

# Studio 8 Significance testing and the probability of hypotheses

## 18.05, Spring 2025

### Overview of the studio

In this studio we will be frequentists. But secretly we will know the actual probability of the different hypotheses. This will illustrate how frequentist values like significance and power are NOT probabilities that a hypothesis is true. They are probabilities **assuming** a specific hypothesis is true. That is, they are likelihoods.

More specifically, we will simulate tossing a coin with unknown probability of heads and testing whether the coin is fair or not. We'll call  $\theta$  the unknown probability of heads. In each problem we will have the following:

1.  $H_0$ : The coin is fair. In each problem, we will give the probability of head given  $H_0$ : `theta.H0 = 0.5`
2.  $H_A$ : The coin has has a specified probability of heads `theta.HA`. `theta.HA` will be given as an argument to the problem functions.
3. We will use a secret prior, named `secret.prior`, so we can see how various error rates change with the prior.

The take-away is that numbers like significance give you error rates conditioned on a specific hypothesis. They do not give absolute error rates. That requires a prior which is not available to the Frequentist.

### R introduced in this studio

See the sample code for examples.

We will use `qbinom()` to find rejection region boundaries (critical values). Because the distribution is discrete, it is easy to make an 'off-by-one' error. The sample code shows how to take care to avoid this.

We use the `%in%` operator to check if one value is contained in a list.

We review giving the `sample` function a prescribed sampling distribution.

### Download the zip file

- You should have downloaded the studio8 zip file from our Canvas site.
- Unzip it in your 18.05 studio folder.
- You should see the following R files

`studio8.r`, `studio8-samplecode.r`, `studio8-test.r`

and the following other files

`studio8-instructions` (this file), `studio8-test-answers.html`

## Prepping R Studio

- In R studio, open `studio8-samplecode.r` and `studio8.r`
- Using the Session menu, set the working directory to source file location. (This is a good habit to develop!)
- Answer the questions in the detailed instructions just below. Your answers should be put in `studio8.r`
- [Solution code will be posted on Saturday at 4 am](#)

## Detailed instructions for the studio

- Go through `studio8-samplecode.r` as a tutorial.

## Right-sided rejection regions

To avoid too much fussy coding, we will always require that `theta_HA > 0.5`. This means we will run a **right-sided significance test** with significance level  $\alpha$ . The code we provide will enforce this.

You should understand why that, if `theta_HA < 0.5`, we would need to do a left-sided test.

## Problem 1

**Problem 1.** Here you will finish the code for the function

```
studio8_problem_1(theta_HA, alpha, n.tosses)
```

The arguments to this function are:

`theta_HA` = probability of heads for an unfair coin  
`alpha` = the significance level for the test  
`n.tosses` = the number of coin tosses in one trial

For this problem you are omniscient. You always know the answer. It's just the poor experimenters below you who don't know if they are making errors by rejecting or not rejecting.

There are two types of coins with 0.5 and `theta_HA` probability of heads. An experimenter has one of the coins and wants to determine which type it is.

Null hypotheses  $\mathcal{H}_0$ : the coin is fair, `theta_H0 = 0.5`.

Alternative hypotheses  $\mathcal{H}_A$ : the coin has probability `theta_HA > 0.5` of heads.

Experiment: toss the coin `n.tosses` times

Test statistic:  $x$  = number of heads

For this problem, your code needs to determine the *rejection region*, actual *significance* for this rejection region and *power* of this test. The *power* is defined as  $P(\text{rejection} | H_A)$ , i.e. the probability of a true positive.

Your code should set the corresponding variables in the code to the these values. Note that the rejection region shall be a vector of numbers.

## Problem 2

**Problem 2.** Here you will finish the code for the function

```
studio8_problem_2(theta_HA, alpha, n_tosses, n_trials, secret_prior)
```

The arguments to this function are:

```
theta_HA = probability of heads for an unfair coin
alpha = the significance level for the test
n_tosses = the number of coin tosses in one trial
n_trials = the number of trials to run in the simulation
secret_prior = the secret prior used to pick the type of coin for each trial, i.e.
               secret_prior = c(probability_of_H0, probability_of_HA)
```

As before  $\mathcal{H}_0$  = the coin is fair,  $\mathcal{H}_A$  = the coin has probability  $\text{theta\_HA} > 0.5$  of heads.

One trial consists of:

1. Choosing a type of coin using the `secret_prior`.
2. Tossing it `n_tosses` times and counting the number of heads. The tosses should use the probability appropriate to the type of coin chosen in step 1.
3. Deciding whether or not to reject  $\mathcal{H}_0$ .

