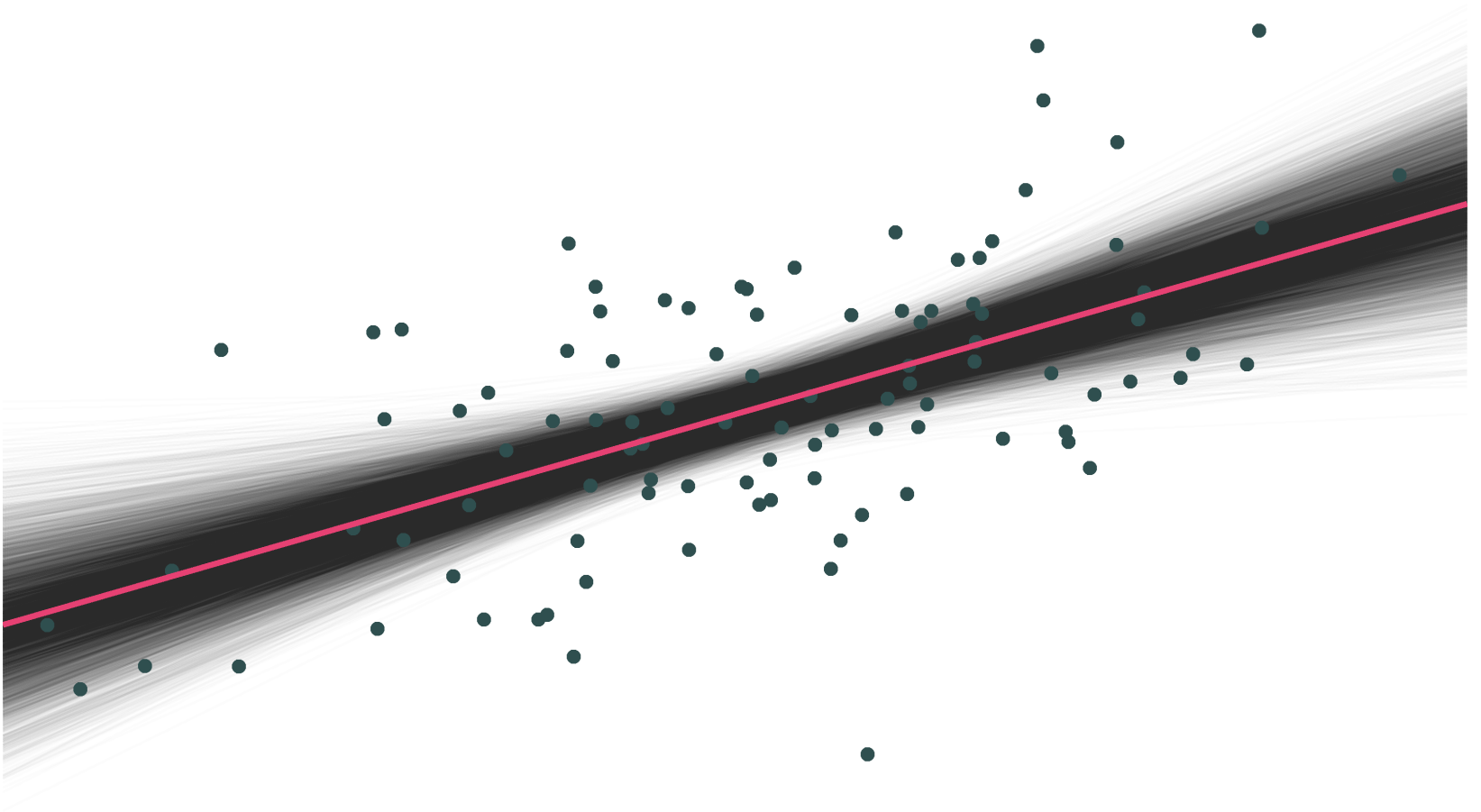
Sample relationship

$$\hat{y}_i = 3.21 + 0.45x_i$$

Let's repeat this **10,000 times**.

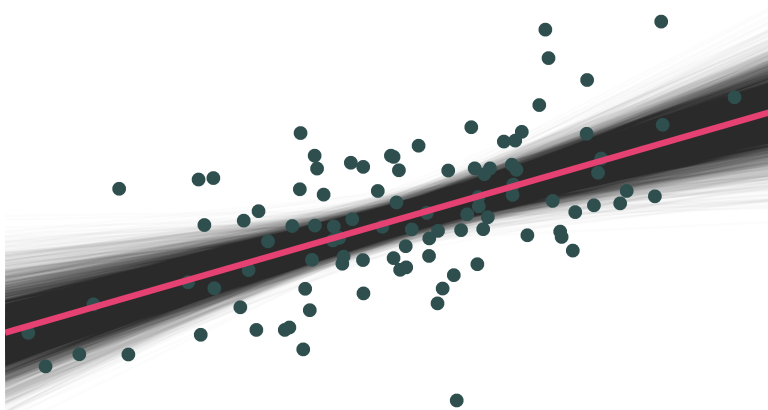
(This exercise is called a (Monte Carlo) simulation.)

Population vs. sample



Population vs. sample

Question: Why do we care about *population vs. sample*?



- On **average**, our regression lines match the population line very nicely.
- However, **individual lines** (samples) can really miss the mark.
- Differences between individual samples and the population lead to **uncertainty** for the econometrician.

Population vs. sample

Question: Why do we care about *population vs. sample*?

Question: Why do we care about *population vs. sample*?

Population vs. sample

Question: Why do we care about *population vs. sample*?

Question: Why do we care about *population vs. sample*?

Answer: Uncertainty matters.

$\hat{\beta}$ itself is a random variable—dependent upon the random sample. When we take a sample and run a regression, we don't know if it's a 'good' sample ($\hat{\beta}$ is close to β) or a 'bad sample' (our sample differs greatly from the population).

Population vs. sample

Uncertainty

Keeping track of this uncertainty is a key concept in econometrics and causal inference.

- Estimating standard errors for our estimates.
- Testing hypotheses.
- Correcting for heteroskedasticity and autocorrelation.

Population vs. sample

Uncertainty

Keeping track of this uncertainty is a key concept in econometrics and causal inference.

- Estimating standard errors for our estimates.
- Testing hypotheses.
- Correcting for heteroskedasticity and autocorrelation.

First, let's refresh on how we get these (uncertain) regression estimates.

Linear regression

The estimator

We can estimate a regression line in R (`lm(y ~ x, my_data)`). But where do these estimates come from?

A few slides back:

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$$

which gives us the *best-fit* line through our dataset.

But what do we mean by "best-fit line"?

Being the "best"

Question: What do we mean by *best-fit line*?

Answers:

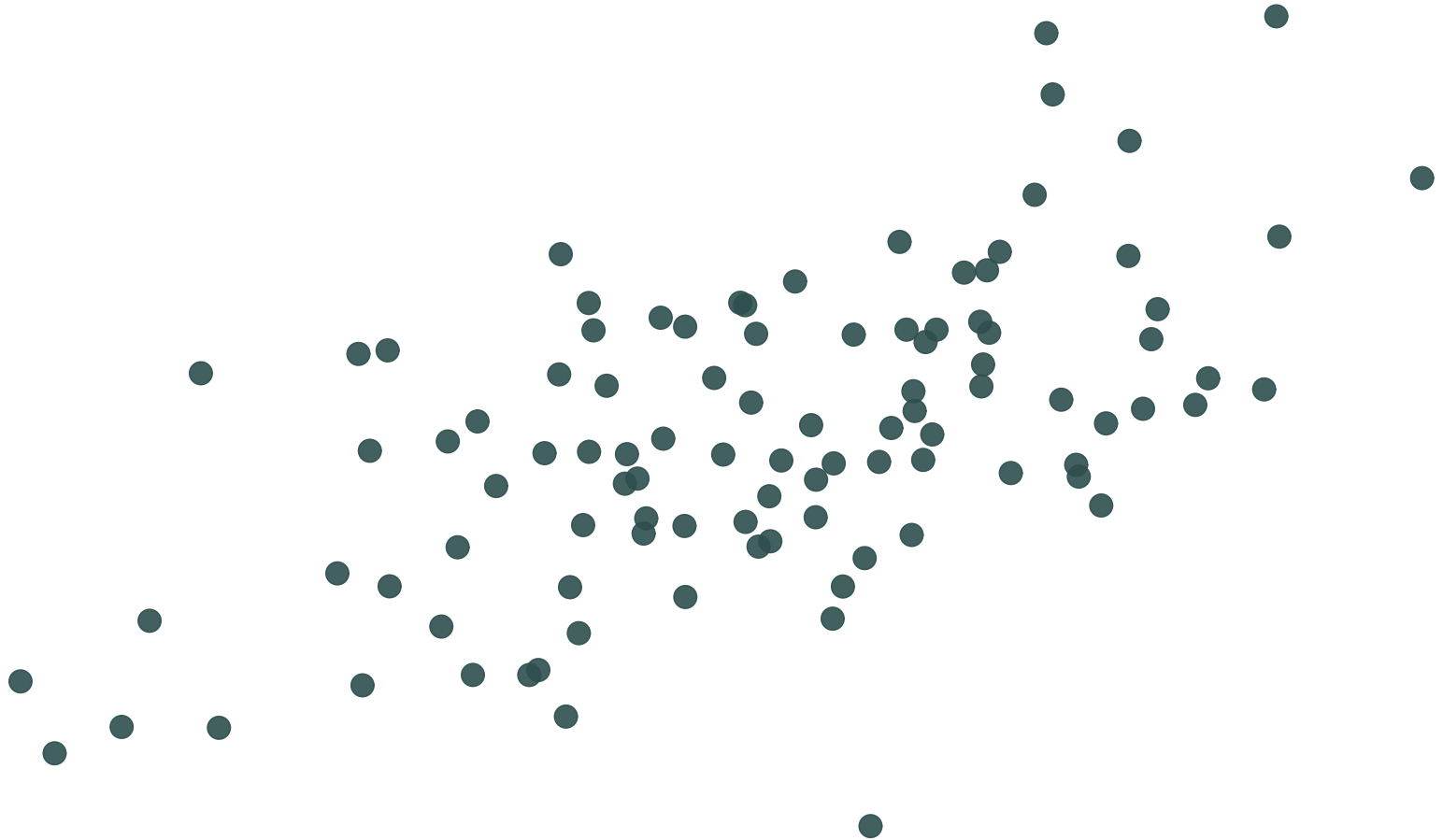
- In general, *best-fit line* means the line that minimizes the sum of squared errors (SSE):

$$\text{SSE} = \sum_{i=1}^n e_i^2 \quad \text{where} \quad e_i = y_i - \hat{y}_i$$

- Ordinary **least squares (OLS)** minimizes the sum of the squared errors.
- Based upon a set of (mostly palatable) assumptions, OLS
 - Is unbiased (and consistent)
 - Is the *best* (minimum variance) linear unbiased estimator (BLUE)

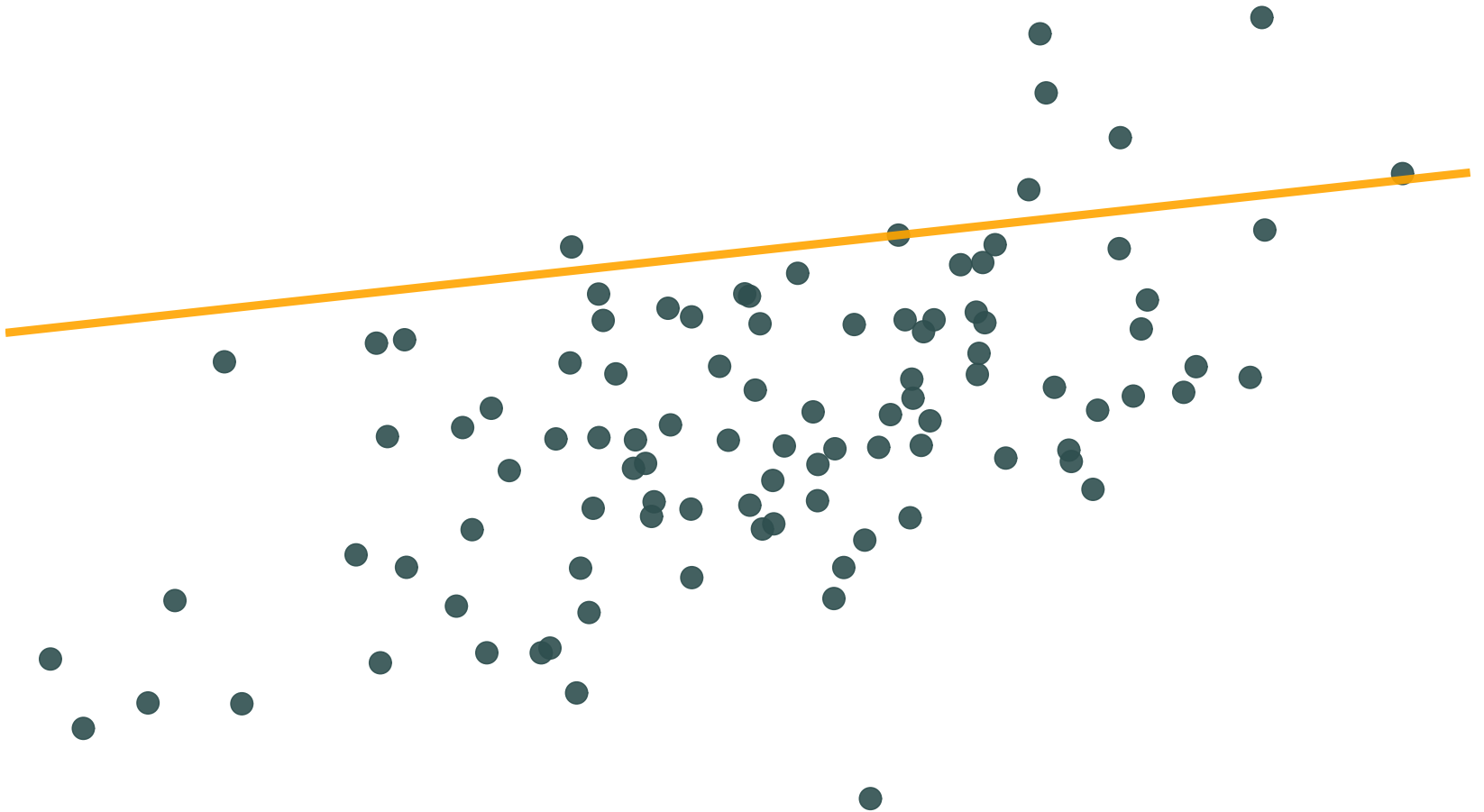
OLS vs. other lines/estimators

Let's consider the dataset we previously generated.



