

Hypothesis tests and z-scores

HYPOTHESIS TESTING IN PYTHON



James Chapman
Curriculum Manager, DataCamp

A/B testing

- In 2013, Electronic Arts (EA) released SimCity 5
- They wanted 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)
- Intuition that "showing an ad would increase sales" was false
- Was this result *statistically significant* or just chance?
- Need EA's data to determine this
- Techniques from Sampling in Python + this course to do so

Stack Overflow Developer Survey 2020

```
import pandas as pd
print(stack_overflow)
```

```
   respondent  age_1st_code  ...   age hobbyist
0          36.0         30.0  ...  34.0      Yes
1          47.0         10.0  ...  53.0      Yes
2          69.0         12.0  ...  25.0      Yes
3         125.0         30.0  ...  41.0      Yes
4         147.0         15.0  ...  28.0       No
...         ...         ...  ...  ...      ...
2259      62867.0         13.0  ...  33.0      Yes
2260      62882.0         13.0  ...  28.0      Yes
```

```
[2261 rows x 8 columns]
```

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 = stack_overflow['converted_comp'].mean()
```

```
119574.71738168952
```

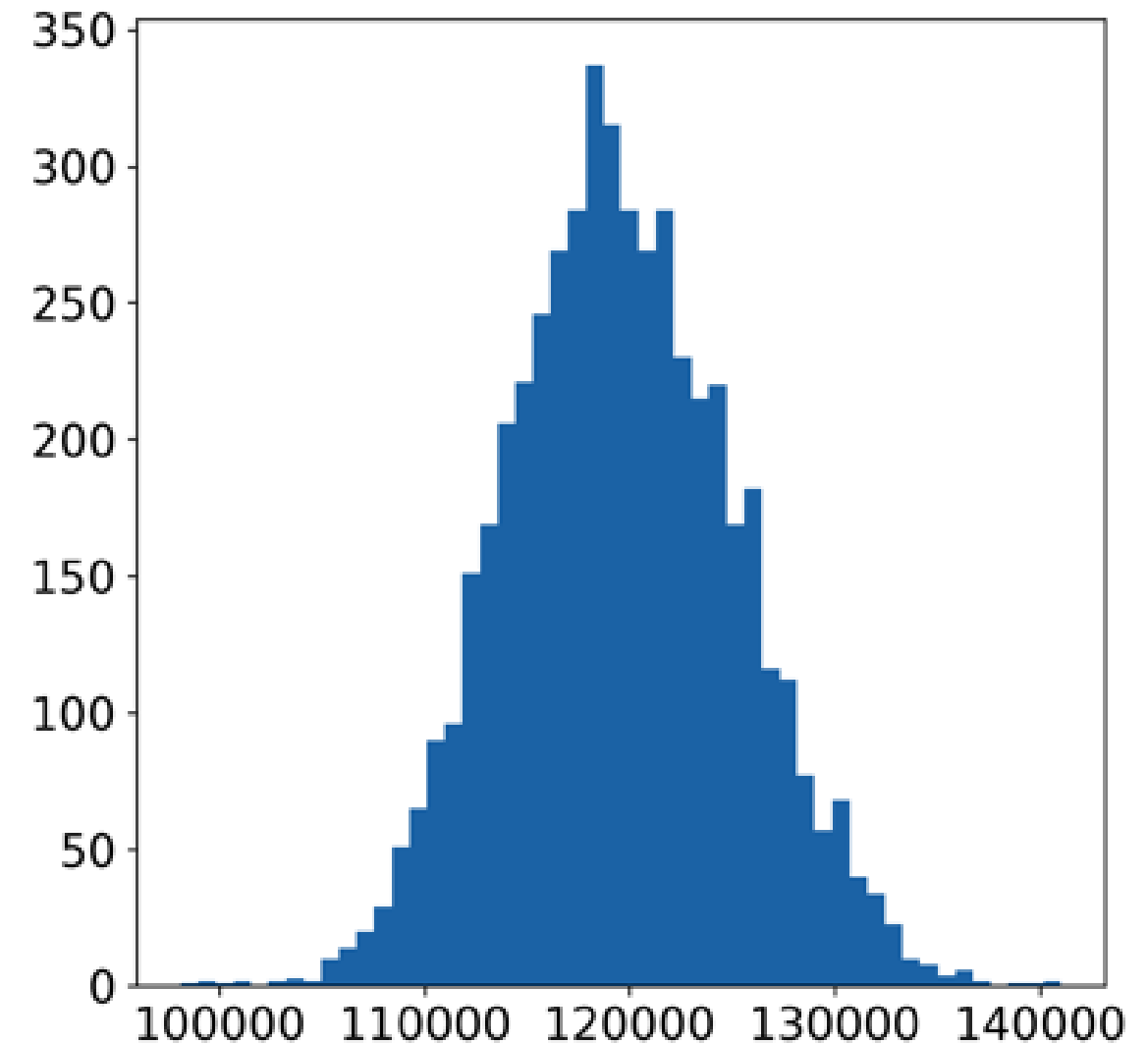
Generating a bootstrap distribution

