To the lab for testing

HYPOTHESIS TESTING IN R

Richie Cotton
Data Evangelist at DataCamp

A/B testing

- Electronic Arts (EA) is a video game company.
- In 2013, they released SimCity 5.
- Their goal was to increase pre-orders of the game.
- They used A/B testing to test different advertising scenarios.
- This involves splitting users into *control* and *treatment* groups.

[1] Image credit: "Electronic Arts" by majaX1 CC BY-NC-SA 2.0

Retail webpage A/B test

Control

Treatment

A/B test results

- The treatment group (no ad) got 43.4% more purchases than the control group (with ad).
- The intuition that "showing an ad would increase sales" was completely wrong.
- Was this result *statistically significant* or just by chance?
- You need EA's data to determine this.
- You'd use techniques from Sampling in R + this course to do so.

Stack Overflow Developer Survey 2020

```r
library(dplyr)
glimpse(stack_overflow)
```

```
Rows: 2,261
Columns: 8
$ respondent          <dbl> 36, 47, 69, 125, 147, 152, 166, 170, 187, 196, 221,…
$ age_first_code_cut <chr> "adult", "child", "child", "adult", "adult", "adult…
$ converted_comp      <dbl> 77556, 74970, 594539, 2000000, 37816, 121980, 48644…
$ job_sat             <fct> Slightly satisfied, Very satisfied, Very satisfied,…
$ purple_link         <chr> "Hello, old friend", "Hello, old friend", "Hello, o…
$ age_cat             <chr> "At least 30", "At least 30", "Under 30", "At least…
$ age                 <dbl> 34, 53, 25, 41, 28, 30, 28, 26, 43, 23, 24, 35, 37,…
$ hobbyist            <chr> "Yes", "Yes", "Yes", "Yes", "No", "Yes", "Yes", "Ye…
```

Hypothesizing about the mean

A hypothesis:

> The mean annual compensation of the population of data scientists is $110,000.

The point estimate (sample statistic):

```
mean_comp_samp <- mean(stack_overflow$converted_comp)
```

```
mean_comp_samp <- stack_overflow %>%
  summarize(mean_compensation = mean(converted_comp)) %>%
  pull(mean_compensation)
```

```
121915.4
```

Generating a bootstrap distribution

```
# Step 3. Repeat steps 1 & 2 many times
so_boot_distn <- replicate(
  n = 5000,
  expr = {
```

```
    # Step 1. Resample
    stack_overflow %>%
      slice_sample(prop = 1, replace = TRUE) %>%
```