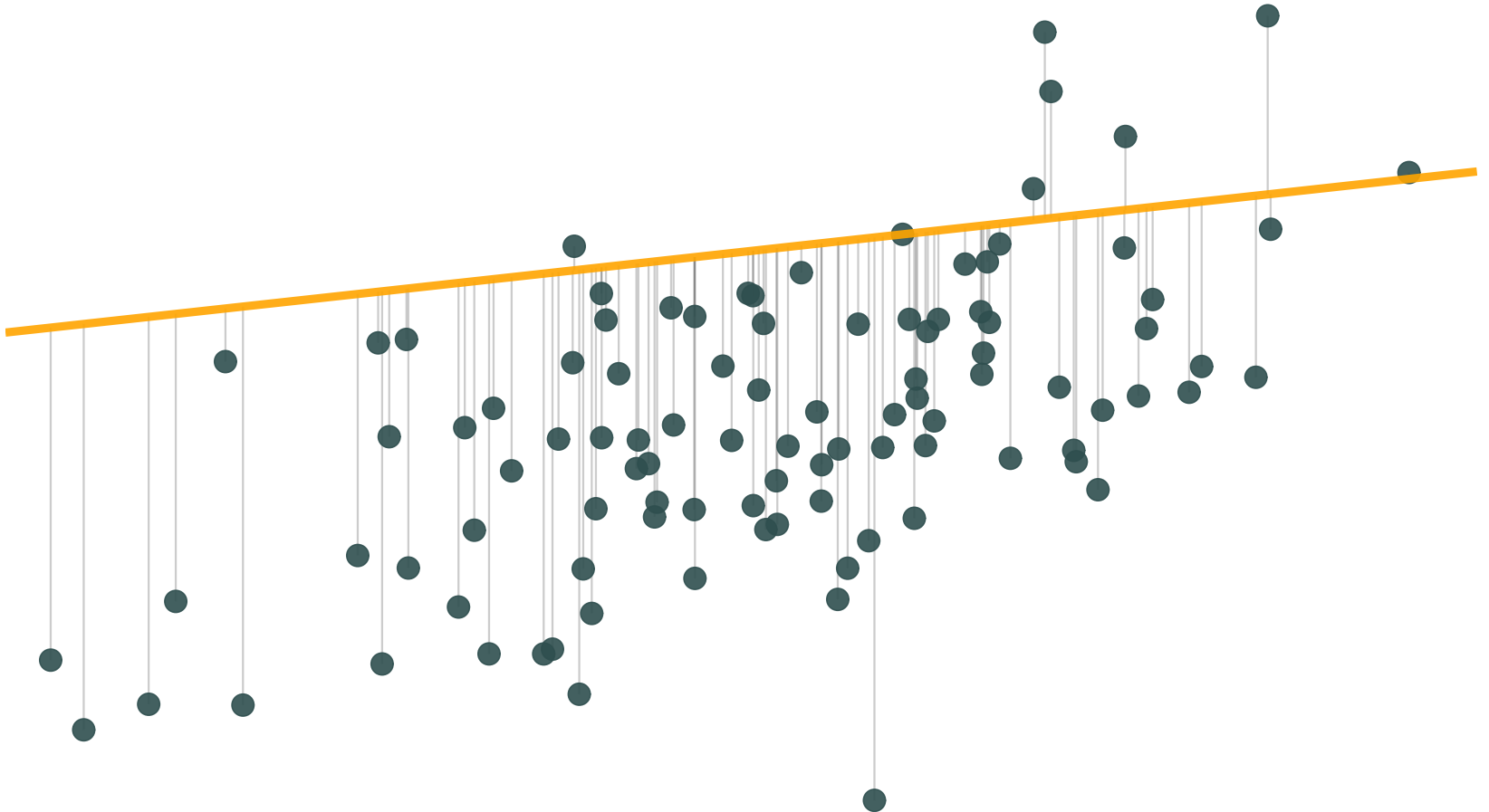
OLS vs. other lines/estimators

For any line $(\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x)$



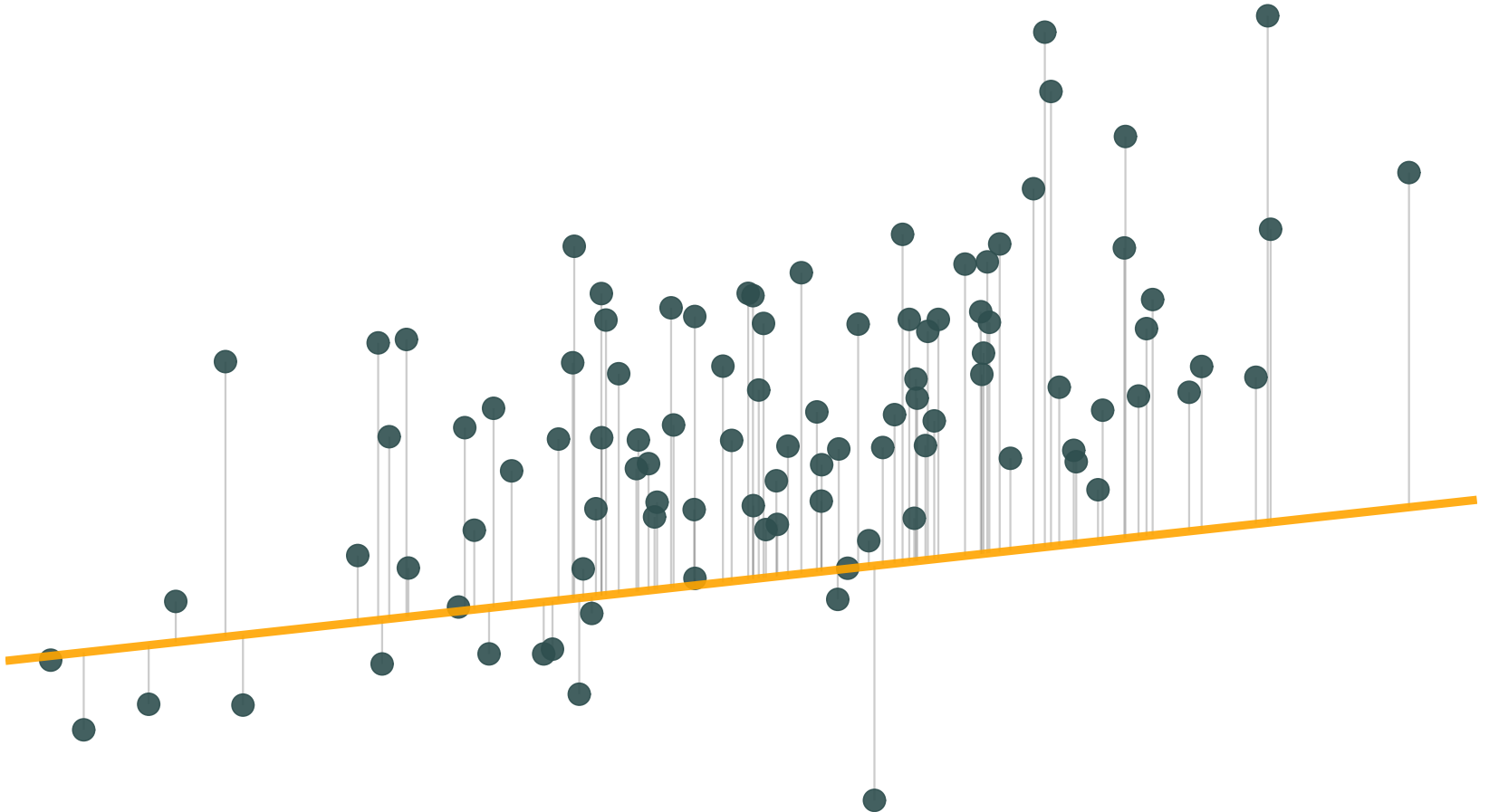
OLS vs. other lines/estimators

For any line $(\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x)$, we can calculate errors: $e_i = y_i - \hat{y}_i$



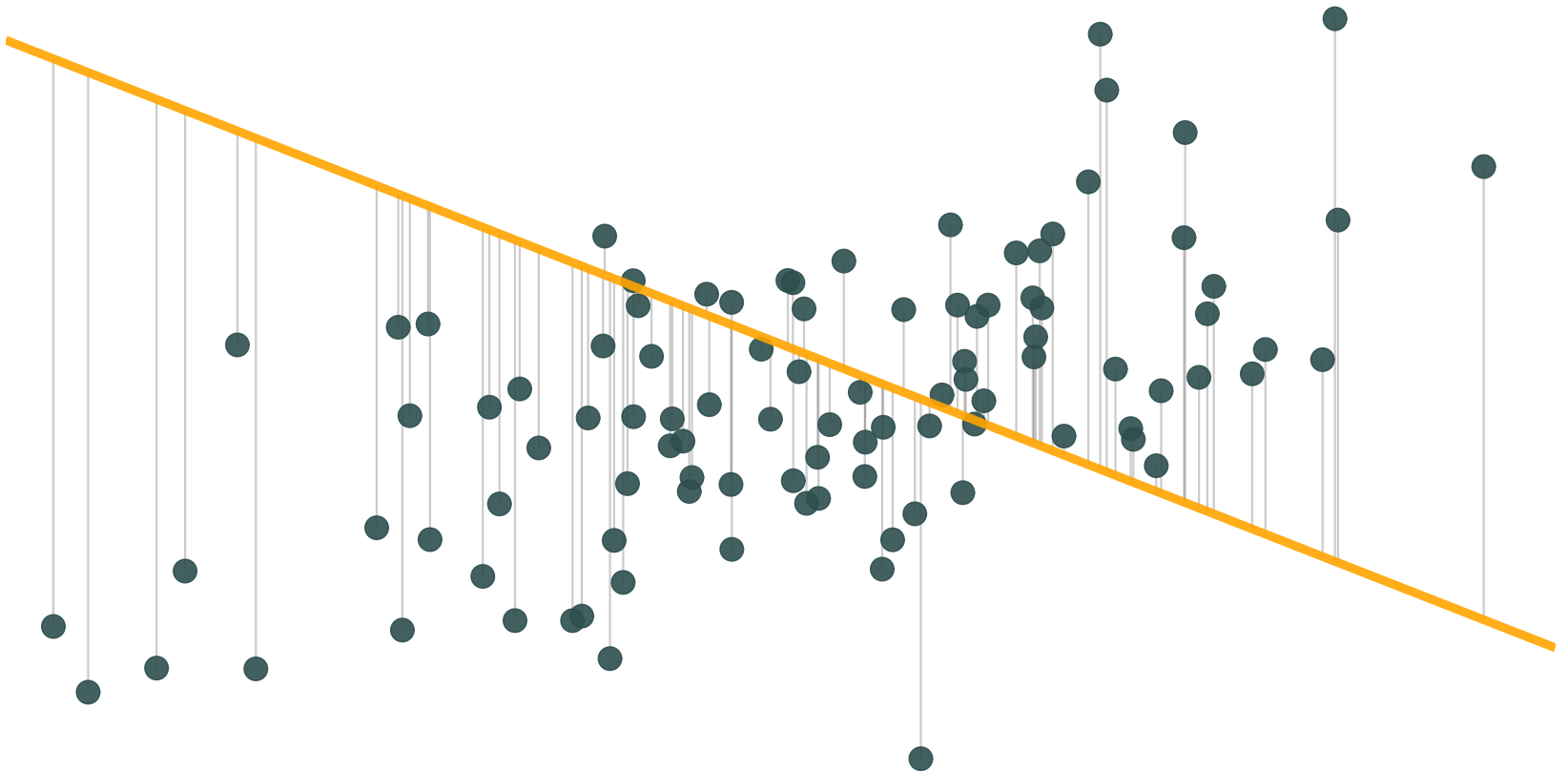
OLS vs. other lines/estimators

For any line $(\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x)$, we can calculate errors: $e_i = y_i - \hat{y}_i$



