Week 4: Hypothesis Testing for One/Two Samples

Or, (Theoretical) Populations VS (Actual) Samples

Scott Schwartz

May 18, 2021

(Theoretical) Populations VS (Actual) Samples

Manually flip a coin 10 times and record the outcomes, or...

```
set.seed(130); for(i in 1:10){
  sample(c("H","T"), size=1)#cat(sample(c("H","T"), size=1));cat(" ")
}
```

```
## H H H T H H T T H T
```

Or...

```
sample(c("H","T"), size=10, p=c(1/2,1/2), replace=TRUE)
```

```
##  [1] "T" "H" "T" "H" "H" "T" "T" "T" "H" "T"
```

- `p` *defaults* to "equal chances", so `p=c(1/2,1/2)` isn't strictly required
- Why is `replace=TRUE` *critically important* in conjunction with `size=10`?

(Theoretical) Populations VS (Actual) Samples

Manually flip a coin 10 times and record the outcomes, or...

```
set.seed(130); for(i in 1:10){
  sample(c("H","T"), size=3)
}
```

```
Error in sample.int(length(x), size,
replace, prob) :
cannot take a sample larger than the
population when 'replace = FALSE'
```

```
## [1] " "
```

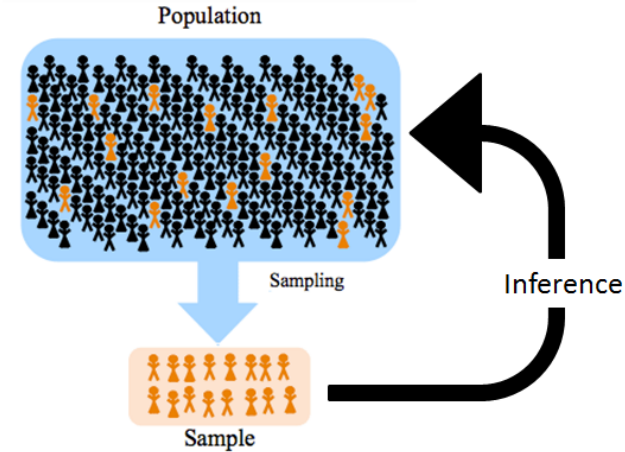
Or...

```
sample(c("H","T"), size=10, p=c(1/2,1/2), replace=TRUE)
```

```
##  [1] "T" "H" "T" "H" "H" "T" "T" "T" "H" "T"
```

- `p` *defaults* to "equal chances", so `p=c(1/2,1/2)` isn't strictly required
- Why is `replace=TRUE` *critically important* in conjunction with `size=10`?

(Theoretical) Populations VS (Actual) Samples



(Theoretical) Populations VS (Actual) Samples

(Theoretical) Populations VS (Actual) Samples

The $\bar{x}$ Sample `mean()` statistic (lower case)

```
n <- 4
set.seed(130)
x <- sample(c("Heads","Tails"), size=n,
            p=c(1/2,1/2), replace=TRUE)

xbar <- mean(x)
```

```
## Warning in mean.default(x): argument is not numeric or logical: returning NA
```

```
mean(as.numeric(factor(x)))
```

```
## [1] 1.75
```

```
xbar <- mean(as.numeric(factor(x))-1)
xbar
```

```
## [1] 0.75
```

```
x
```

```
## [1] "Heads" "Tails" "Tails" "Tails"
```

```
as.factor(x)
```

```
## [1] Heads Tails Tails Tails
## Levels: Heads Tails
```

```
as.numeric(as.factor(x))
```

```
## [1] 1 2 2 2
```

```
as.numeric(as.factor(x))-1
```

```
## [1] 0 1 1 1
```

(Theoretical) Populations VS (Actual) Samples

The *Sampling Distribution* VS $\bar{x}$ the Sample `mean()` statistic (lower case)

```
N <- 100#000000000000?
n <- 10 # <- What's this?
simulated_xbars <- 1:N # <- What's this?
set.seed(130) # <- What happens if this goes inside the for loop?
for(i in 1:N){
  simulated_x <- sample(c("Heads","Tails"), size=n, p=c(1/2,1/2),
                        replace=TRUE)
  simulated_xbar <- mean(2-as.numeric(as.factor(simulated_x)))
                  # mean(as.numeric(as.factor(simulated_x))-1) ?
  simulated_xbars[i] <- simulated_xbar
} # What do we have in `simulated_xbars` once the for loop completes?
```