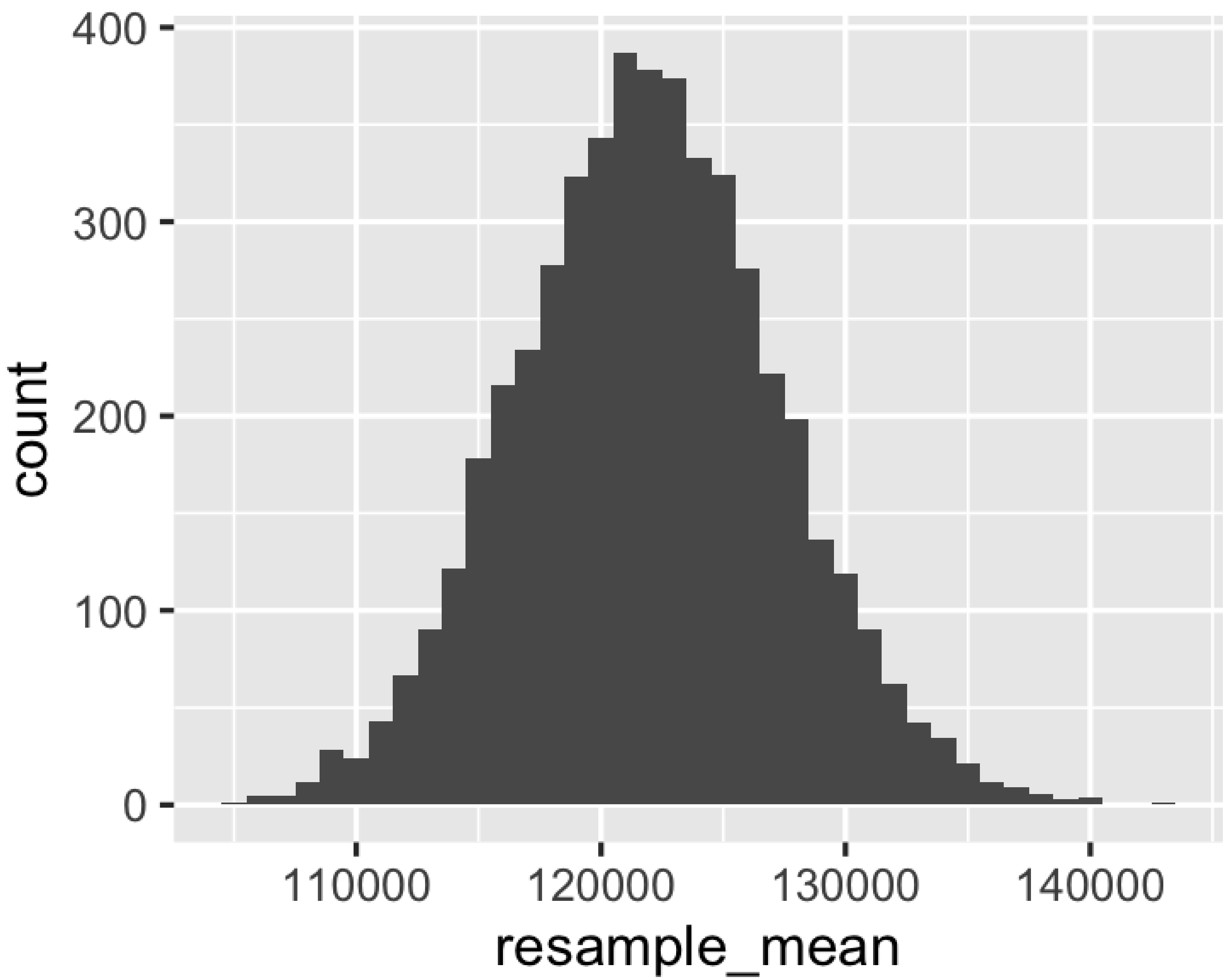
```
      # Step 2. Calculate point estimate
      summarize(mean_compensation = mean(converted_comp)) %>%
      pull(mean_compensation)
```

```
  }
)
```

[1] Bootstrap distributions are taught in Chapter 4 of Sampling in R

Visualizing the bootstrap distribution

```
tibble(resample_mean = so_boot_distn) %>%
  ggplot(aes(resample_mean)) +
  geom_histogram(binwidth = 1000)
```



Standard error

```
std_error <- sd(so_boot_distn)
```

```
5344.653
```

z-scores

$$\text{standardized value} = \frac{\text{value} - \text{mean}}{\text{standard deviation}}$$

$$z = \frac{\text{sample stat} - \text{hypoth. param. value}}{\text{standard error}}$$

$$z = \frac{\$121,915.4 - \$110,000}{\$5344.65} = 2.233$$

```
mean_comp_samp
```

```
121915.4
```

```
mean_comp_hyp <- 110000
```

```
std_error
```

```
5344.653
```

```
z_score <- (mean_comp_samp - mean_comp_hyp) / std_error
```

```
2.233
```

Testing the hypothesis

- Is 2.233 a high or low number?
- This is the goal of the course!