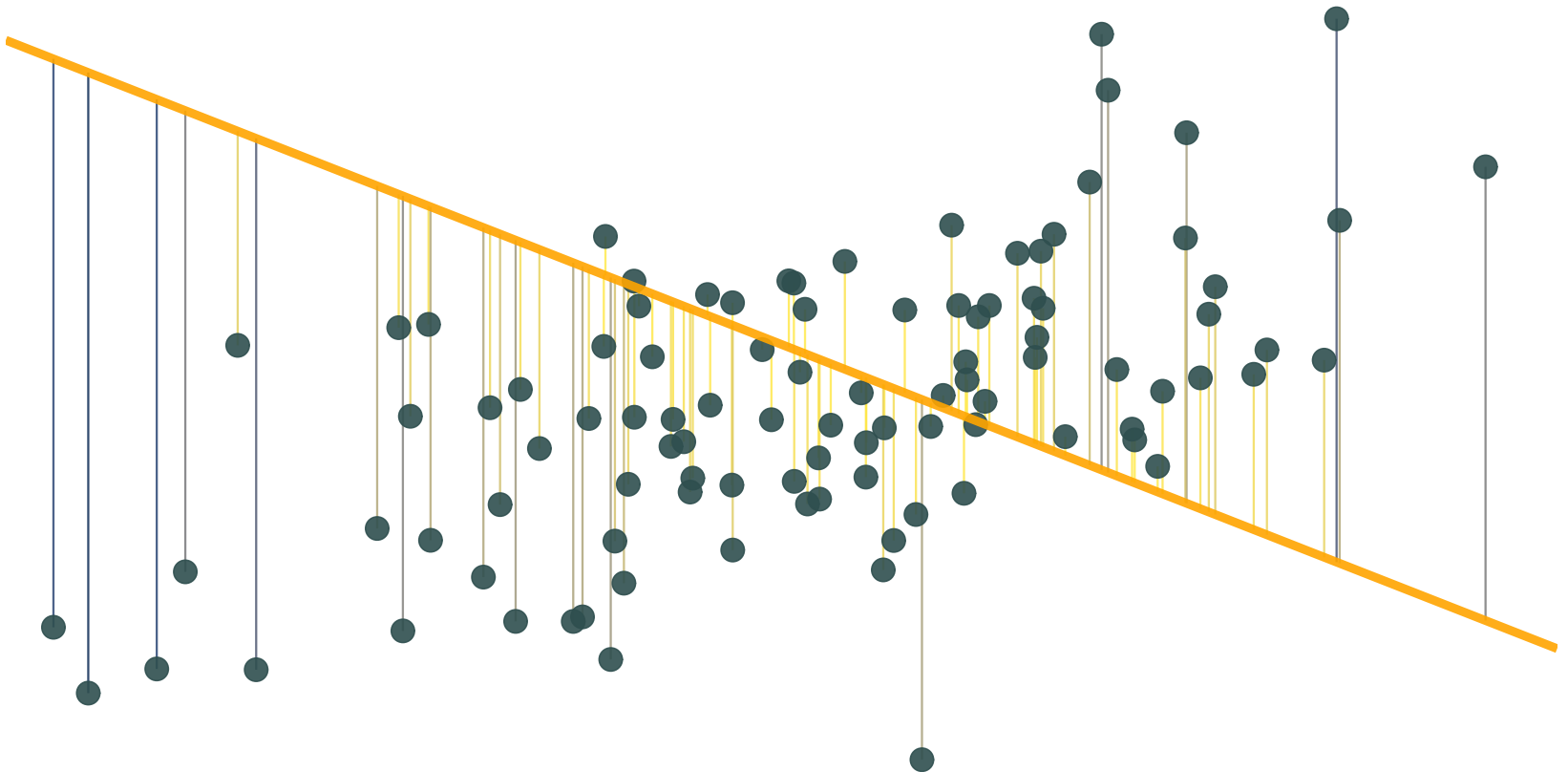
OLS vs. other lines/estimators

For any line $(\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x)$, we can calculate errors: $e_i = y_i - \hat{y}_i$



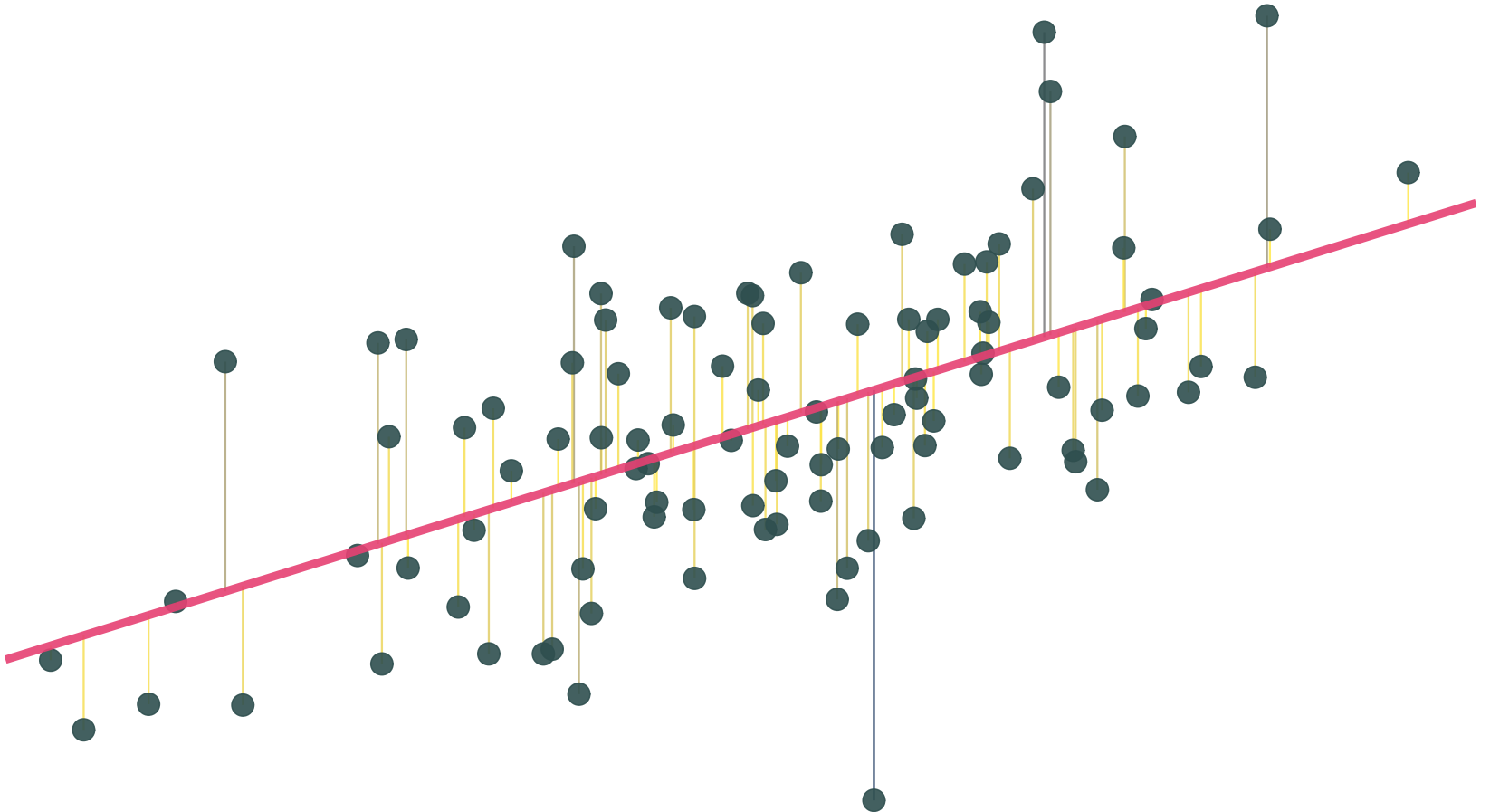
OLS vs. other lines/estimators

SSE squares the errors ($\sum e_i^2$): bigger errors get bigger penalties.



OLS vs. other lines/estimators

The OLS estimate is the combination of $\hat{\beta}_0$ and $\hat{\beta}_1$ that minimize SSE.



Formally

In simple linear regression, the OLS estimator comes from choosing the $\hat{\beta}_0$ and $\hat{\beta}_1$ that minimize the sum of squared errors (SSE), *i.e.*,

$$\min_{\hat{\beta}_0, \hat{\beta}_1} \text{SSE}$$

Formally

In simple linear regression, the OLS estimator comes from choosing the $\hat{\beta}_0$ and $\hat{\beta}_1$ that minimize the sum of squared errors (SSE), *i.e.*,

$$\min_{\hat{\beta}_0, \hat{\beta}_1} \text{SSE}$$

but we already know $\text{SSE} = \sum_i e_i^2$. Now use the definitions of e_i and \hat{y} .

$$\begin{aligned} e_i^2 &= (y_i - \hat{y}_i)^2 = (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i)^2 \\ &= y_i^2 - 2y_i\hat{\beta}_0 - 2y_i\hat{\beta}_1 x_i + \hat{\beta}_0^2 + 2\hat{\beta}_0\hat{\beta}_1 x_i + \hat{\beta}_1^2 x_i^2 \end{aligned}$$

Formally

In simple linear regression, the OLS estimator comes from choosing the $\hat{\beta}_0$ and $\hat{\beta}_1$ that minimize the sum of squared errors (SSE), *i.e.*,

$$\min_{\hat{\beta}_0, \hat{\beta}_1} \text{SSE}$$

but we already know $\text{SSE} = \sum_i e_i^2$. Now use the definitions of e_i and \hat{y} .

$$\begin{aligned} e_i^2 &= (y_i - \hat{y}_i)^2 = (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i)^2 \\ &= y_i^2 - 2y_i\hat{\beta}_0 - 2y_i\hat{\beta}_1 x_i + \hat{\beta}_0^2 + 2\hat{\beta}_0\hat{\beta}_1 x_i + \hat{\beta}_1^2 x_i^2 \end{aligned}$$

Recall: Minimizing a multivariate function requires **(1)** first derivatives equal zero (the *1st-order conditions*) and **(2)** second-order conditions (concavity).

Formally

We're getting close. We need to **minimize SSE**. We've showed how SSE relates to our sample (our data: x and y) and our estimates (*i.e.*, $\hat{\beta}_0$ and $\hat{\beta}_1$).

$$\text{SSE} = \sum_i e_i^2 = \sum_i \left(y_i^2 - 2y_i\hat{\beta}_0 - 2y_i\hat{\beta}_1x_i + \hat{\beta}_0^2 + 2\hat{\beta}_0\hat{\beta}_1x_i + \hat{\beta}_1^2x_i^2 \right)$$

For the first-order conditions of minimization, we now take the first derivatives of SSE with respect to $\hat{\beta}_0$ and $\hat{\beta}_1$.

$$\begin{aligned} \frac{\partial \text{SSE}}{\partial \hat{\beta}_0} &= \sum_i \left(2\hat{\beta}_0 + 2\hat{\beta}_1x_i - 2y_i \right) = 2n\hat{\beta}_0 + 2\hat{\beta}_1 \sum_i x_i - 2 \sum_i y_i \\ &= 2n\hat{\beta}_0 + 2n\hat{\beta}_1\bar{x} - 2n\bar{y} \end{aligned}$$

where $\bar{x} = \frac{\sum x_i}{n}$ and $\bar{y} = \frac{\sum y_i}{n}$ are sample means of x and y (size n).

Formally

The first-order conditions state that the derivatives are equal to zero, so:

$$\frac{\partial \text{SSE}}{\partial \hat{\beta}_0} = 2n\hat{\beta}_0 + 2n\hat{\beta}_1\bar{x} - 2n\bar{y} = 0$$

which implies

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1\bar{x}$$

Now for $\hat{\beta}_1$.

Formally

Take the derivative of SSE with respect to $\hat{\beta}_1$

$$\begin{aligned}\frac{\partial \text{SSE}}{\partial \hat{\beta}_1} &= \sum_i \left(2\hat{\beta}_0 x_i + 2\hat{\beta}_1 x_i^2 - 2y_i x_i \right) = 2\hat{\beta}_0 \sum_i x_i + 2\hat{\beta}_1 \sum_i x_i^2 - 2 \sum_i y_i x_i \\ &= 2n\hat{\beta}_0 \bar{x} + 2\hat{\beta}_1 \sum_i x_i^2 - 2 \sum_i y_i x_i\end{aligned}$$

set it equal to zero (first-order conditions, again)

$$\frac{\partial \text{SSE}}{\partial \hat{\beta}_1} = 2n\hat{\beta}_0 \bar{x} + 2\hat{\beta}_1 \sum_i x_i^2 - 2 \sum_i y_i x_i = 0$$

and substitute in our relationship for $\hat{\beta}_0$, i.e., $\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$. Thus,

$$2n \left(\bar{y} - \hat{\beta}_1 \bar{x} \right) \bar{x} + 2\hat{\beta}_1 \sum_i x_i^2 - 2 \sum_i y_i x_i = 0$$

Formally

Continuing from the last slide

$$2n \left(\bar{y} - \hat{\beta}_1 \bar{x} \right) \bar{x} + 2\hat{\beta}_1 \sum_i x_i^2 - 2 \sum_i y_i x_i = 0$$

we multiply out

$$2n\bar{y}\bar{x} - 2n\hat{\beta}_1\bar{x}^2 + 2\hat{\beta}_1 \sum_i x_i^2 - 2 \sum_i y_i x_i = 0$$

$$\implies 2\hat{\beta}_1 \left(\sum_i x_i^2 - n\bar{x}^2 \right) = 2 \sum_i y_i x_i - 2n\bar{y}\bar{x}$$

$$\implies \hat{\beta}_1 = \frac{\sum_i y_i x_i - 2n\bar{y}\bar{x}}{\sum_i x_i^2 - n\bar{x}^2} = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sum_i (x_i - \bar{x})^2}$$

OLS

Formally


Done!

We now have (lovely) OLS estimators for the slope

$$\hat{\beta}_1 = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sum_i (x_i - \bar{x})^2} = \frac{\hat{Cov}(X, Y)}{\hat{V}(X)}$$

and the intercept

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

And now you know where the *least squares* part of ordinary least squares comes from. 

Formally

Simple linear regression estimator:

$$\hat{\beta}_1 = \frac{\hat{\text{Cov}}(x, y)}{\hat{\text{Var}}(x)}$$

moving to multiple linear regression, the estimator changes slightly:

$$\hat{\beta}_1 = \frac{\hat{\text{Cov}}(\tilde{x}_1, y)}{\hat{\text{Var}}(\tilde{x}_1)}$$

where \tilde{x}_1 is the *residualized* x_1 variable—the variation remaining in x after controlling for the other explanatory variables.