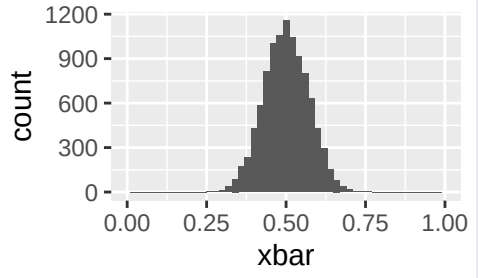
(Theoretical) Populations VS (Actual) Samples

The *Sampling Distribution* VS $\bar{x}$ the Sample `mean()` statistic (lower case)

```
n <- 50; N<-10000; simulated_xbars<-1:N
set.seed(42); for(i in 1:N){
  sim_x <- sample(c("Heads","Tails"),
      size=n, p=c(1/2,1/2), replace=TRUE)
  sim_x <- 2-as.numeric(as.factor(sim_x))
  simulated_xbars[i] <- mean(sim_x)
}
```

```
tibble("xbar"=simulated_xbars) %>%
  ggplot(aes(x=xbar)) +
  xlim(0,1) + geom_histogram(bins=51)
# IGNORE the warning "Removed 2 rows
# containing missing values (geom_bar)."
```

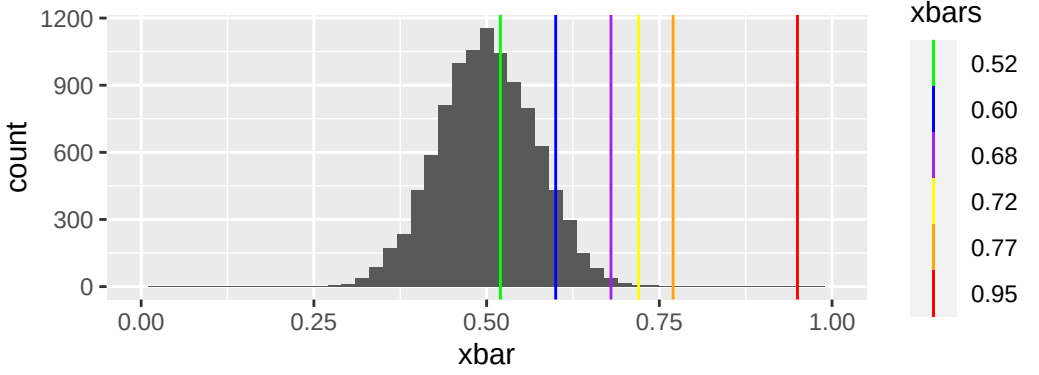
```
# {r, fig.width=2.5, fig.height=1.5}
library(tidyverse)
```



(Theoretical) Populations VS (Actual) Samples

The *Sampling Distribution* VS $\bar{x}$ the Sample `mean()` statistic (lower case)

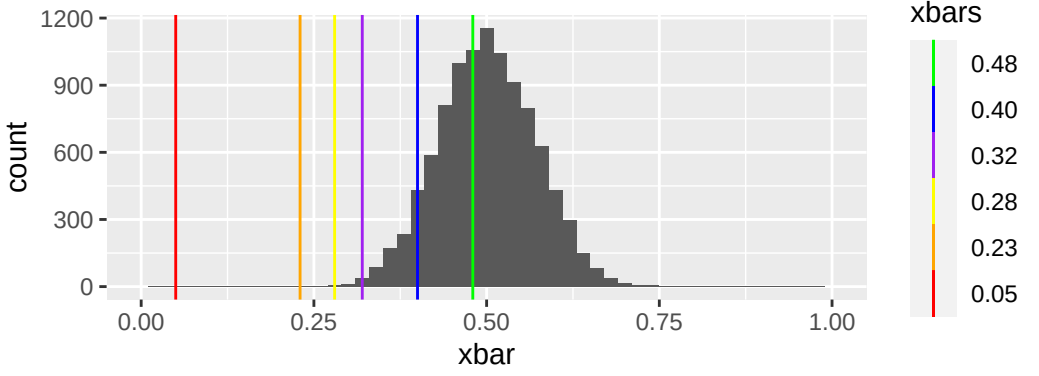
- What would make these observed statistic possible?



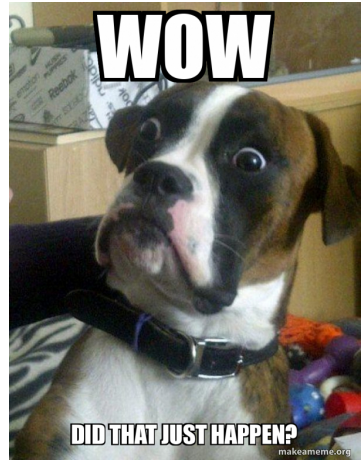
(Theoretical) Populations VS (Actual) Samples

The *Sampling Distribution* VS $\bar{x}$ the Sample `mean()` statistic (lower case)

- What would make these observed statistic possible?