Run `n_trials` trials and keep track of

```
the number of rejections, the number of type 1 errors, the number of type 2 errors
number of times  $\mathcal{H}_0$  is chosen, number of times  $\mathcal{H}_A$  is chosen
```

(You can only track errors and the true state of nature because you are omniscient. In real life, researchers can't do this.)

Use your results to estimate each of the following

```
P(rejection |  $\mathcal{H}_0$ ), P( $\mathcal{H}_0$  | rejection), P(rejection |  $\mathcal{H}_A$ ), P( $\mathcal{H}_A$  | rejection)
```

Then set out all of the following information to their corresponding variables: (see the comment in the code)

- `theta_HA`, `alpha`, `n_tosses`, `n_trials`, `secret_prior`
- number of rejections
- number of type 1 errors
- number of type 2 errors
- estimated  $P(\text{rejection} | \mathcal{H}_0)$
- estimated  $P(\mathcal{H}_0 | \text{rejection})$
- estimated  $P(\text{rejection} | \mathcal{H}_A)$
- estimated  $P(\mathcal{H}_A | \text{rejection})$
- estimated  $P(\text{rejection})$

We print out this information at the end of the function for you.

**Problem 3**

In problem 3 you will use your function for problem 2 to explore the difference between  $P(\text{hypothesis} \mid \text{rejection})$  and  $P(\text{rejection} \mid \text{hypothesis})$ .

**Problem 3a.** Here you will finish the code for the function

```
studio8_problem_3a(theta_HA, alpha, n_tosses, n_trials)
```

The arguments to this function are the same as in Problem 2, except they don't include `secret_prior`.

For this problem, the experiment is run with only fair coins, i.e.

```
secret_prior = c(1.0, 0.0) .
```

The code provided specifies the secret prior and calls `studio7_problem2()` for you.

First, call this function with various values of its arguments. Look at the results.

Your job is to modify the cat statements at the end of this function to print a short explanation of the estimated probabilities that go with the given prior.

**Problem 3b.** Here you will finish the code for the function

```
studio8_problem_3b(theta_HA, alpha, n_tosses, n_trials)
```

This problem is identical to 3a, except the prior is set to only produce unfair coins, i.e.

```
secret_prior = c(0.0, 1.0) .
```

**Problem 3c.** Use the function

```
studio8_problem_3c()
```

to print out a few lines explaining the difference between  $P(\mathcal{H}_0 \mid \text{rejection})$  and significance.

**Problem 3d.** Use the function

```
studio8_problem_3d()
```

to have R shout 5 times in all caps, "THE SIGNIFICANCE IS NOT THE PROBABILITY THAT A REJECTION IS IN ERROR!"

Then have it more calmly state once, "It is the probability of rejection given  $\mathcal{H}_0$ . Frequentists don't compute  $P(\text{Error} \mid \text{rejection})$ "

**OPTIONAL Problem 4**

**Optional Problem 4.** This problem is unrelated to the previous problems and will allow you to practice the z-test. Finish the code for the function

```
studio8_problem_4 = function(data, mu0, known_sigma, alpha)
```

The arguments to this function are:

`data` = list of sample values

`mu0` = hypothesized mean in the null hypothesis

`known_sigma` = known value of the standard deviation

`alpha` = significance for the test

In this problem, you are given some data drawn from a normal distribution with a known

standard deviation. The null hypothesis is that the mean of the normal distribution is  $\mu_0$ . The alternative hypothesis is that the mean is not  $\mu_0$ .

Your job is to run a z-test on the data using the given significance. Print out the mean of the data, z-statistic, p-value, and your conclusion on whether or not to reject the null hypothesis.

## Testing your code

For each problem, we ran the problem function with certain parameters. You can see the function call and the output in `studio8-test-answers.html`. If you call the same function with the same parameters, you should get the same results as in `studio8-test-answers.html` – if there is randomness involved the answers should be close but not identical.

For your convenience, the file `studio8-test.r` contains all the function calls used to make `studio8-test-answers.html`.

## Before uploading your code

1. Make sure all your code is in `studio8.r`. Also make sure it is all inside the functions for the problems.
2. Clean the environment and plots window.
3. Source the file.
4. Call each of the problem functions with the same parameters as the test file `studio8-test-answers.html`.
5. Make sure it runs without error and outputs just the answers asked for in the questions.
6. Compare the output to the answers given in `studio8-test-answers.html`.

## Upload your code

Upload your code to Gradescope.

Leave the file name as `studio8.r`.

You can upload more than once. We will grade the last file you upload.

## Due date

**Due date:** The goal is to upload your work by the end of class. If you need extra time, you can upload your work any time before 6 PM ET on the day of the studio (Friday).

**Solutions uploaded:** Solution code will be posted on Canvas at 4 AM the day after the studio.