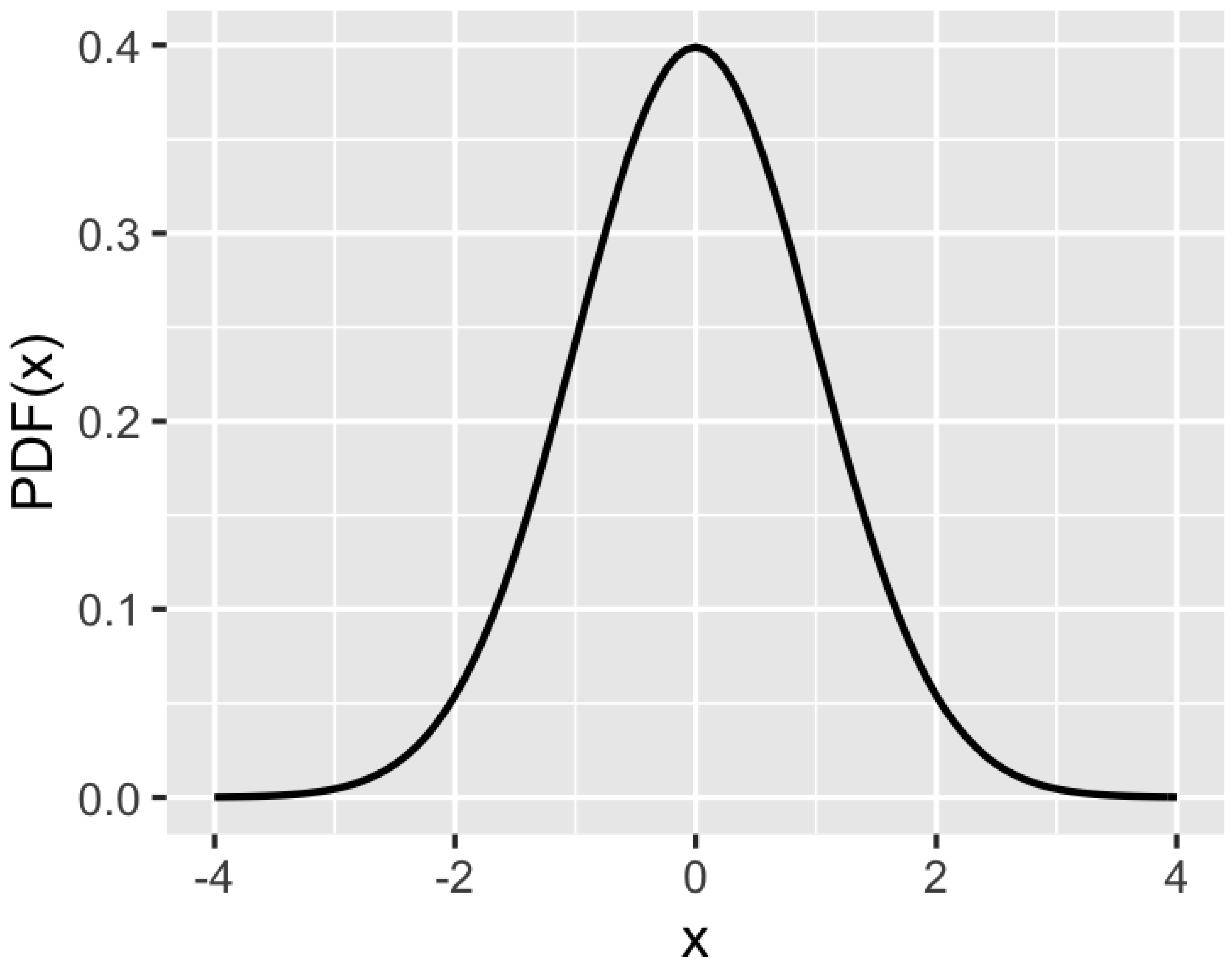
Hypothesis testing use case:

> Determine whether sample statistics are close to or far away from expected (or "hypothesized" values).

Standard normal (z) distribution

Standard normal distribution: the normal distribution with mean zero, standard deviation 1.

```
tibble(x = seq(-4, 4, 0.01)) %>%
  ggplot(aes(x)) +
  stat_function(fun = dnorm) +
  ylab("PDF(x)")
```



Let's practice!

HYPOTHESIS TESTING IN R

A tail of two z's

HYPOTHESIS TESTING IN R

Richie Cotton
Data Evangelist at DataCamp

Criminal trials

- Two possible true states.
  1. Defendant committed the crime.
  2. Defendant did not commit the crime.
- Two possible verdicts.
  1. Guilty.
  2. Not guilty.
- Initially the defendant is assumed to be not guilty.
- If the evidence is "beyond a reasonable doubt" that the defendant committed the crime, then a "guilty" verdict is given, else a "not guilty" verdict is given.

Age of first programming experience

- `age_first_code_cut` classifies when Stack Overflow user first started programming
  1. `"adult"` means they started at 14 or older
  2. `"child"` means they started before 14
- Previous research suggests that 35% of software developers started programming as children
- Does our sample provide evidence that data scientists have a greater proportion starting programming as a child?

Definitions

A *hypothesis* is a statement about an unknown population parameter.

A *hypothesis test* is a test of two competing hypotheses.

- The *null hypothesis* (H_0) is the existing "champion" idea.
- The *alternative hypothesis* (H_A) is the new "challenger" idea of the researcher.

For our problem

- H_0: The proportion of data scientists starting programming as children is the same as that of software developers (35%).
- H_A: The proportion of data scientists starting programming as children is greater than 35%.