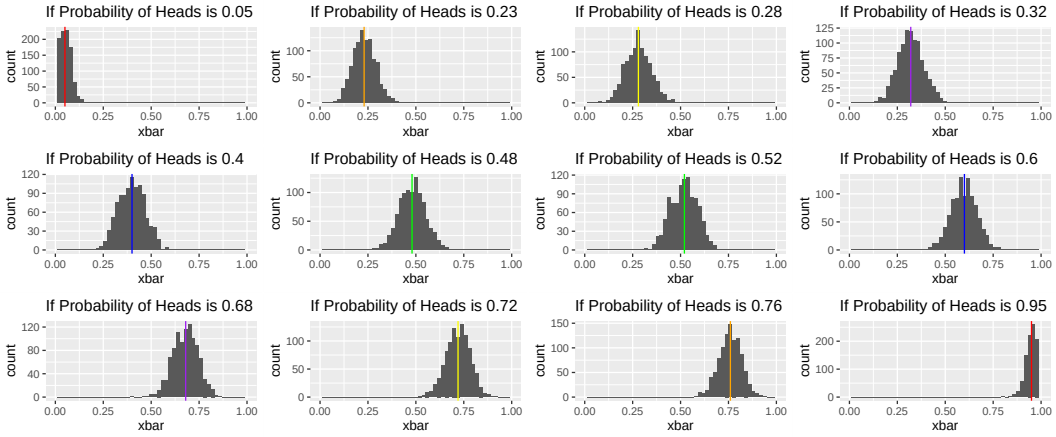


(Theoretical) Populations VS (Actual) Samples



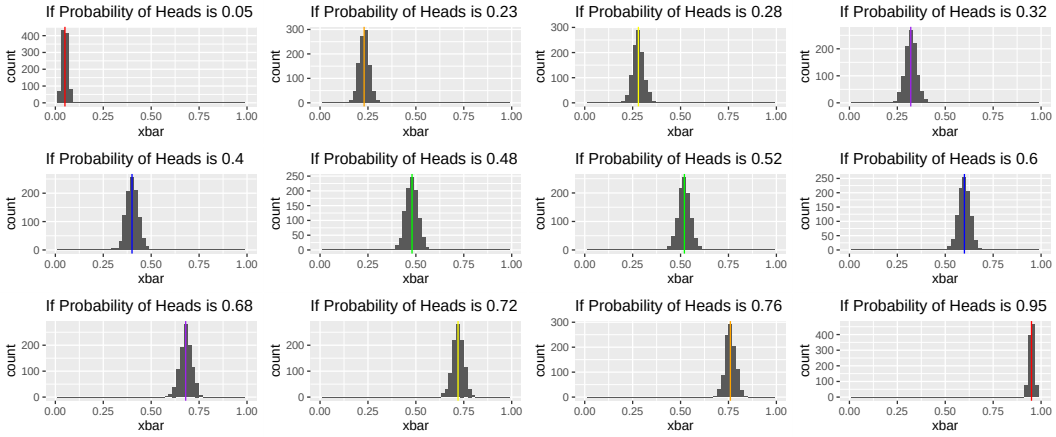
(Theoretical) Populations VS (Actual) Samples

The *Sampling Distribution* VS $\bar{x}$ the Sample `mean()` statistic (lower case)



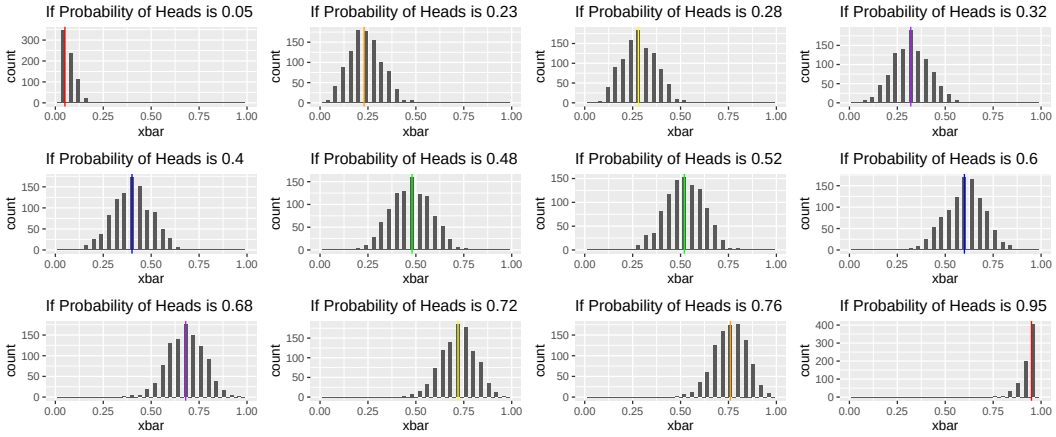
(Theoretical) Populations VS (Actual) Samples