Formally

More formally, consider the multiple-regression model

$$y_i = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + u_i$$

Our residualized x_1 (which we named \tilde{x}_1) comes from regressing x_1 on an intercept and all of the other explanatory variables and collecting the residuals, *i.e.*,

$$\hat{x}_{1i} = \hat{\gamma}_0 + \hat{\gamma}_2 x_{2i} + \hat{\gamma}_3 x_{3i}$$

$$\tilde{x}_{1i} = x_{1i} - \hat{x}_{1i}$$

Formally

More formally, consider the multiple-regression model

$$y_i = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + u_i$$

Our residualized x_1 (which we named \tilde{x}_1) comes from regressing x_1 on an intercept and all of the other explanatory variables and collecting the residuals, *i.e.*,

$$\hat{x}_{1i} = \hat{\gamma}_0 + \hat{\gamma}_2 x_{2i} + \hat{\gamma}_3 x_{3i}$$

$$\tilde{x}_{1i} = x_{1i} - \hat{x}_{1i}$$

allowing us to better understand our OLS multiple-regression estimator

$$\hat{\beta}_1 = \frac{\hat{\text{Cov}}(\tilde{x}_1, y)}{\hat{\text{Var}}(\tilde{x}_1)}$$

Formally

More formally, consider the multiple-regression model

$$y_i = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + u_i$$

Our residualized x_1 (which we named \tilde{x}_1) comes from regressing x_1 on an intercept and all of the other explanatory variables and collecting the residuals, *i.e.*,

$$\hat{x}_{1i} = \hat{\gamma}_0 + \hat{\gamma}_2 x_{2i} + \hat{\gamma}_3 x_{3i}$$

$$\tilde{x}_{1i} = x_{1i} - \hat{x}_{1i}$$

allowing us to better understand our OLS multiple-regression estimator

$$\hat{\beta}_1 = \frac{\hat{\text{Cov}}(\tilde{x}_1, y)}{\hat{\text{Var}}(\tilde{x}_1)}$$

OLS: Assumptions and properties

OLS: Assumptions and properties

Properties

Question: What properties might we care about for an estimator?

OLS: Assumptions and properties

Properties

Question: What properties might we care about for an estimator?

Tangent: Let's review statistical properties first.

OLS: Assumptions and properties

Properties

Refresher: Density functions

Recall that we use **probability density functions** (PDFs) to describe the probability a **continuous random variable** takes on a range of values. (The total area = 1.)

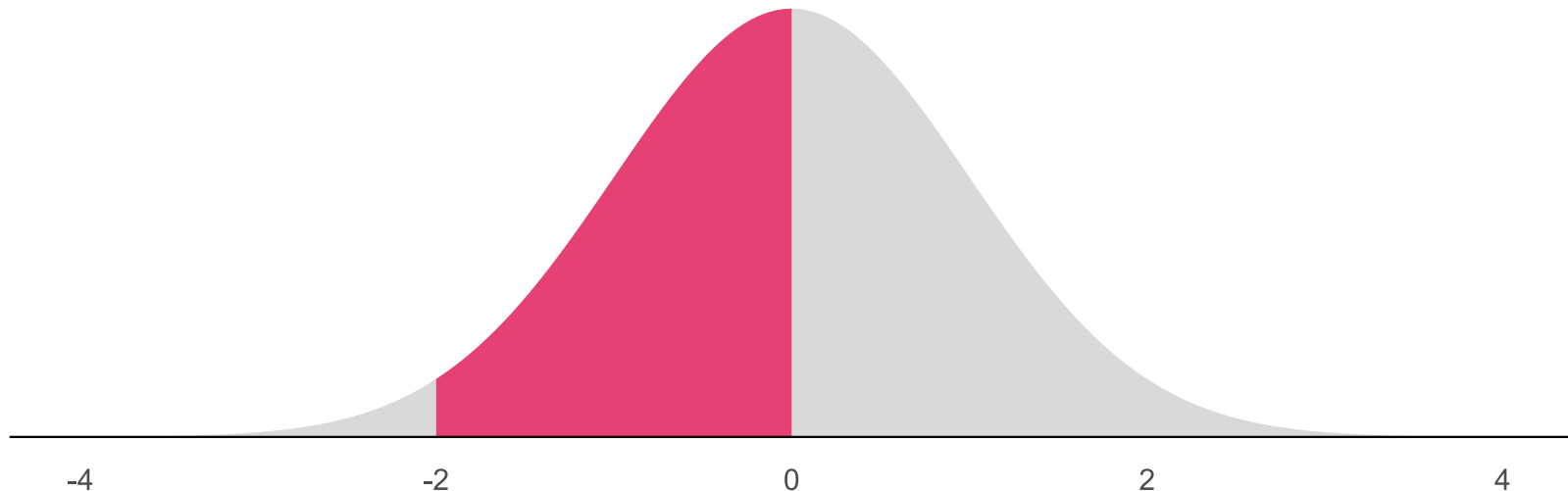
These PDFs characterize probability distributions, and the most common/famous/popular distributions get names (*e.g.*, normal, *t*, Gamma).

OLS: Assumptions and properties

Properties

Refresher: Density functions

The probability a standard normal random variable takes on a value between -2 and 0: $P(-2 \leq X \leq 0) = 0.48$

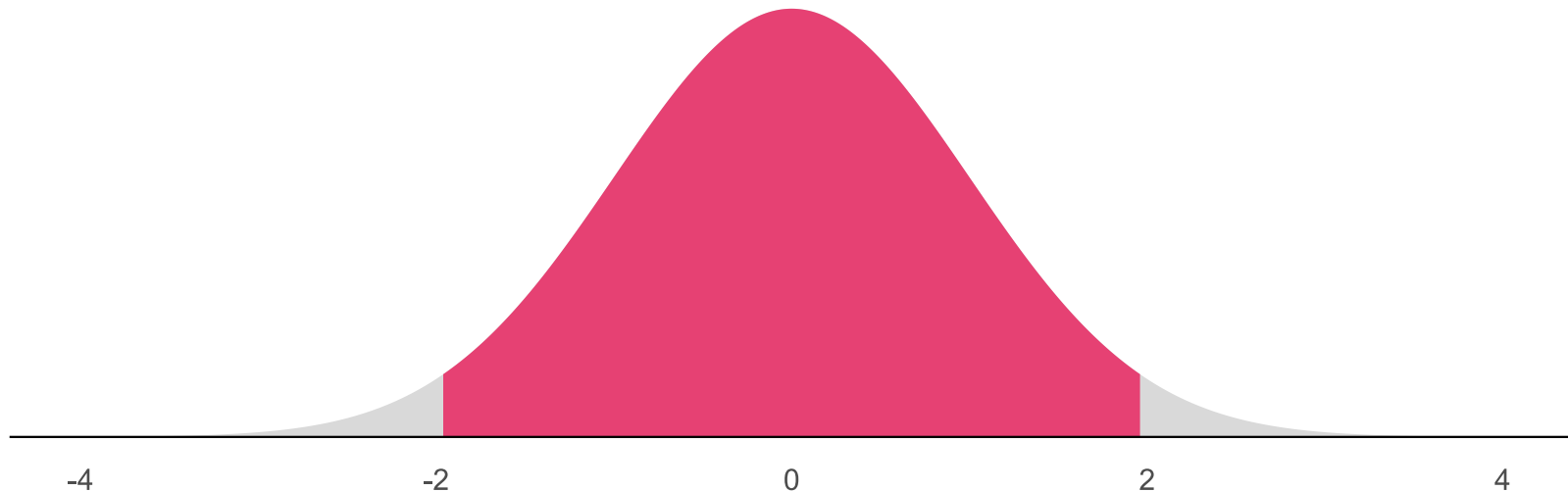


OLS: Assumptions and properties

Properties

Refresher: Density functions

The probability a standard normal random variable takes on a value between -1.96 and 1.96: $P(-1.96 \leq X \leq 1.96) = 0.95$

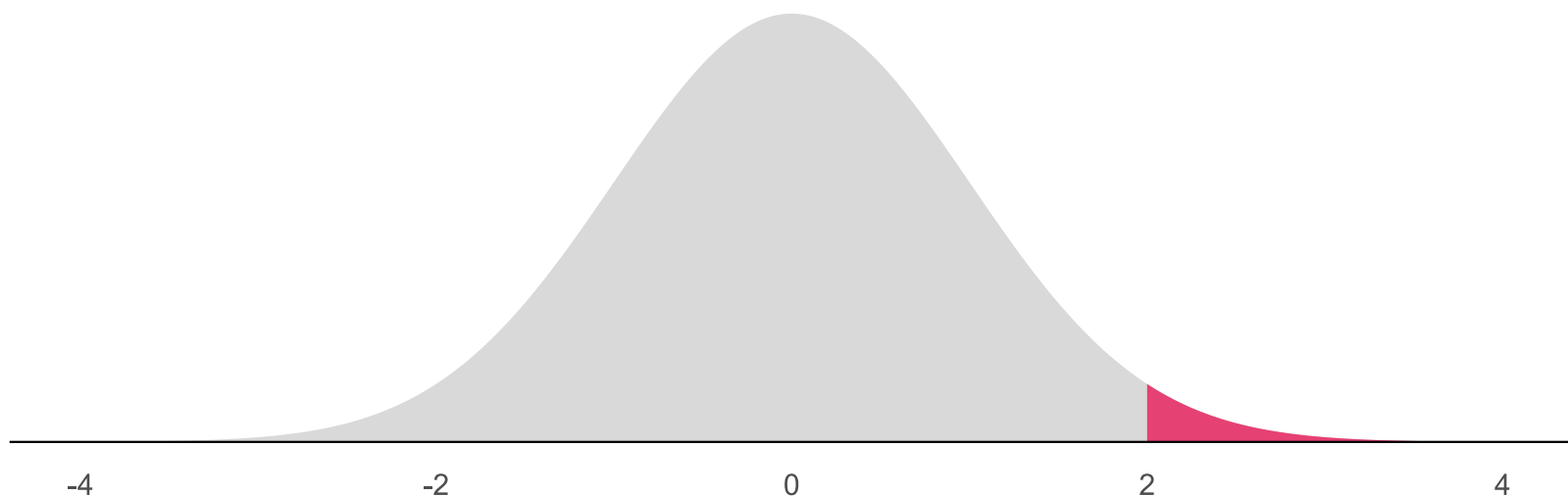


OLS: Assumptions and properties

Properties

Refresher: Density functions

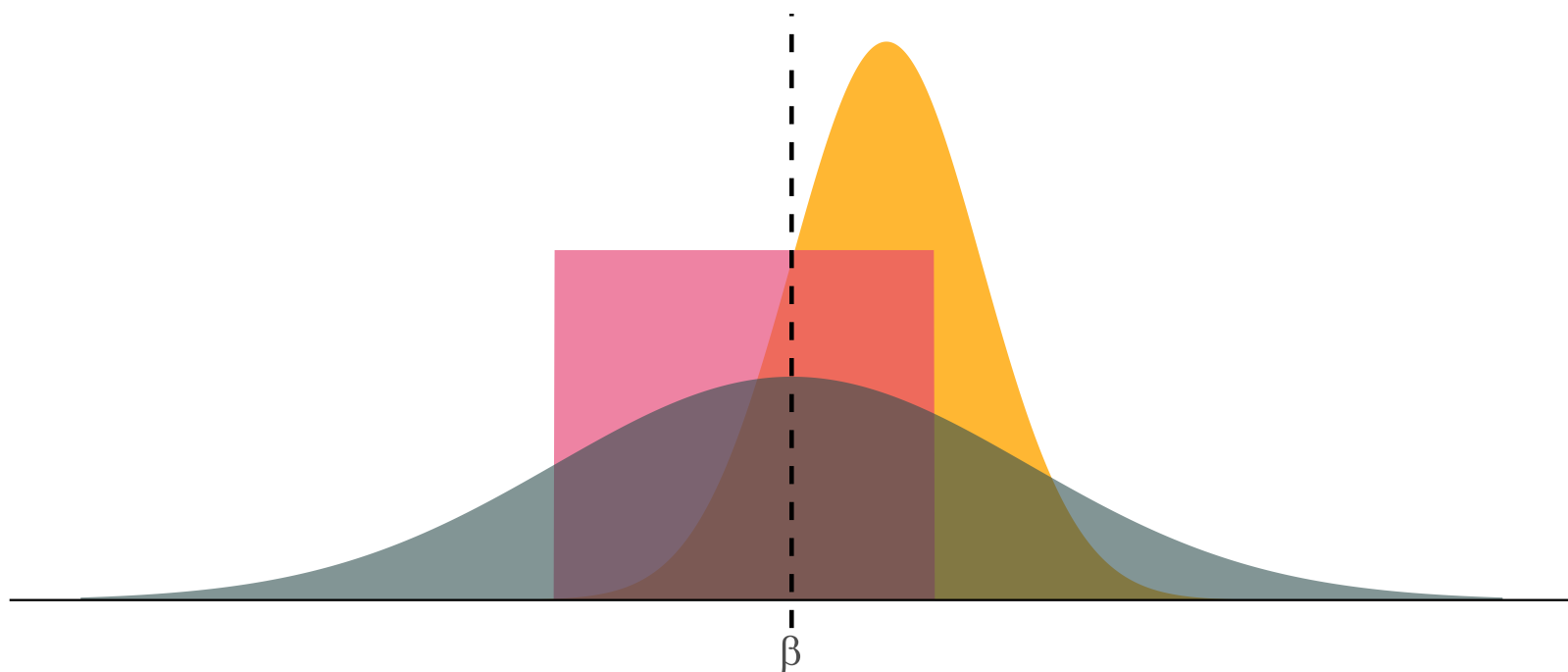
The probability a standard normal random variable takes on a value beyond 2: $P(X > 2) = 0.023$



OLS: Assumptions and properties

Properties

Imagine we are trying to estimate an unknown parameter β , and we know the distributions of three competing estimators. Which one would we want? How would we decide?



OLS: Assumptions and properties

Properties

Question: What properties might we care about for an estimator?

OLS: Assumptions and properties

Properties

Question: What properties might we care about for an estimator?

Answer one: Bias.

On average (after *many* samples), does the estimator tend toward the correct value?

More formally: Does the mean of estimator's distribution equal the parameter it estimates?