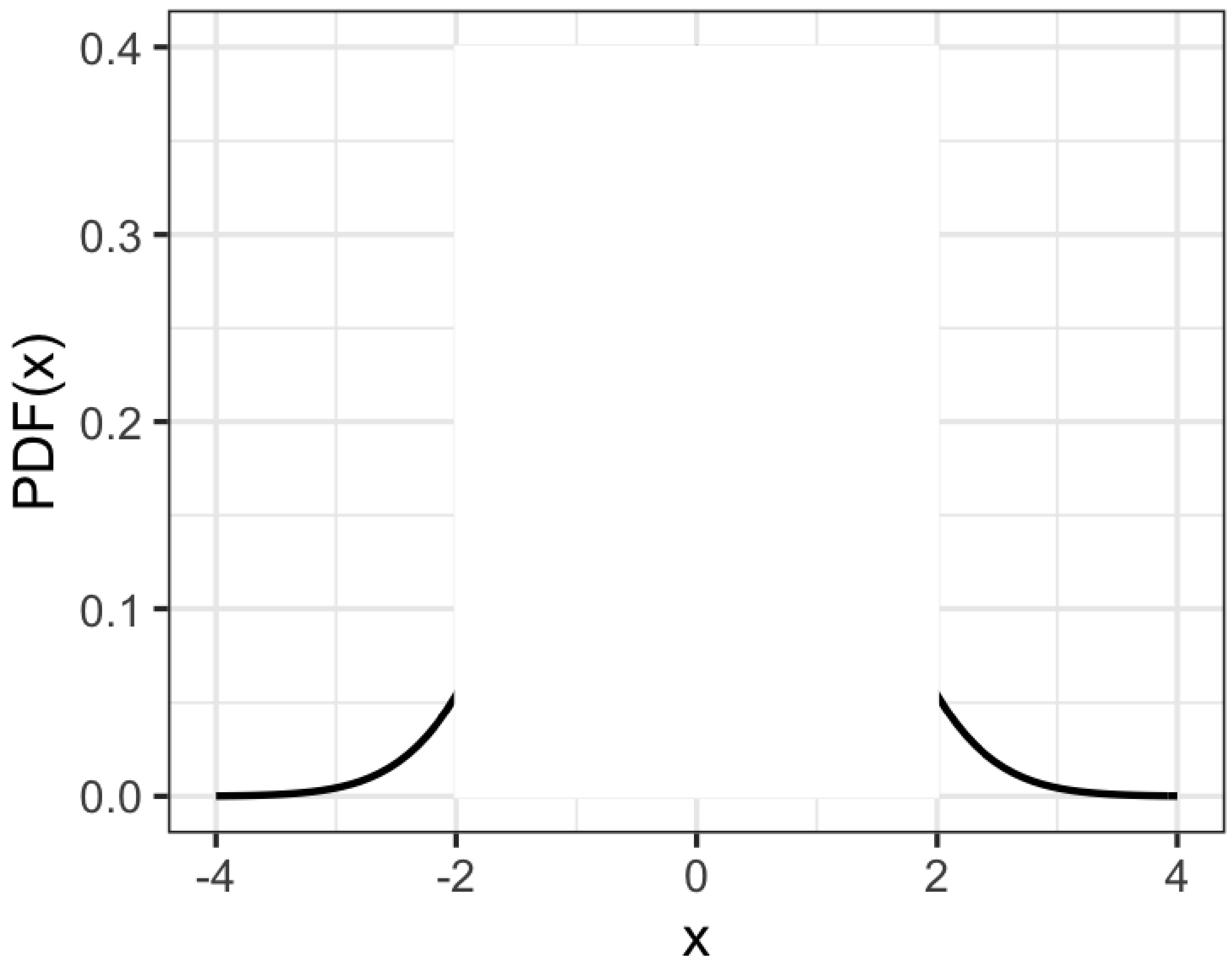
[1] "Naught" is British English for "zero". For historical reasons, "H-naught" is the international convention for pronouncing the null hypothesis.

- Two possible true states.
  1. Defendant committed the crime.
  2. Defendant did not commit the crime.
- Two possible verdicts.
  1. Guilty.
  2. Not guilty.
- Initially the defendant is assumed to be not guilty.
- If the evidence is "beyond a reasonable doubt" that the defendant committed the crime, then a "guilty" verdict is given, else a "not guilty" verdict is given.