The *Sampling Distribution* VS $\bar{x}$ the Sample `mean()` statistic (lower case)



(Theoretical) Populations VS (Actual) Samples

The *Sampling Distribution* VS $\bar{x}$ the Sample `mean()` statistic (lower case)



Statistical Inference and Hypothesis Testing

Statistical Inference
Can we infer [some specific thing] from the data?

- We'll be doing **Statistical Inference** in a specific way called **Hypothesis Testing**

Hypothesis Testing
Could the observed data be plausibly generated under a given assumption?

- We'll do **Hypothesis Testing** in a specific way with an α-**significance level** test

α-**Significance Level Hypothesis Testing**
α is the probability we make a wrong decision about a chosen assumption.

(Theoretical) Populations VS (Actual) Samples

The *Sampling Distribution* VS $\bar{x}$ the Sample `mean()` statistic (lower case)

The NULL Hypothesis

The *assumed value* of the **parameter**

$$H_0 : p = 0.5$$

implying a **sampling distribution** to be *compared against* the **observed test stat**

The ALTERNATIVE Hypothesis

$H_1 : p \neq 0.5$ or $H_A : p \neq 0.5$

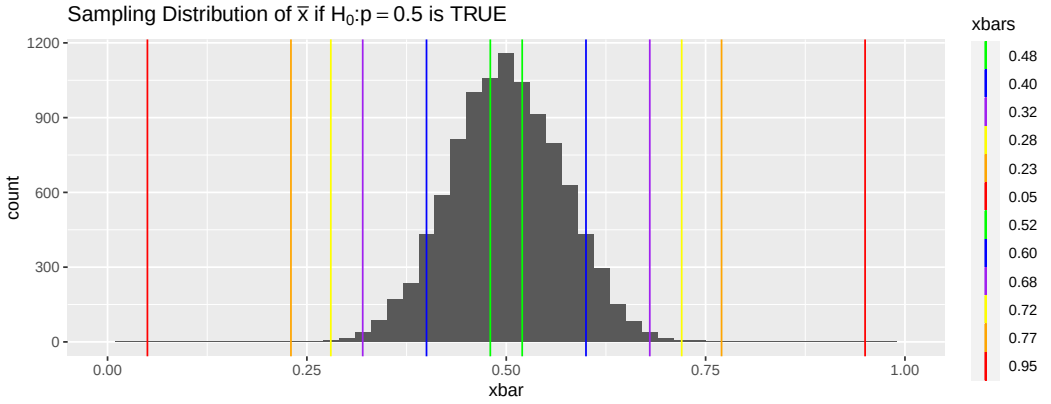
or just $H_1/H_A : H_0$ is `FALSE`

p-value

The probability [*which can be approximated*] of observing a test statistic that is *as or more extreme* than the one we got **if the NULL Hypothesis is actually** `TRUE`

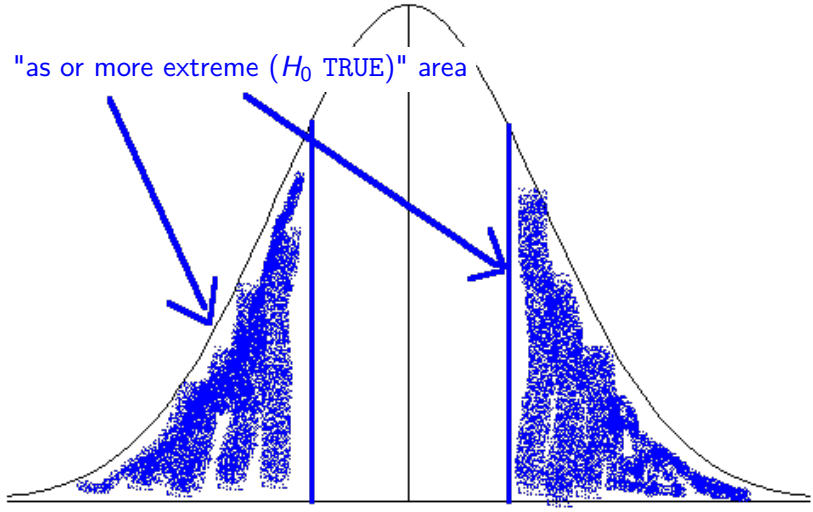
(Theoretical) Populations VS (Actual) Samples

How to measure how *unlikely* these observations are?