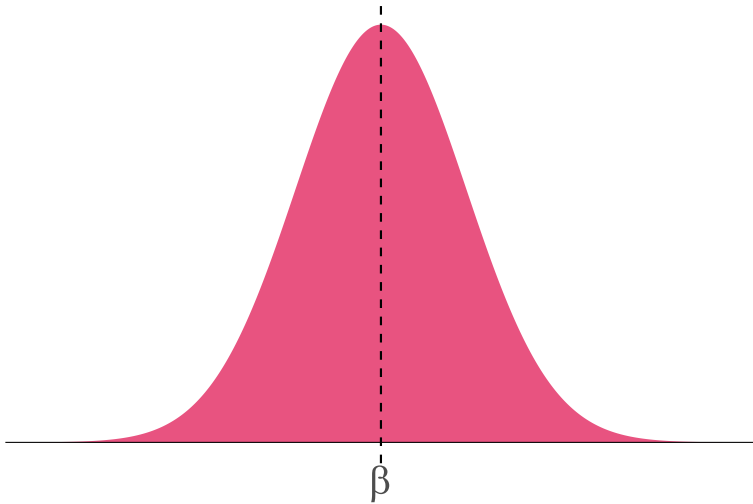
$$\text{Bias}_{\beta}(\hat{\beta}) = \mathbf{E}[\hat{\beta}] - \beta$$

OLS: Assumptions and properties

Properties

Answer one: Bias.

Unbiased estimator: $E[\hat{\beta}] = \beta$

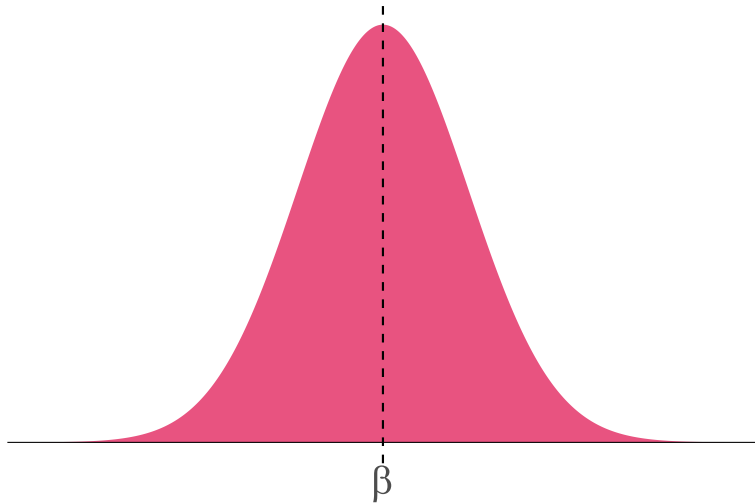


OLS: Assumptions and properties

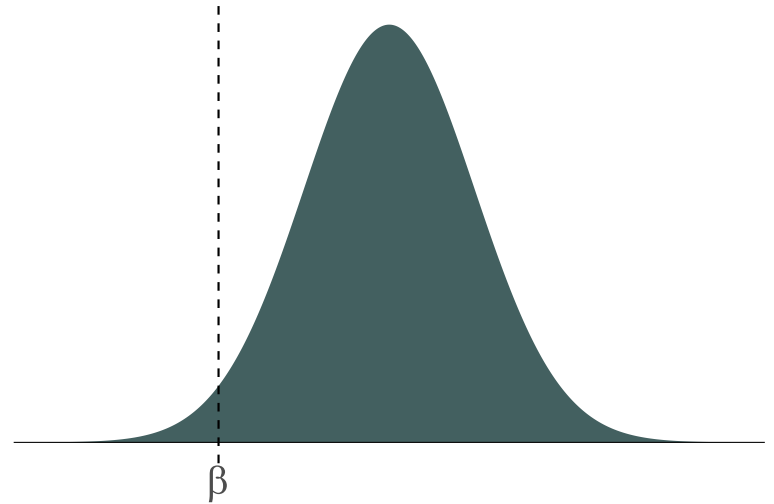
Properties

Answer one: Bias.

Unbiased estimator: $E[\hat{\beta}] = \beta$



Biased estimator: $E[\hat{\beta}] \neq \beta$



OLS: Assumptions and properties

Properties

Answer two: Variance.

The central tendencies (means) of competing distributions are not the only things that matter. We also care about the **variance** of an estimator.

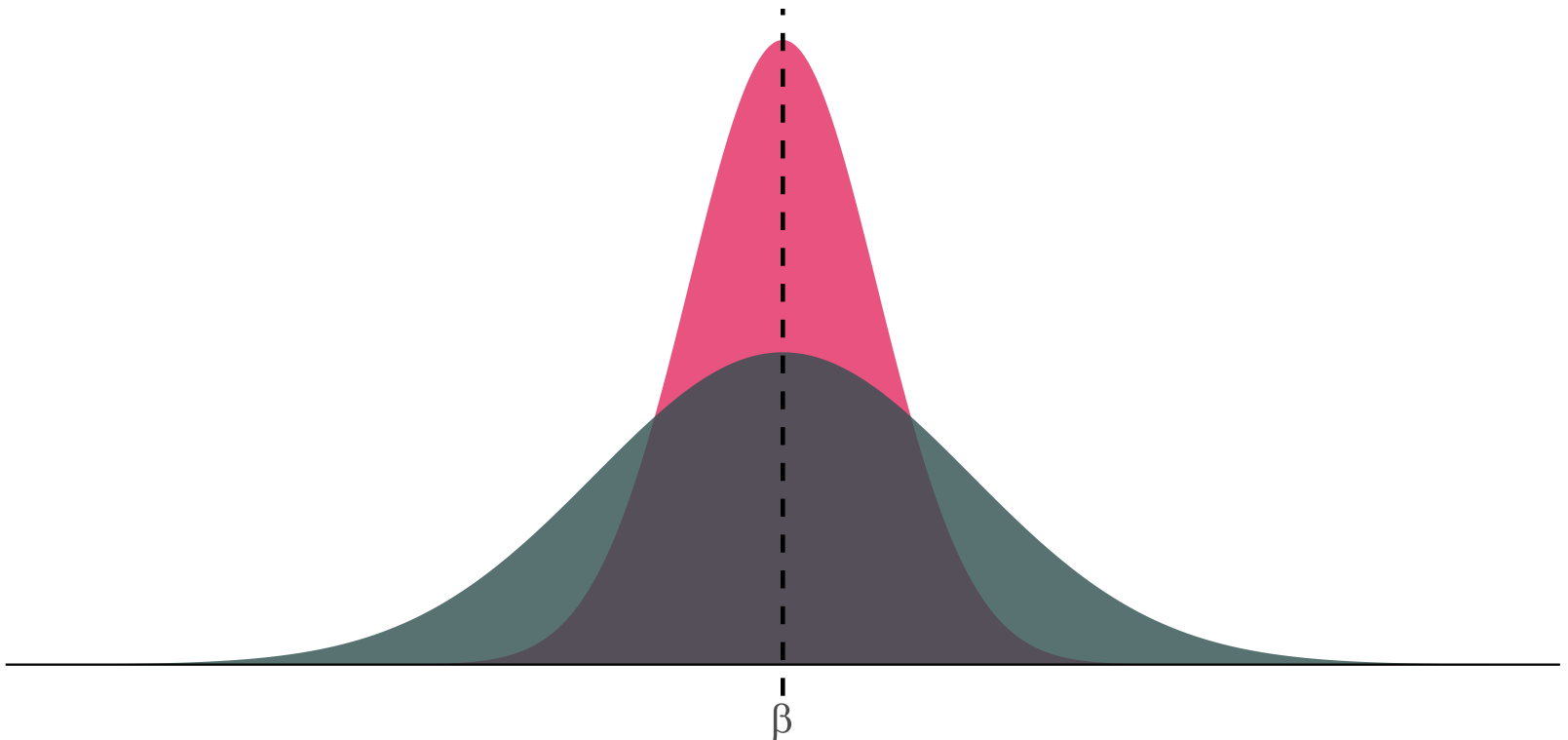
$$\text{Var}(\hat{\beta}) = \mathbf{E} \left[\left(\hat{\beta} - \mathbf{E}[\hat{\beta}] \right)^2 \right]$$

Lower variance estimators mean we get estimates closer to the mean in each sample.

OLS: Assumptions and properties

Properties

Answer two: Variance.



OLS: Assumptions and properties

Properties

Answer one: Bias.

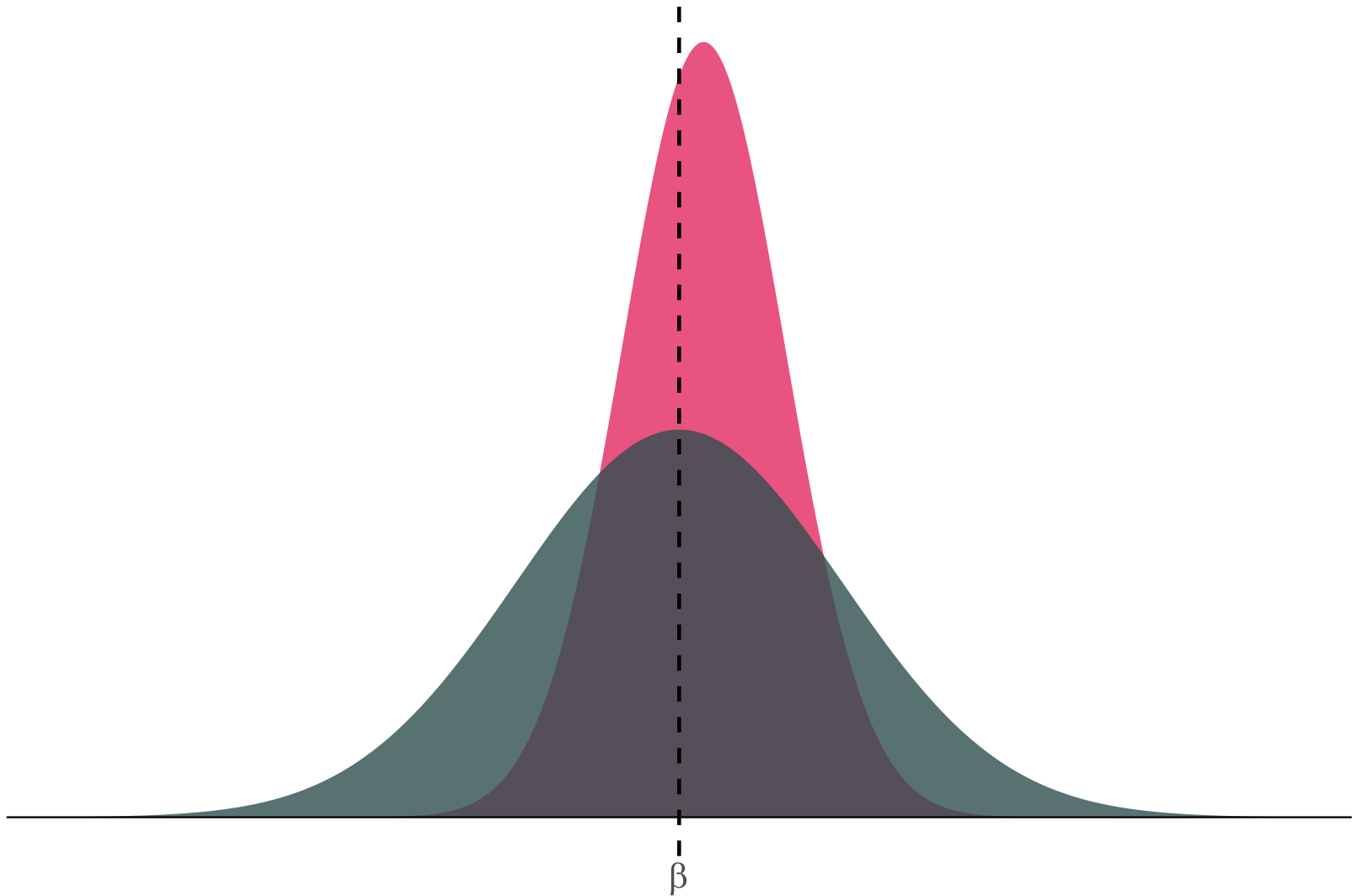
Answer two: Variance.

Subtlety: The bias-variance tradeoff.

Should we be willing to take a bit of bias to reduce the variance?

In econometrics, we generally stick with unbiased (or consistent) estimators. But other disciplines (especially computer science) think a bit more about this tradeoff.

The bias-variance tradeoff.



OLS: Assumptions and properties

Properties

As you might have guessed by now,

- OLS is **unbiased**.
- OLS has the **minimum variance** of all unbiased linear estimators.

OLS: Assumptions and properties

Properties

But... these (very nice) properties depend upon a set of assumptions:

1. The population relationship is linear in parameters with an additive disturbance.
2. Our X variable is **exogenous**, i.e., $\mathbf{E}[u \mid X] = 0$.
3. The X variable has variation. And if there are multiple explanatory variables, they are not perfectly collinear.
4. The population disturbances u_i are independently and identically distributed as normal random variables with mean zero ($\mathbf{E}[u] = 0$) and variance σ^2 (i.e., $\mathbf{E}[u^2] = \sigma^2$). Independently distributed and mean zero jointly imply $\mathbf{E}[u_i u_j] = 0$ for any $i \neq j$.

OLS: Assumptions and properties

Assumptions

Different assumptions guarantee different properties:

- Assumptions (1), (2), and (3) make OLS unbiased.
- Assumption (4) gives us an unbiased estimator for the variance of our OLS estimator.

During our course, we will discuss the many ways real life may **violate these assumptions**. For instance:

- Non-linear relationships in our parameters/disturbances (or misspecification).
- Disturbances that are not identically distributed and/or not independent.
- Violations of exogeneity (especially omitted-variable bias).

OLS: Assumptions and properties

Conditional expectation

For many applications, our most important assumption is **exogeneity**, *i.e.*,

$$E[u \mid X] = 0$$

but what does it actually mean?

OLS: Assumptions and properties

Conditional expectation

For many applications, our most important assumption is **exogeneity**, *i.e.*,

$$E[u \mid X] = 0$$

but what does it actually mean?