```
import numpy as np
# Step 3. Repeat steps 1 & 2 many times, appending to a list
so_boot_distn = []
for i in range(5000):
    so_boot_distn.append(
        # Step 2. Calculate point estimate
        np.mean(
            # Step 1. Resample
            stack_overflow.sample(frac=1, replace=True)['converted_comp']
        )
    )
```

¹ Bootstrap distributions are taught in Chapter 4 of Sampling in Python

Visualizing the bootstrap distribution

```
import matplotlib.pyplot as plt
plt.hist(so_boot_distn, bins=50)
plt.show()
```



Standard error

```
std_error = np.std(so_boot_distn, ddof=1)
```

```
5607.997577378606
```

Z-scores

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

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

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

```
stack_overflow['converted_comp'].mean()
```

```
119574.71738168952
```

```
mean_comp_hyp = 110000
```

```
std_error
```

```
5607.997577378606
```

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

```
1.7073326529796957
```

Testing the hypothesis

- Is 1.707 a high or low number?
- This is the goal of the course!

Testing the hypothesis

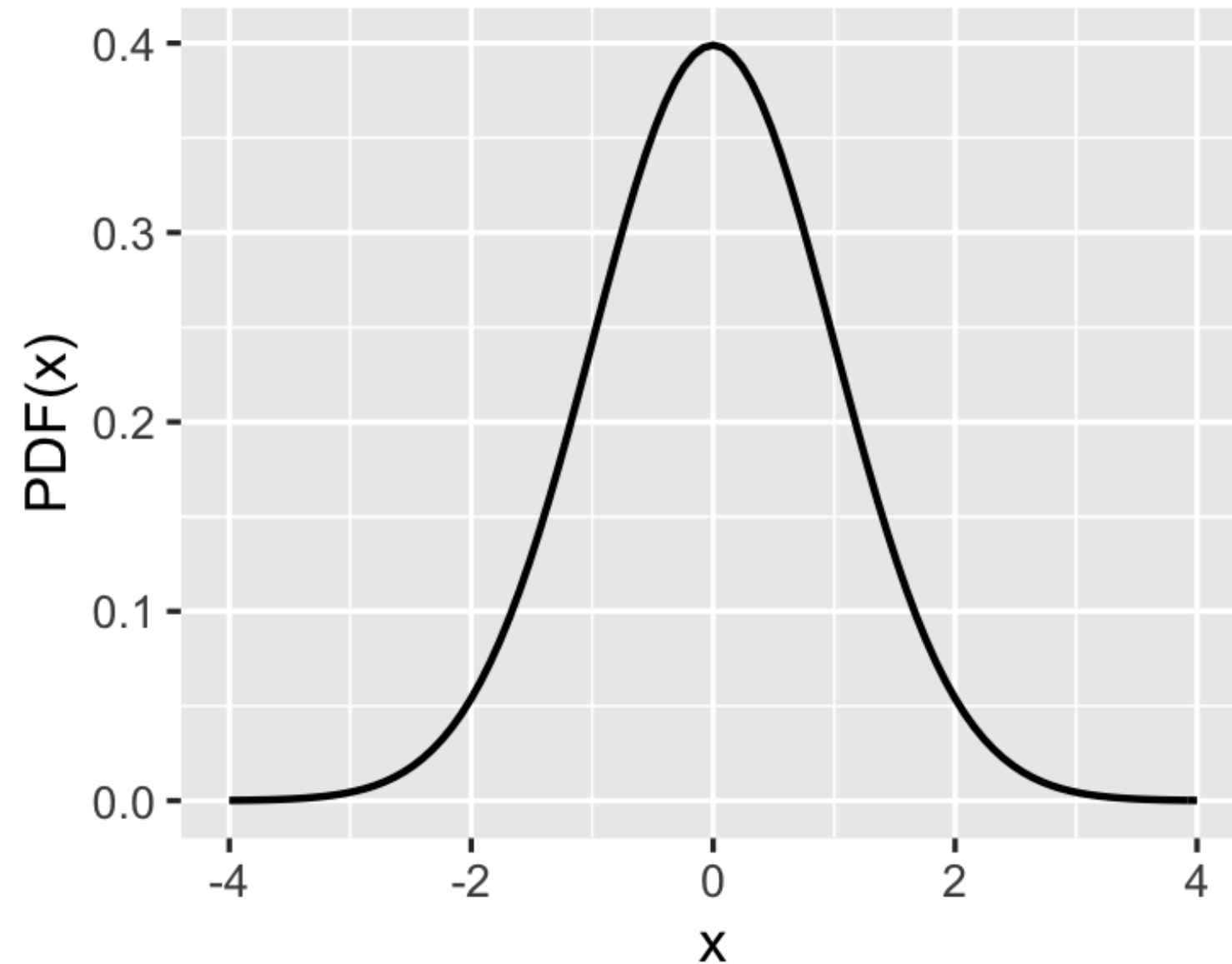
- Is 1.707 a high or low number?
- This is the goal of the course!

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: normal distribution with **mean = 0** + **standard deviation = 1**



Let's practice!
HYPOTHESIS TESTING IN PYTHON

p-values

HYPOTHESIS TESTING IN PYTHON



James Chapman

Curriculum Manager, 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
- Prosecution must present evidence "beyond reasonable doubt" for a guilty verdict

Age of first programming experience

- `age_first_code_cut` classifies when Stack Overflow user first started programming
 - `"adult"` means they started at 14 or older
 - `"child"` means they started before 14
- Previous research: 35% of software developers started programming as children
- Evidence that a greater proportion of data scientists starting programming as children?

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

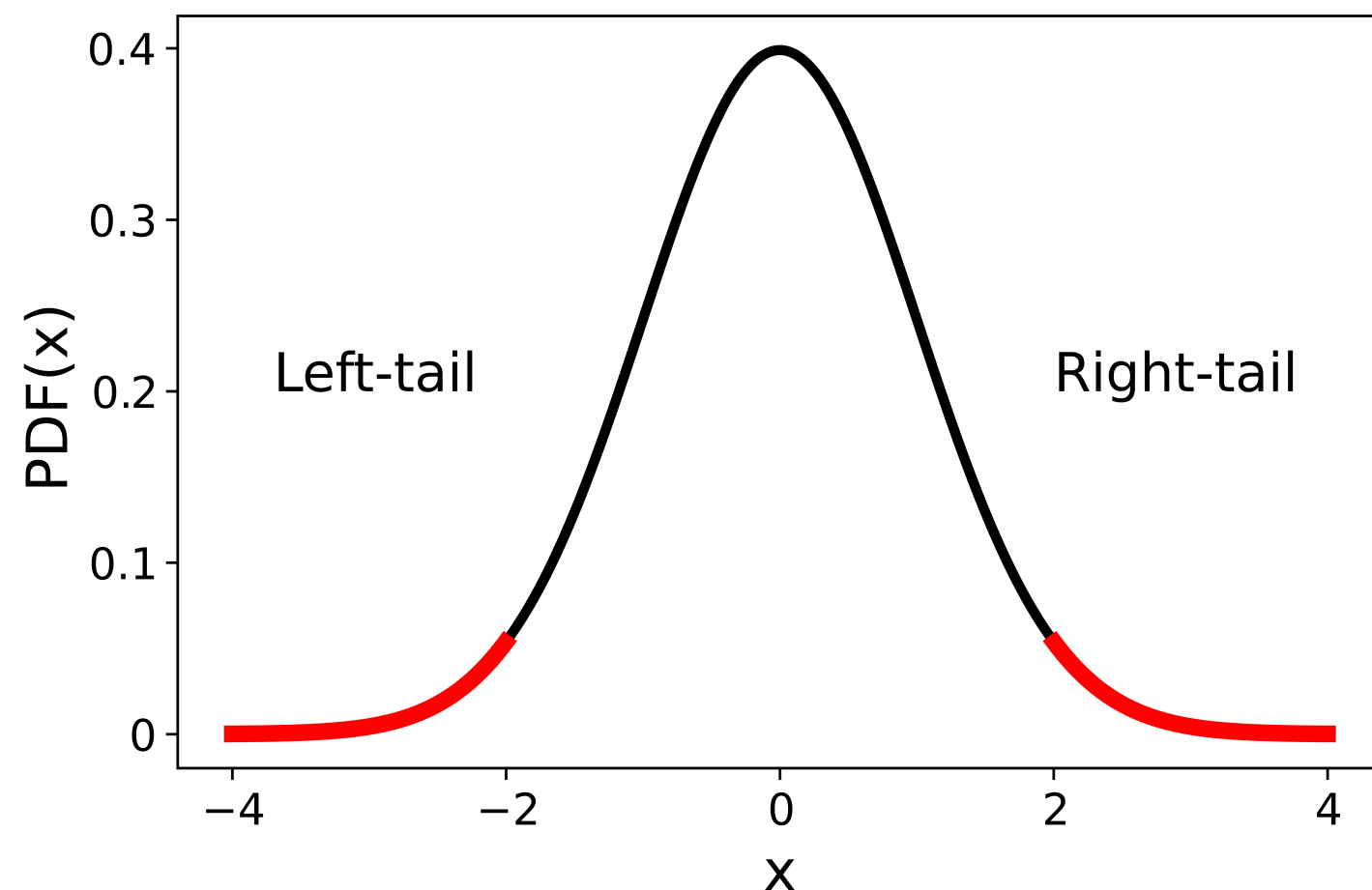