DO NOT MESS THIS UP (or else)

p-value: "as or more extreme (H_0 TRUE)" area



DO NOT MESS THIS UP (or else)

p-value

The probability [*which can be approximated*] of observing a test statistic that is *as or more extreme* than the one we got **if the NULL Hypothesis is actually** `TRUE`

Not a p-value: ~~The probability of the Null Hypothesis is TRUE~~

That's not how* Statistical Hypothesis Testing *works...

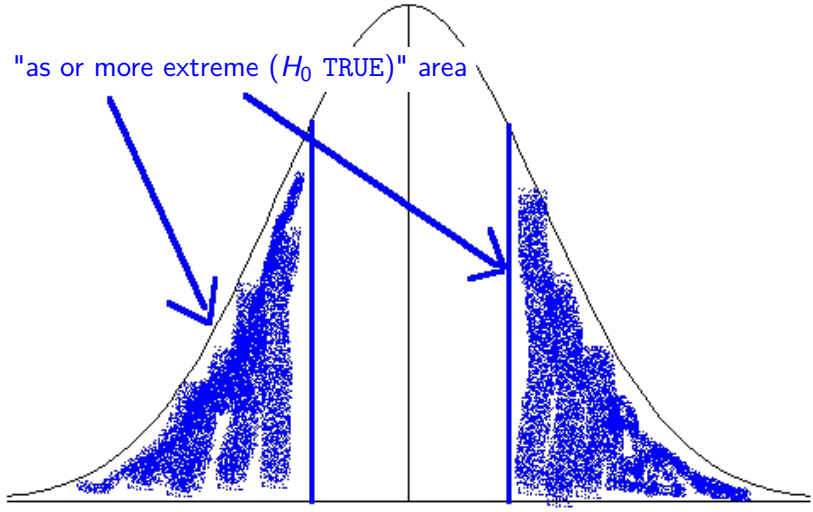
- The **NULL Hypothesis** *IS* either `TRUE` or *IS* `FALSE` (not both)
- The **NULL Hypothesis** can't be *sometimes* `TRUE` and *sometimes* `FALSE`
- The **NULL Hypothesis** can't be `TRUE` *for me* and `FALSE` *for you*

Saying "**I put a** x% **chance on the *Null Hypothesis* being** `TRUE/FALSE`" is
$\longrightarrow$ using probability to express *belief* rather than random chance.

- If you want to use probability to express **belief** then you'll need to be *Bayesian*...

DO NOT MESS THIS UP (or else)

p-value: "as or more extreme (H_0 TRUE)" area



DO NOT MESS THIS UP (or else)

p-value

The probability [*which can be approximated*] of observing a test statistic that is *as or more extreme* than the one we got **if the NULL Hypothesis is actually** `TRUE`

Not a p-value: ~~The probability parameter is the NULL hypothesis value~~

That's not how* Statistical Hypothesis Testing *works...

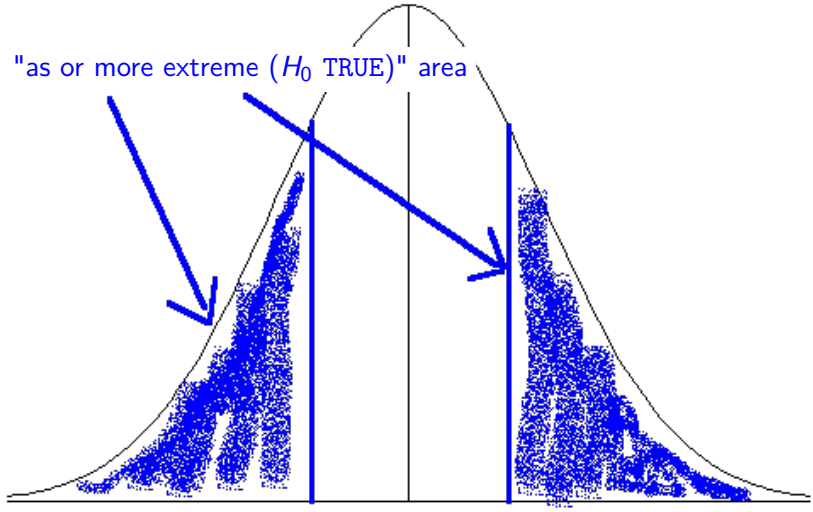
- The **NULL Hypothesis parameter** isn't a "random event"
- The **NULL Hypothesis parameter** doesn't change values at different times
- The **NULL Hypothesis parameter** isn't drawn from some "distribution"