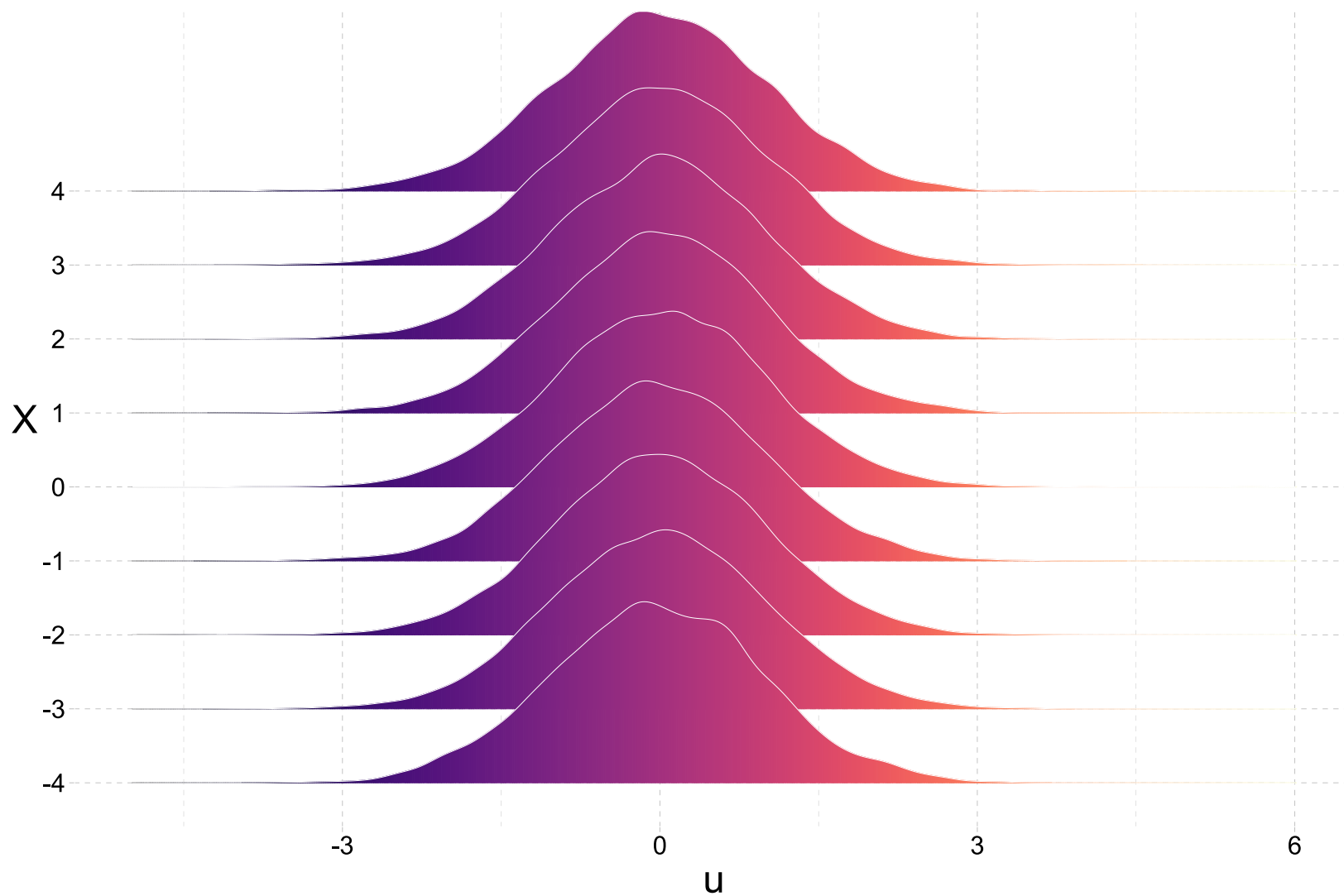
One way to think about this definition:

For *any* value of X , the mean of the residuals must be zero.

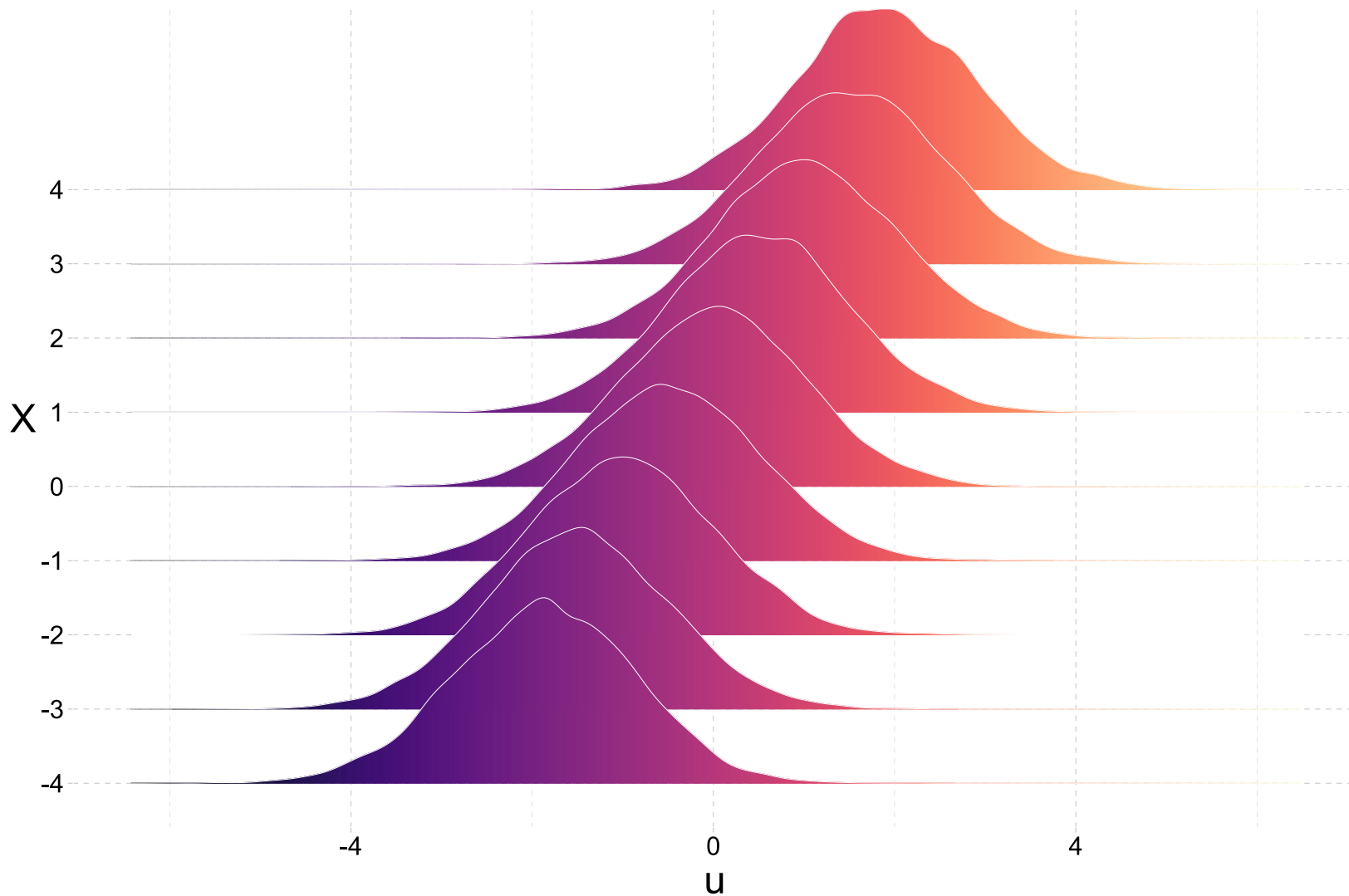
- *E.g.*, $E[u \mid X = 1] = 0$ and $E[u \mid X = 100] = 0$
- *E.g.*, $E[u \mid X_2 = \text{Female}] = 0$ and $E[u \mid X_2 = \text{Male}] = 0$
- Notice: $E[u \mid X] = 0$ is more restrictive than $E[u] = 0$

Graphically...

Valid exogeneity, *i.e.*, $E[u \mid X] = 0$



Invalid exogeneity, i.e., $E[u | X] \neq 0$



Uncertainty and inference

Uncertainty and inference

Is there more?

Up to this point, we know OLS has some nice properties, and we know how to estimate an intercept and slope coefficient via OLS.

Our current workflow:

- Get data (points with x and y values)
- Regress y on x
- Plot the OLS line (i.e., $\hat{y} = \hat{\beta}_0 + \hat{\beta}_1$)
- Done?

But how do we actually **learn** something from this exercise?

Uncertainty and inference

There is more

But how do we actually **learn** something from this exercise?

- Based upon our value of $\hat{\beta}_1$, can we rule out previously hypothesized values?
- How confident should we be in the precision of our estimates?
- How well does our model explain the variation we observe in y ?

We need to be able to deal with uncertainty. Enter: **Inference**.

Uncertainty and inference

Learning from our errors

As our previous simulation pointed out, our problem with **uncertainty** is that we don't know whether our sample estimate is *close* or *far* from the unknown population parameter.[†]

However, all is not lost. We can use the errors ($e_i = y_i - \hat{y}_i$) to get a sense of how well our model explains the observed variation in y .

When our model appears to be doing a "nice" job, we might be a little more confident in using it to learn about the relationship between y and x .

Now we just need to formalize what a "nice job" actually means.

[†]: Except when we run the simulation ourselves—which is why we like simulations.

Uncertainty and inference

Learning from our errors

First off, we will estimate the variance of u_i (recall: $\text{Var}(u_i) = \sigma^2$) using our squared errors, *i.e.*,

$$s^2 = \frac{\sum_i e_i^2}{n - k}$$

where k gives the number of slope terms and intercepts that we estimate (*e.g.*, β_0 and β_1 would give $k = 2$).

s^2 is an unbiased estimator of σ^2 .

Uncertainty and inference

Learning from our errors

You then showed that the variance of $\hat{\beta}_1$ (for simple linear regression) is

$$\text{Var}(\hat{\beta}_1) = \frac{s^2}{\sum_i (x_i - \bar{x})^2}$$

which shows that the variance of our slope estimator

1. increases as our disturbances become noisier
2. decreases as the variance of x increases

Uncertainty and inference

Learning from our errors

More common: The **standard error** of $\hat{\beta}_1$

$$\text{SE}(\hat{\beta}_1) = \sqrt{\frac{s^2}{\sum_i (x_i - \bar{x})^2}}$$

Recall: The standard error of an estimator is the standard deviation of the estimator's distribution.

Uncertainty and inference

Learning from our errors

Standard error output is standard in R's `lm`:

```
tidy(lm(y ~ x, pop_df))
```

```
#> # A tibble: 2 x 5
#>   term          estimate std.error statistic  p.value
#>   <chr>          <dbl>    <dbl>    <dbl>    <dbl>
#> 1 (Intercept)    2.53      0.422      6.00 3.38e- 8
#> 2 x              0.567      0.0793      7.15 1.59e-10
```

Uncertainty and inference

Learning from our errors

We use the standard error of $\hat{\beta}_1$, along with $\hat{\beta}_1$ itself, to learn about the parameter β_1 .

After deriving the distribution of $\hat{\beta}_1$,[†] we have two (related) options for formal statistical inference (learning) about our unknown parameter β_1 :

- **Confidence intervals:** Use the estimate and its standard error to create an interval that, when repeated, will generally^{††} contain the true parameter.
- **Hypothesis tests:** Determine whether there is statistically significant evidence to reject a hypothesized value or range of values.

[†]: Hint: it's normal with the mean and variance we've derived/discussed above)

^{††}: E.g., Similarly constructed 95% confidence intervals will contain the true parameter 95% of the time.

Uncertainty and inference

Confidence intervals

We construct $(1 - \alpha)$ -level confidence intervals for β_1

$$\hat{\beta}_1 \pm t_{\alpha/2, \text{df}} \text{SE}(\hat{\beta}_1)$$

$t_{\alpha/2, \text{df}}$ denotes the $\alpha/2$ quantile of a t dist. with $n - k$ degrees of freedom.