¹ "Naught" is British English for "zero". For historical reasons, "H-naught" is the international convention for pronouncing the null hypothesis.

Criminal trials vs. hypothesis testing

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

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

One-tailed and two-tailed tests



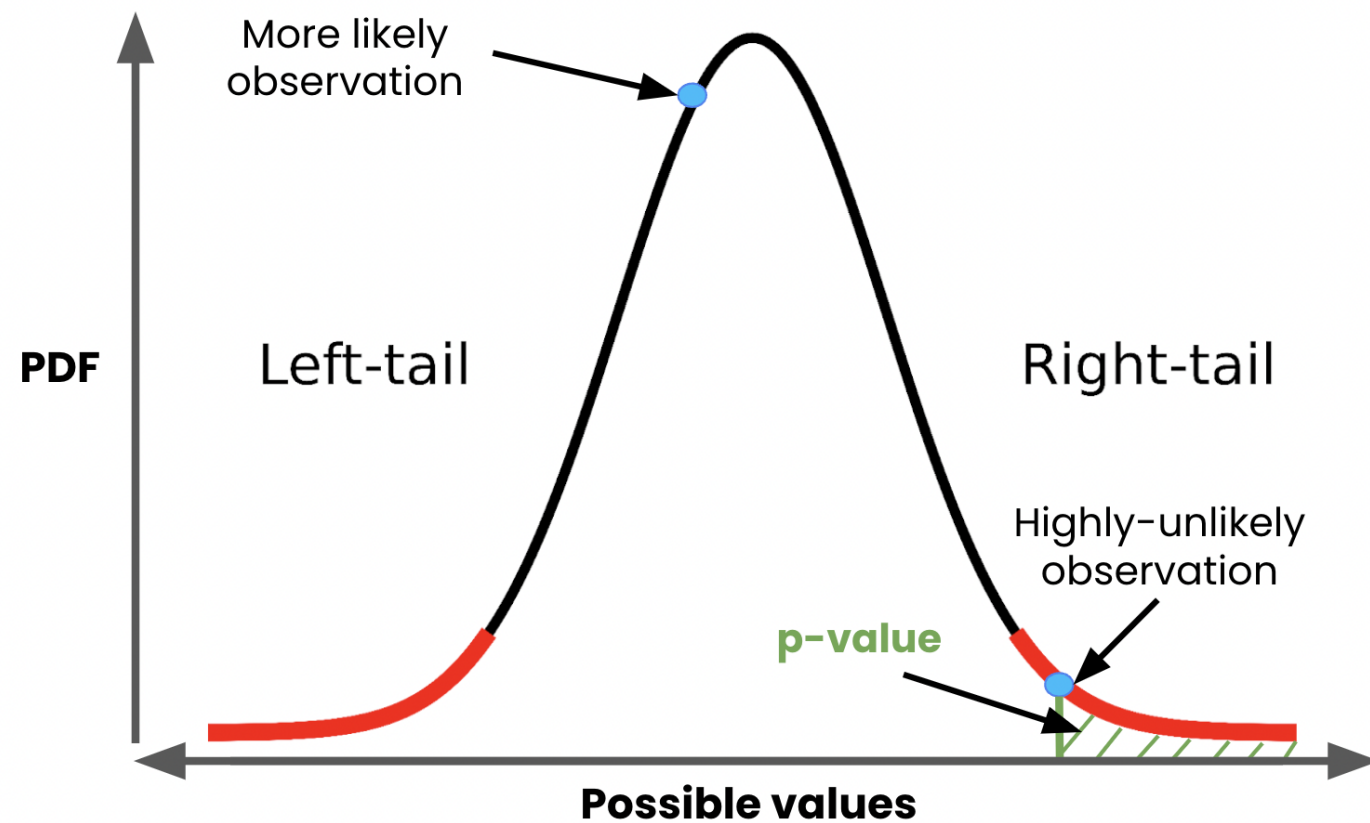
Hypothesis tests check if the sample statistics lie in the tails of the **null distribution**

| <i>Test</i> | <i>Tails</i> |
|--|--------------|
| alternative <i>different from</i> null | two-tailed |
| alternative <i>greater than</i> null | right-tailed |
| alternative <i>less than</i> null | left-tailed |

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

This is a **right-tailed** test

p-values



p-values: probability of obtaining a result, assuming the null hypothesis is true

- Large p-value, large support for H_0
 - Statistic likely **not** in the tail of the *null distribution*
- Small p-value, strong evidence against H_0
 - Statistic likely in the tail of the *null distribution*
- "p" in *p-value* → probability
- "small" means "close to zero"

Calculating the z-score