Except if you're *Bayesian*, in which case you model *belief* about **parameters**
$\longrightarrow$ as distributions, and then *do* make probability statements about **parameters**

- but this is a *different* statistical paradigm than **Hypothesis Testing**

DO NOT MESS THIS UP (or else)

p-value: "as or more extreme (H_0 TRUE)" area



(Theoretical) Populations VS (Actual) Samples

The *Sampling Distribution* VS $\bar{x}$ the Sample `mean()` statistic (lower case)

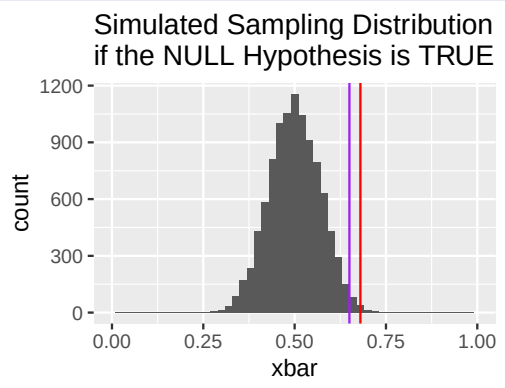
```
n <- 50; N<-10000; simulated_xbars<-1:N
p <- 0.5 # <- This isn't "sometimes 0.5"
# The NULL Hypothesis and n are "fixed"
set.seed(42); for(i in 1:N){
  # Each flip is where there's p "chance"
  x <- sample(c("Heads","Tails"), size=n,
              p=c(p,1-p), replace=TRUE)
  simulated_x<-2-as.numeric(as.factor(x))
  simulated_xbars[i] <- mean(simulated_x)
} # what are the following two values?
mean(abs(simulated_xbars-p)>=abs(0.65-p))
```

```
## [1] 0.0294
```

```
mean(abs(simulated_xbars-p)>=abs(0.68-p))
```

```
## [1] 0.0088
```

```
# {r, fig.width=3.25, fig.height=2.5}
```



Kissing the "Right" Way

⟵ Rodin's sculpture The Kiss

- Güntürkün (2003) recorded how kissing couples tilt their heads.
- 80 out of 124 couples, or 64.5% tilted their heads to the right.
- Would we reject a NULL hypothesis H_0 that the population of humans don't have left or right head tilt tendencies when kissing?

Hypothesis Testing

1. State the **NULL Hypothesis** $H_0 : p = 0.5$ for the *population* [which is?]
   - Assume the value of the **parameter** of the **NULL Hypothesis** is `TRUE`
   - The **ALTERNATIVE Hypothesis** is just that the NULL Hypothesis is `FALSE`
2. Set an **α-significance level** which specifies a "H_0 rejection rule"
   - You will "Reject H_0 at the α-significance level" for **p-values** less than α
   - ***This is also the probability of a Type I error of "rejecting a true*** H_0 ***[Why?]***
3. For the sample size n of the observed **test statistic**
   - Simulate the **Sampling Distribution** asumming the **NULL Hypothesis** is `TRUE`
4. Compute the **p-value** of the **observed test statistic**

   The probability [*which can be approximated*] of observing a test statistic that is *as or more extreme* than the one we got **if the NULL Hypothesis is actually** `TRUE`
5. "**Reject H_0 at the α-significance level**" if the **p-value** is less than α
   - Otherwise, "**Fail to reject H_0 at the α-significance level**"

Type I and II Errors in Hypothesis Testing

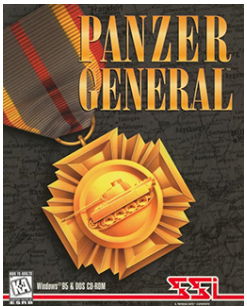
	Innocent	**Guilty**
Convicted	× Oops! Type I	✓ Gotcha! Justice!
Acquitted	✓ Justice! Freedome!	× Oops! Type II

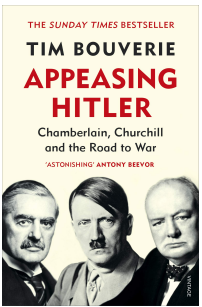
Type I and II Errors in Hypothesis Testing

This is weird, but...

this is how I remember the difference between Type I and Type II Errors...

because...	WW I → **Type I Error**	WW II → **Type II Error**





- WW I wrongly rejected H_0 : *peace* when it shouldn't have → **Type I Error**
- WW II appeasement failed to reject H_0 when it should have → **Type II Error**

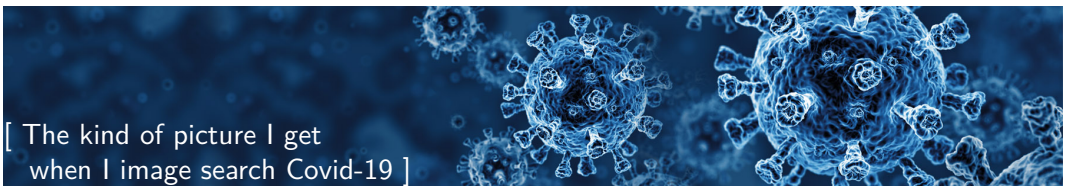
Type I and II Errors in Hypothesis Testing

What's the NULL hypothesis in a Covid Test?