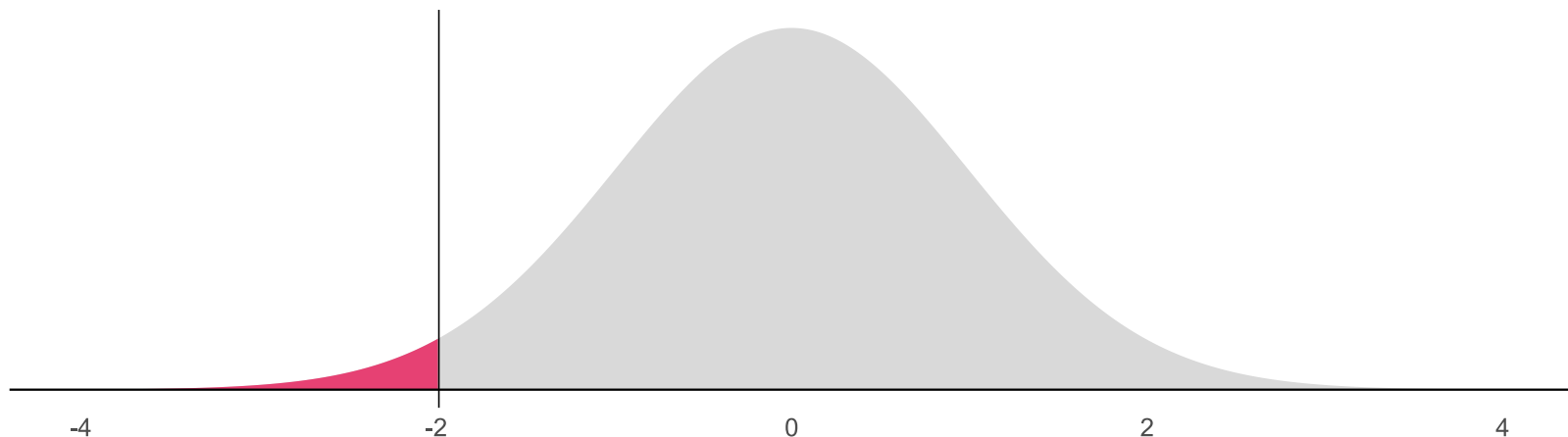
Uncertainty and inference

Confidence intervals

We construct $(1 - \alpha)$ -level confidence intervals for β_1

$$\hat{\beta}_1 \pm t_{\alpha/2, \text{df}} \text{SE}(\hat{\beta}_1)$$

For example, 100 obs., two coefficients (*i.e.*, $\hat{\beta}_0$ and $\hat{\beta}_1 \implies k = 2$), and $\alpha = 0.05$ (for a 95% confidence interval) gives us $t_{0.025, 98} = -1.98$



Uncertainty and inference

Confidence intervals

We construct $(1 - \alpha)$ -level confidence intervals for β_1

$$\hat{\beta}_1 \pm t_{\alpha/2, df} \text{SE}(\hat{\beta}_1)$$

Example:

```
lm(y ~ x, data = pop_df) %>% tidy()
```

```
#> # A tibble: 2 x 5
#>   term          estimate std.error statistic  p.value
#>   <chr>          <dbl>    <dbl>    <dbl>    <dbl>
#> 1 (Intercept)    2.53      0.422      6.00 3.38e- 8
#> 2 x              0.567      0.0793     7.15 1.59e-10
```

Uncertainty and inference

Confidence intervals

We construct $(1 - \alpha)$ -level confidence intervals for β_1

$$\hat{\beta}_1 \pm t_{\alpha/2, df} \text{SE}(\hat{\beta}_1)$$

Example:

```
lm(y ~ x, data = pop_df) %>% tidy()
```

```
#> # A tibble: 2 x 5
#>   term          estimate std.error statistic  p.value
#>   <chr>          <dbl>    <dbl>    <dbl>    <dbl>
#> 1 (Intercept)    2.53      0.422      6.00 3.38e- 8
#> 2 x              0.567      0.0793     7.15 1.59e-10
```

Our 95% confidence interval is thus $0.567 \pm 1.98 \times 0.0793 = [0.410, 0.724]$

Uncertainty and inference

Confidence intervals

So we have a confidence interval for β_1 , *i.e.*, $[0.410, 0.724]$.

What does it mean?

Uncertainty and inference

Confidence intervals

So we have a confidence interval for β_1 , *i.e.*, $[0.410, 0.724]$.

What does it mean?

Informally: The confidence interval gives us a region (interval) in which we can place some trust (confidence) for containing the parameter.

Uncertainty and inference

Confidence intervals

So we have a confidence interval for β_1 , i.e., $[0.410, 0.724]$.

What does it mean?

Informally: The confidence interval gives us a region (interval) in which we can place some trust (confidence) for containing the parameter.

More formally: If repeatedly sample from our population and construct confidence intervals for each of these samples, $(1 - \alpha)$ percent of our intervals (e.g., 95%) will contain the population parameter *somewhere in the interval*.

Uncertainty and inference

Confidence intervals

So we have a confidence interval for β_1 , i.e., $[0.410, 0.724]$.

What does it mean?

Informally: The confidence interval gives us a region (interval) in which we can place some trust (confidence) for containing the parameter.

More formally: If repeatedly sample from our population and construct confidence intervals for each of these samples, $(1 - \alpha)$ percent of our intervals (e.g., 95%) will contain the population parameter *somewhere in the interval*.

Now back to our simulation...

Uncertainty and inference

Confidence intervals

We drew 10,000 samples (each of size $n = 30$) from our population and estimated our regression model for each of these simulations:

$$y_i = \hat{\beta}_0 + \hat{\beta}_1 x_i + e_i$$

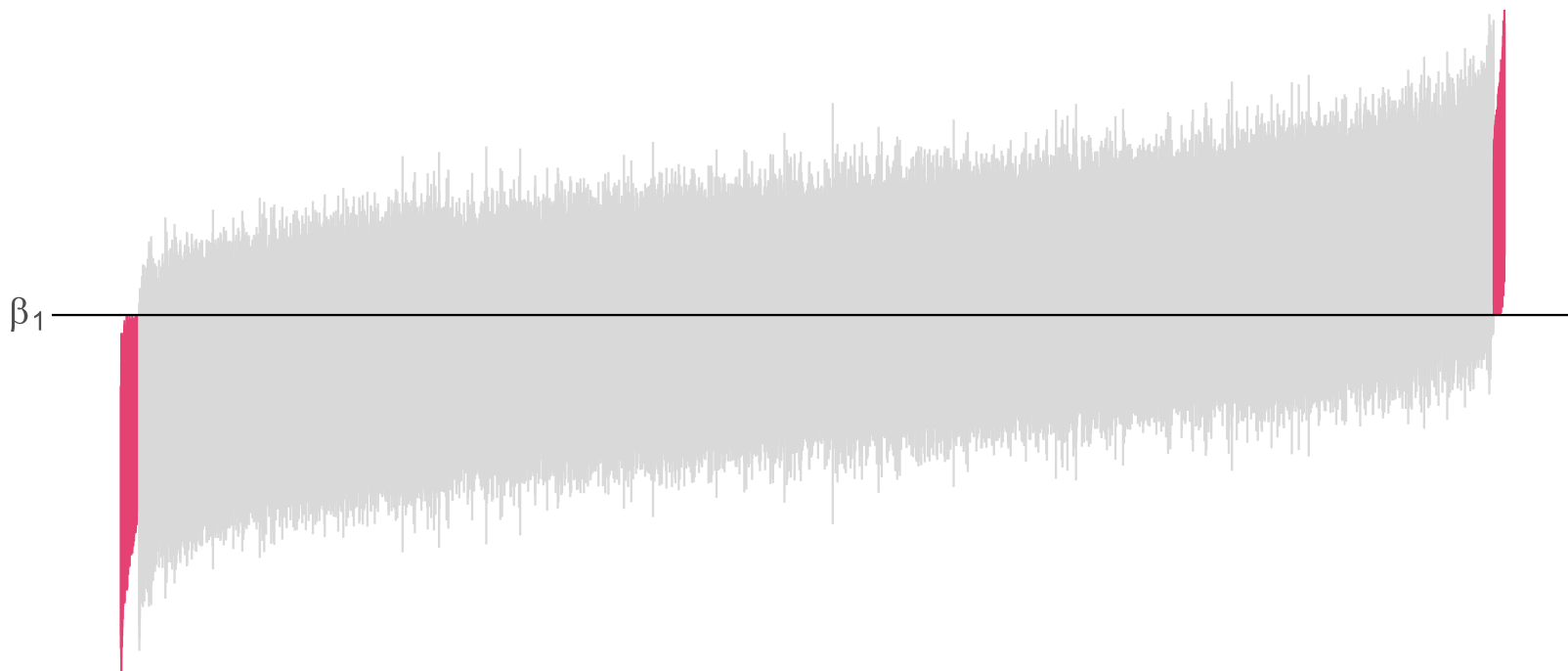
(repeated 10,000 times)

Now, let's estimate 95% confidence intervals for each of these intervals...

Uncertainty and inference

Confidence intervals

From our previous simulation: 97.9% of 95% confidence intervals contain the true parameter value of β_1 .



Uncertainty and inference

Hypothesis testing

In many applications, we want to know more than a point estimate or a range of values. We want to know what our statistical evidence says about existing theories.

We want to test hypotheses posed by officials, politicians, economists, scientists, friends, weird neighbors, *etc.*

Examples

- Does increasing police presence **reduce crime**?
- Does building a giant wall **reduce crime**?
- Does shutting down a government **adversely affect the economy**?
- Does legal cannabis **reduce drunk driving** or **reduce opioid use**?
- Do air quality standards **increase health** and/or **reduce jobs**?

Uncertainty and inference

Hypothesis testing

Hypothesis testing relies upon very similar results and intuition.

While uncertainty certainly exists, we can still build *reliable* statistical tests (rejecting or failing to reject a posited hypothesis).

Uncertainty and inference

Hypothesis testing

Hypothesis testing relies upon very similar results and intuition.

While uncertainty certainly exists, we can still build *reliable* statistical tests (rejecting or failing to reject a posited hypothesis).

OLS t test Our (null) hypothesis states that β_1 equals a value c , *i.e.*,
 $H_o : \beta_1 = c$

From OLS's properties, we can show that the test statistic

$$t_{\text{stat}} = \frac{\hat{\beta}_1 - c}{\text{SE}(\hat{\beta}_1)}$$

follows the t distribution with $n - k$ degrees of freedom.

Uncertainty and inference

Hypothesis testing

For an α -level, **two-sided** test, we reject the null hypothesis (and conclude with the alternative hypothesis) when

$$|t_{\text{stat}}| > |t_{1-\alpha/2, df}|$$

meaning that our **test statistic is more extreme than the critical value**.

Alternatively, we can calculate the **p-value** that accompanies our test statistic, which effectively gives us the probability of seeing our test statistic *or a more extreme test statistic* if the null hypothesis were true.

Very small p-values (generally < 0.05) mean that it would be unlikely to see our results if the null hypothesis were really true—we tend to reject the null for p-values below 0.05.

Uncertainty and inference

Hypothesis testing

R default to testing hypotheses against the value zero.

```
lm(y ~ x, data = pop_df) %>% tidy()
```

```
#> # A tibble: 2 x 5
#>   term          estimate std.error statistic  p.value
#>   <chr>          <dbl>    <dbl>    <dbl>    <dbl>
#> 1 (Intercept)    2.53      0.422      6.00 3.38e- 8
#> 2 x              0.567     0.0793     7.15 1.59e-10
```

Uncertainty and inference

Hypothesis testing

R default to testing hypotheses against the value zero.

```
lm(y ~ x, data = pop_df) %>% tidy()
```

```
#> # A tibble: 2 x 5
#>   term          estimate std.error statistic  p.value
#>   <chr>          <dbl>    <dbl>    <dbl>    <dbl>
#> 1 (Intercept)    2.53      0.422      6.00 3.38e- 8
#> 2 x              0.567     0.0793     7.15 1.59e-10
```

$H_0: \beta_1 = 0$ vs. $H_a: \beta_1 \neq 0$

Uncertainty and inference

Hypothesis testing

R default to testing hypotheses against the value zero.

```
lm(y ~ x, data = pop_df) %>% tidy()
```

```
#> # A tibble: 2 x 5
#>   term          estimate std.error statistic  p.value
#>   <chr>          <dbl>    <dbl>    <dbl>    <dbl>
#> 1 (Intercept)    2.53      0.422      6.00 3.38e- 8
#> 2 x              0.567     0.0793     7.15 1.59e-10
```

$H_0: \beta_1 = 0$ vs. $H_a: \beta_1 \neq 0$

$t_{\text{stat}} = 7.15$ and $t_{0.975, 28} = 2.05$

Uncertainty and inference

Hypothesis testing

R default to testing hypotheses against the value zero.

```
lm(y ~ x, data = pop_df) %>% tidy()
```

```
#> # A tibble: 2 x 5
#>   term          estimate std.error statistic  p.value
#>   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
#> 1 (Intercept)    2.53      0.422      6.00 3.38e- 8
#> 2 x              0.567     0.0793     7.15 1.59e-10
```

$H_0: \beta_1 = 0$ vs. $H_a: \beta_1 \neq 0$

$t_{\text{stat}} = 7.15$ and $t_{0.975, 28} = 2.05$ which implies $p\text{-value} < 0.05$

Uncertainty and inference

Hypothesis testing

R default to testing hypotheses against the value zero.

```
lm(y ~ x, data = pop_df) %>% tidy()
```

```
#> # A tibble: 2 x 5  
#>   term          estimate std.error statistic  p.value  
#>   <chr>          <dbl>    <dbl>    <dbl>    <dbl>  
#> 1 (Intercept)    2.53      0.422      6.00 3.38e- 8  
#> 2 x              0.567     0.0793     7.15 1.59e-10
```

$H_0: \beta_1 = 0$ vs. $H_a: \beta_1 \neq 0$

$t_{\text{stat}} = 7.15$ and $t_{0.975, 28} = 2.05$ which implies $p\text{-value} < 0.05$

Therefore, we **reject H_0** .

Uncertainty and inference

Hypothesis testing

Back to our simulation! Let's see what our t statistic is actually doing.

In this situation, we can actually know (and enforce) the null hypothesis, since we generated the data.