- In reality, either H_A or H_0 is true (but not both).
- The test ends in either "reject H_0" verdict or "fail to reject H_0".
- Initially the null hypothesis, H_0, is assumed to be true.
- If the evidence from the sample is "significant" that H_A is true, choose that hypothesis, else choose H_0 .

Significance level is "beyond a reasonable doubt" for hypothesis testing.

One-tailed and two-tailed tests



Hypothesis tests determine whether the sample statistics lie in the tails of the **null** distribution.

Test	***Tails***
alternative *different from* null	two-tailed
alternative *greater than* null	right-tailed
alternative *less than* null	left-tailed

H_A: The proportion of data scientists starting programming as children is **greater than** 35%.

Our alternative hypothesis uses "greater than," so we need a **right-tailed** test.

p-values

- The larger the p-value, the stronger the support for H_0.
- The smaller the p-value, the stronger the evidence against H_0.
- Small p-values mean the statistic is in the tail of the *null distribution* (the distribution of the statistic if the null hypothesis was true).
  - The "p" in *p-value* stands for probability.
  - For p-values, "small" means "close to zero".

Defining p-values

A *p-value* is

the probability of observing a test statistic

as extreme or more extreme

than what was observed in our original sample,

assuming the null hypothesis is true.

Calculating the z-score

```
prop_child_samp <- stack_overflow %>%
  summarize(point_estimate = mean(age_first_code_cut == "child")) %>%
  pull(point_estimate)
```

```
0.388
```

```
prop_child_hyp <- 0.35
```

```
std_error <- 0.0096028
```

```
z_score <- (prop_child_samp - prop_child_hyp) / std_error
```

```
3.956
```

Calculating the p-value

- `pnorm()` is normal CDF.
- Left-tailed test → use default `lower.tail = TRUE` .
- Right-tailed test → set `lower.tail = FALSE` .

```
p_value <- pnorm(z_score, lower.tail = FALSE)
```

```
3.818e-05
```

Let's practice!

HYPOTHESIS TESTING IN R

Statistically significant other

HYPOTHESIS TESTING IN R

Richie Cotton
Data Evangelist at DataCamp

p-value recap

- p-values quantify evidence for the null hypothesis.
- Large p-value → fail to reject null hypothesis.
- Small p-value → reject null hypothesis.
- Where is the cutoff point?

Significance level

The *significance level* of a hypothesis test (α) is the threshold point for "beyond a reasonable doubt".

- Common values of α are `0.1`, `0.05`, and `0.01`.
- If $p \leq \alpha$, reject H_0, else fail to reject H_0.
- α should be set **prior** to conducting the hypothesis test.

Calculating the p-value

```
alpha <- 0.05
```

```
prop_child_samp <- stack_overflow %>%
  summarize(
    point_estimate = mean(age_first_code_cut == "child")
  ) %>%
  pull(point_estimate)
prop_child_hyp <- 0.35
std_error <- 0.0096028
z_score <- (prop_child_samp - prop_child_hyp) / std_error
```

```
p_value <- pnorm(z_score, lower.tail = FALSE)
```

```
3.818e-05
```

```
p_value <= alpha
```

```
TRUE
```

`p_value` is less than or equal to `alpha`, so reject H_0 and accept H_A.

The proportion of data scientists starting programming as children is greater than 35%.

Confidence intervals

For a significance level of 0.05, it's common to choose a confidence interval of `1 - 0.05 = 0.95`.

```
conf_int <- first_code_boot_distn %>%
  summarize(
    lower = quantile(first_code_child_rate, 0.025),
    upper = quantile(first_code_child_rate, 0.975)
  )
```

```
# A tibble: 1 x 2
  lower upper
  <dbl> <dbl>
1 0.369 0.407
```

Types of errors

	Truly didn't commit crime	Truly committed crime
Verdict not guilty	correct	they got away with it
Verdict guilty	wrongful conviction	correct

	actual H_0	**actual** H_A
chosen H_0	correct	false negative
chosen H_A	false positive	correct

False positives are *Type I errors*; false negatives are *Type II errors*.

Possible errors in our example

If $p \leq \alpha$, we reject H_0:

- A false positive (Type I) error could have occurred: we thought that data scientists started coding as children at a higher rate when in reality they did not.

If $p > \alpha$, we fail to reject H_0:

- A false negative (Type II) error could have occurred: we thought that data scientists coded as children at the same rate as software engineers when in reality they coded as children at a higher rate.

Let's practice!

HYPOTHESIS TESTING IN R