- You don't have Covid? You probably have Covid?
- What are the corresponding Type I and II Errors?

Do people know the difference between covid-19 vs sars-cov 2?

- What test statistic could we use?
- What NULL hypothesis parameter could we use?



[The kind of picture I get
when I image search Covid-19]

Two Sample Hypothesis Testing

Can we follow the above steps for the following H_0?

$$H_0 : p_1 = p_2 \implies H_{1/A} : p_1 \neq p_2$$

- What kind of example problems could this represent? Treatment/Control?

1. [✓] $H_0 : p_1 = p_2 \implies H_0 : p_1 - p_2 = 0$
2. [✓] Choose **significance level** $\alpha = 0.5$
3. [✓] Use ***observed test statistic*** $\bar{x}_1 - \bar{x}_2$ based on n_1 and n_2 samples
   - [?] Simulate the **Sampling Distribution** asumming the **NULL Hypothesis** is `TRUE`
4. [✓] Compute **p-value**
5. [✓] Reject / Fail to reject H_0

Two Sample Hypothesis Testing

```
set.seed(13)
n1 <- 30; n2 <- 40; ns <- paste("n1=", n1, " and n2=", n2, sep="")
x1 <- sample(c(0,1), size=n1, replace=TRUE)
x2 <- sample(c(0,1), size=n2, p=c(1/3,2/3), replace=TRUE)
observed_test_statistic <- mean(x1)-mean(x2); observed_test_statistic
```

```
## [1] -0.2083333
```

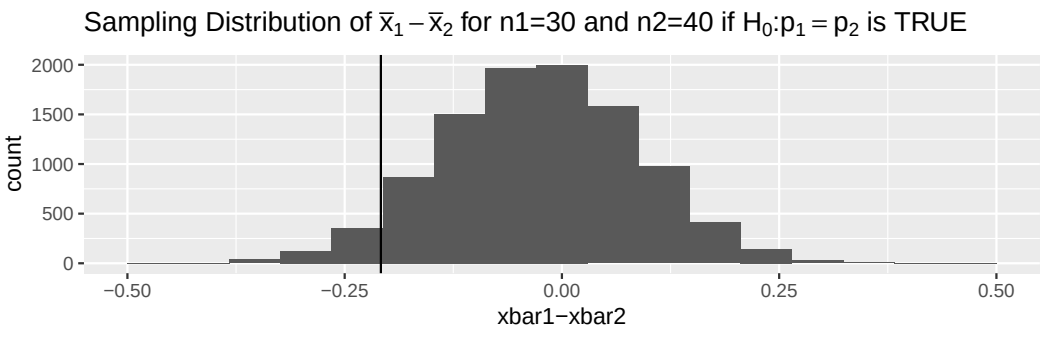
```
N <- 10000; permutation_test_statistics <- 1:N
set.seed(130); for(i in 1:N){
  shuffled_xs <- sample(c(x1,x2), size=n1+n2, replace=FALSE)
  tmp <- mean(shuffled_xs[1:n1])-mean(shuffled_xs[(n1+1):(n1+n2)])
  permutation_test_statistics[i] <- tmp
} # What does `permutation_test_statistics` assume about H_0?
```

```
mean(abs(permutation_test_statistics)>=abs(observed_test_statistic))
```

```
## [1] 0.0701
```

Two Sample Hypothesis Testing

```
# {r, fig.width=6.5, fig.height=2.25}
tibble("xbar1-xbar2"=permutation_test_statistics) %>%
  ggplot(aes(x=`xbar1-xbar2`)) + geom_histogram(bins=18) +
  xlim(-.5,.5) + geom_vline(xintercept=observed_test_statistic) +
  ggtitle(TeX(paste("Sampling Distribution of $\\bar{x}_1 - \\bar{x}_2$ for",
                    ns, "if $H_0: p_1=p_2$ is TRUE")))
```



More General Hypothesis Testing

$\bar{x} = \frac{1}{n}\sum x_i$ VS $\hat{p} = \frac{1}{n}\sum x_i$ (don't confuse with p (don't confuse with *p-value*))