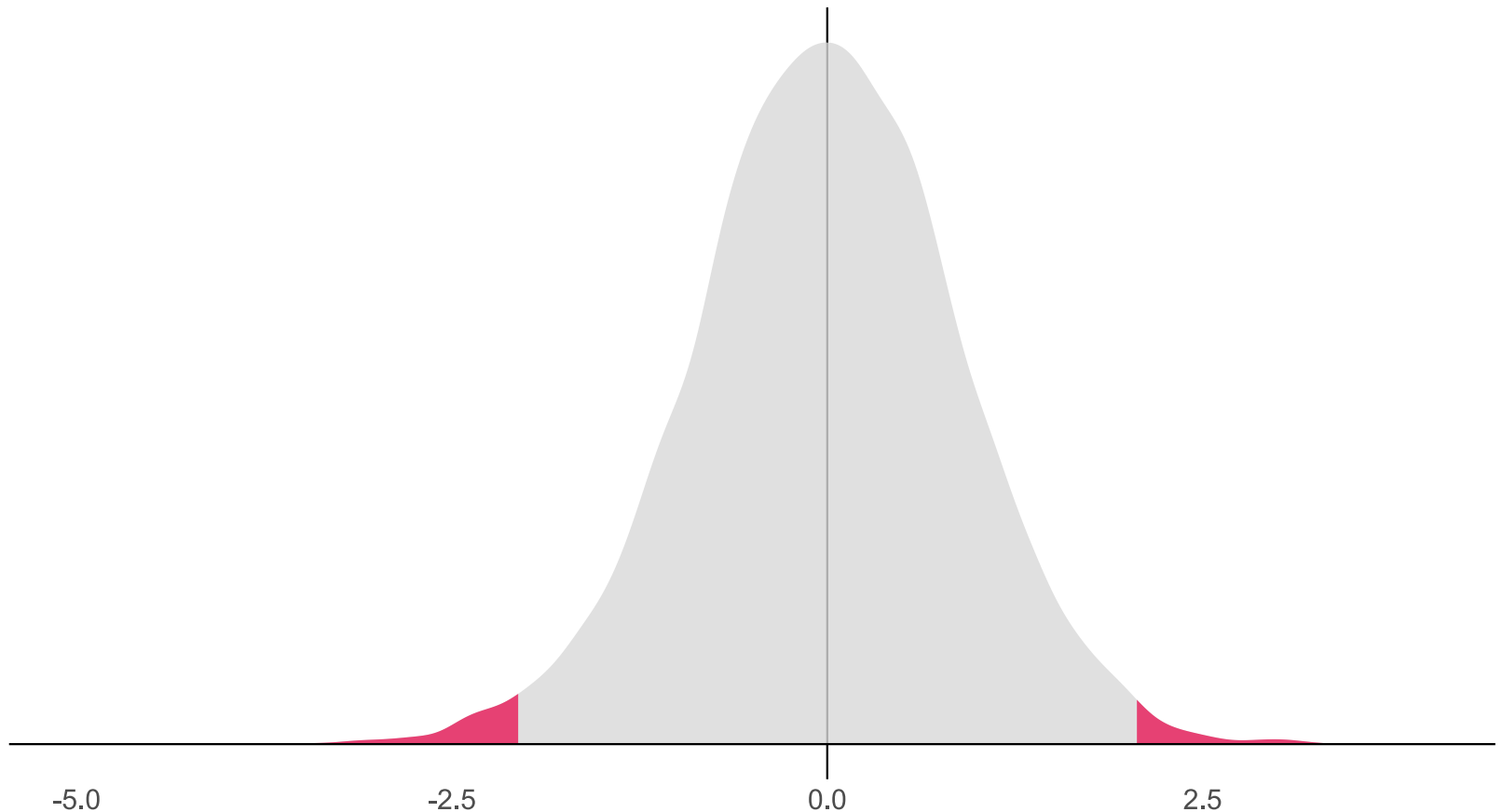
For each of the 10,000 samples, we will calculate the t statistic, and then we can see how many t statistics exceed our critical value (2.05, as above).

The answer should be approximately 5 percent—our α level.

Uncertainty and inference

In our simulation, 2.1 percent of our t statistics reject the null hypothesis.

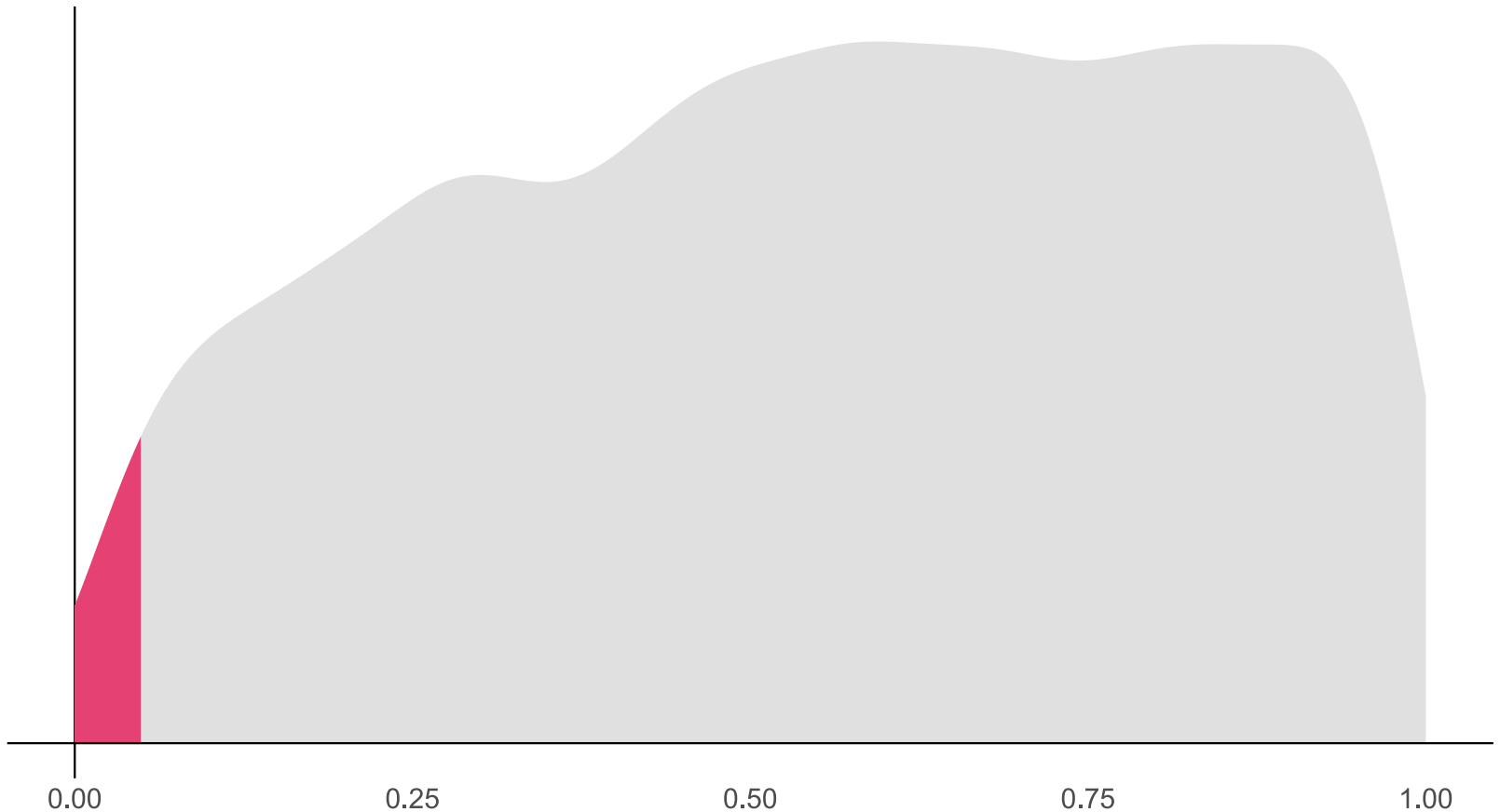
The distribution of our t statistics (shading the rejection regions).



Uncertainty and inference

Correspondingly, 2.1 percent of our p-values reject the null hypothesis.

The distribution of our p-values (shading the p-values below 0.05).



Uncertainty and inference

F tests (will leave you to check this section)

You will sometimes see F tests in econometrics.

We use F tests to test hypotheses that involve multiple parameters
(*e.g.*, $\beta_1 = \beta_2$ or $\beta_3 + \beta_4 = 1$),

rather than a single simple hypothesis

(*e.g.*, $\beta_1 = 0$, for which we would just use a t test).

Uncertainty and inference

F tests (will leave you to check this section)

Example

Economists love to say "Money is fungible."

Imagine that we might want to test whether money received as income actually has the same effect on consumption as money received from tax rebates/returns.

$$\text{Consumption}_i = \beta_0 + \beta_1 \text{Income}_i + \beta_2 \text{Rebate}_i + u_i$$

Uncertainty and inference

F tests (will leave you to check this section)

Example, continued

We can write our null hypothesis as

$$H_o : \beta_1 = \beta_2 \iff H_o : \beta_1 - \beta_2 = 0$$

Imposing this null hypothesis gives us the **restricted model**

$$\text{Consumption}_i = \beta_0 + \beta_1 \text{Income}_i + \beta_1 \text{Rebate}_i + u_i$$

$$\text{Consumption}_i = \beta_0 + \beta_1 (\text{Income}_i + \text{Rebate}_i) + u_i$$

Uncertainty and inference

F tests (will leave you to check this section)

Example, continued

To test the null hypothesis $H_o : \beta_1 = \beta_2$ against $H_a : \beta_1 \neq \beta_2$, we use the F statistic

$$F_{q, n-k-1} = \frac{(\text{SSE}_r - \text{SSE}_u) / q}{\text{SSE}_u / (n - k - 1)}$$

which (as its name suggests) follows the F distribution with q numerator degrees of freedom and $n - k - 1$ denominator degrees of freedom.

Here, q is the number of restrictions we impose via H_o .

Uncertainty and inference

F tests (will leave you to check this section)

Example, continued

SSE_r is the sum of squared errors (SSE) from our **restricted model**

$$\text{Consumption}_i = \beta_0 + \beta_1 (\text{Income}_i + \text{Rebate}_i) + u_i$$

and SSE_u is the sum of squared errors (SSE) from our **unrestricted model**

$$\text{Consumption}_i = \beta_0 + \beta_1 \text{Income}_i + \beta_2 \text{Rebate}_i + u_i$$

Uncertainty and inference

F tests (will leave you to check this section)

Example, continued

SSE_r is the sum of squared errors (SSE) from our **restricted model**

$$\text{Consumption}_i = \beta_0 + \beta_1 (\text{Income}_i + \text{Rebate}_i) + u_i$$

and SSE_u is the sum of squared errors (SSE) from our **unrestricted model**

$$\text{Consumption}_i = \beta_0 + \beta_1 \text{Income}_i + \beta_2 \text{Rebate}_i + u_i$$

An *F* test compares the unrestricted model's performance to the restricted model's performance using their SSEs.

Uncertainty and inference

F tests (will leave you to check this section)

Example, continued

SSE_r is the sum of squared errors (SSE) from our **restricted model**

$$\text{Consumption}_i = \beta_0 + \beta_1 (\text{Income}_i + \text{Rebate}_i) + u_i$$

and SSE_u is the sum of squared errors (SSE) from our **unrestricted model**

$$\text{Consumption}_i = \beta_0 + \beta_1 \text{Income}_i + \beta_2 \text{Rebate}_i + u_i$$

An F test compares the unrestricted model's performance to the restricted model's performance using their SSEs.

$$F_{\text{stat}} = \frac{(SSE_r - SSE_u) / q}{SSE_u / (n - k - 1)} \sim F_{q, n-k-1}$$

Uncertainty and inference

F tests (will leave you to check this section)

Model fit

Measures of *goodness of fit* try to analyze how well our model describes (*fits*) the data.

Common measure: R^2 [R-squared] (*a.k.a.* coefficient of determination)

$$R^2 = \frac{\sum_i (\hat{y}_i - \bar{y})^2}{\sum_i (y_i - \bar{y})^2} = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}$$

Notice our old friend SSE: $\sum_i (y_i - \hat{y}_i)^2 = \sum_i e_i^2$.

Uncertainty and inference

F tests (will leave you to check this section)

Model fit

Measures of *goodness of fit* try to analyze how well our model describes (*fits*) the data.

Common measure: R^2 [R-squared] (*a.k.a.* coefficient of determination)

$$R^2 = \frac{\sum_i (\hat{y}_i - \bar{y})^2}{\sum_i (y_i - \bar{y})^2} = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}$$

Notice our old friend SSE: $\sum_i (y_i - \hat{y}_i)^2 = \sum_i e_i^2$.

R^2 literally tells us the share of the variance in y our current models accounts for. Thus $0 \leq R^2 \leq 1$.

Uncertainty and inference

F tests (will leave you to check this section)

The problem: As we add variables to our model, R^2 *mechanically* increases.

Uncertainty and inference

F tests (will leave you to check this section)

The problem: As we add variables to our model, R^2 *mechanically* increases.

To see this problem, we can simulate a dataset of 10,000 observations on y and 1,000 random x_k variables. **No relations between y and the x_k !**

Pseudo-code outline of the simulation:

Uncertainty and inference

F tests (will leave you to check this section)

The problem: As we add variables to our model, R^2 *mechanically* increases.

To see this problem, we can simulate a dataset of 10,000 observations on y and 1,000 random x_k variables. **No relations between y and the x_k !**

Pseudo-code outline of the simulation:

- Generate 10,000 observations on y
- Generate 10,000 observations on variables x_1 through x_{1000}
- Regressions
 - LM₁: Regress y on x_1 ; record R^2
 - LM₂: Regress y on x_1 and x_2 ; record R^2
 - LM₃: Regress y on x_1 , x_2 , and x_3 ; record R^2
 - ...

Uncertainty and inference

F tests (will leave you to check this section)

The problem: As we add variables to our model, R^2 *mechanically* increases.

R code for the simulation:

```
set.seed(1234)
y <- rnorm(1e4)
x <- matrix(data = rnorm(1e7), nrow = 1e4)
x %<>% cbind(matrix(data = 1, nrow = 1e4, ncol = 1), x)
r_df <- lapply(X = 1:(1e3-1), FUN = function(i) {
  tmp_reg <- lm(y ~ x[,1:(i+1)]) %>% summary()
  data.frame(
    k = i + 1,
    r2 = tmp_reg %$% r.squared,
    r2_adj = tmp_reg %$% adj.r.squared
  )
}) %>% bind_rows()
```

Uncertainty and inference

F tests (will leave you to check this section)

The problem: As we add variables to our model, R^2 *mechanically* increases.

Uncertainty and inference

F tests (will leave you to check this section)

One solution: Adjusted R^2

Uncertainty and inference

F tests (will leave you to check this section)

The problem: As we add variables to our model, R^2 *mechanically* increases.

One solution: Penalize for the number of variables, e.g., adjusted R^2 :

$$\bar{R}^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2 / (n - k - 1)}{\sum_i (y_i - \bar{y})^2 / (n - 1)}$$

Note: Adjusted R^2 need not be between 0 and 1.

Uncertainty and inference

F tests (will leave you to check this section)

Tradeoffs

There are tradeoffs to remember as we add/remove variables:

Fewer variables

- Generally explain less variation in y
- Provide simple interpretations and visualizations (*parsimonious*)
- May need to worry about omitted-variable bias

More variables

- More likely to find *spurious* relationships (statistically significant due to chance—does not reflect a true, population-level relationship)
- More difficult to interpret the model

Next: Potential outcomes + ML/CI distinction

Mullainathan and Spiess (2017)