```
prop_child_samp = (stack_overflow['age_first_code_cut'] == "child").mean()
```

```
0.39141972578505085
```

```
prop_child_hyp = 0.35
```

```
std_error = np.std(first_code_boot_distn, ddof=1)
```

```
0.010351057228878566
```

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

```
4.001497129152506
```

Calculating the p-value

- `norm.cdf()` is normal CDF from `scipy.stats`.
- Left-tailed test → use `norm.cdf()`.
- Right-tailed test → use `1 - norm.cdf()`.

```
from scipy.stats import norm  
1 - norm.cdf(z_score, loc=0, scale=1)
```

```
3.1471479512323874e-05
```

Let's practice!
HYPOTHESIS TESTING IN PYTHON

Statistical significance

HYPOTHESIS TESTING IN PYTHON



James Chapman

Curriculum Manager, DataCamp

p-value recap

- p-values quantify evidence for the null hypothesis
- Large p-value \rightarrow fail to reject null hypothesis
- Small p-value \rightarrow 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 0.2 , 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['age_first_code_cut'] == "child").mean()  
prop_child_hyp = 0.35  
std_error = np.std(first_code_boot_distn, ddof=1)
```

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

```
p_value = 1 - norm.cdf(z_score, loc=0, scale=1)
```

```
3.1471479512323874e-05
```

Making a decision

```
alpha = 0.05  
print(p_value)
```

```
3.1471479512323874e-05
```

```
p_value <= alpha
```

```
True
```

Reject H_0 in favor of H_A

Confidence intervals

For a significance level of α , it's common to choose a confidence interval level of $1 - \alpha$

- $\alpha = 0.05 \rightarrow 95\%$ confidence interval

```
import numpy as np
lower = np.quantile(first_code_boot_distn, 0.025)
upper = np.quantile(first_code_boot_distn, 0.975)
print((lower, upper))
```

```
(0.37063246351172047, 0.41132242370632466)
```

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: data scientists didn't start coding as children at a higher rate

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

- A false negative (Type II) error: data scientists started coding as children at a higher rate

Let's practice!

HYPOTHESIS TESTING IN PYTHON