- We've considered $\frac{1}{n}\sum x_i$ when x_i is 0 or 1 with probability p and $1-p$, respectively
  - In this case, we often write $\hat{p}$ instead of $\bar{x}$ since $\hat{p}$, the observed proportion of x_i that are 1, estimates p, **NULL hypothesis parameter** chance that $x_i = 1$

$x_i \sim f(E[x_i] = \mu,\ \theta)$, $E[x_i] = \mu$, and $H_0 : \mu = m_0$ and $H_0 : \mu_1 - \mu_2 = 0$

- *Everything we did also works if x_i has a different distribution of possible values*
  - **Not just when x_i can only be 0 or 1**
  - A Gaussian distribution is a common example: $x_i \sim N\left(E[x_i] = \mu,\ SD[x_i] = \sigma\right)$

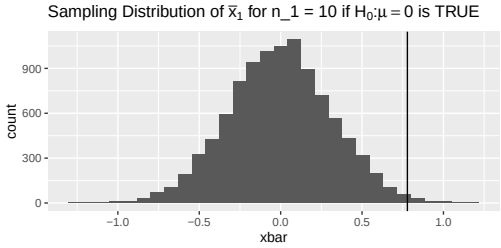
`mean()` VS `median()`, `var()`, etc.

$H_0 : Median = m_0$ and $H_0 : Median_1 = Median_2$ and $H_0 : \sigma^2 = s_0^2$ and $H_0 : \sigma_1^2 = \sigma_2^2$

$$H_0 : \mu = m_0 \quad | \quad x_i \sim N(\mu,\ \sigma) \quad | \quad H_0 : \mu_1 = \mu_2$$

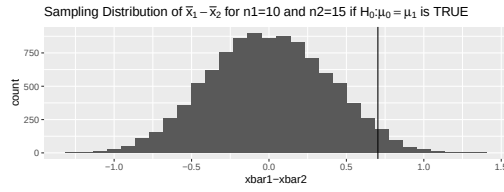
```
n1 <- 10; N<-10000; simulated_xbars<-1:N
set.seed(130); x1 <- rnorm(mean=1, n=n1)
set.seed(42); for(i in 1:N){
  simulated_x <- rnorm(mean=0, n=n1)#H_0
  # sample(c(0,1), size=n, replace=TRUE)
  simulated_xbars[i] <- mean(simulated_x)
}; mean(x1)
```

```
## [1] 0.7785339
```




```
n2 <- 15; permutation_test_statistics<-1:N
set.seed(131); x2 <- rnorm(mean=0, n=n2)
set.seed(43); for(i in 1:N){
  shuffled_xs <- sample(c(x1,x2),
                        size=n1+n2,replace=FALSE)
  tmp <- mean(shuffled_xs[1:n1]) -
         mean(shuffled_xs[(n1+1):(n1+n2)])
  permutation_test_statistics[i] <- tmp
}; mean(x1)-mean(x2)
```

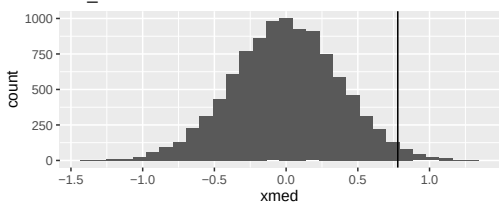
```
## [1] 0.703317
```



H_0: Median=m_0 | 50^{th} %tile | H_0: $Median_1$=$Median_2$

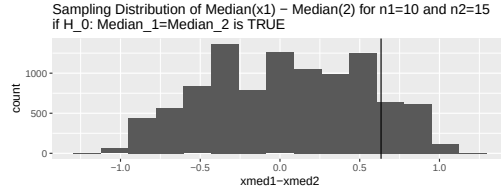
```
n1 <- 10; N<-10000; simulated_xmeds<-1:N
set.seed(130); x1 <- rnorm(mean=1, n=n1)
set.seed(42); for(i in 1:N){
  simulated_x <- rnorm(mean=0, n=n1)#H_0
  # sample(c(0,1), size=n, replace=TRUE)
  simulated_xmeds[i] <- median(simulated_x)
}; median(x1)
```

```
## [1] 0.6916658
```




```
n2 <- 15; permutation_test_statistics<-1:N
set.seed(131); x2 <- rnorm(mean=0, n=n2)
set.seed(43); for(i in 1:N){
  shuffled_xs <- sample(c(x1,x2),
                        size=n1+n2,replace=FALSE)
  tmp <- median(shuffled_xs[1:n1]) -
         median(shuffled_xs[(n1+1):(n1+n2)])
  permutation_test_statistics[i] <- tmp
}; median(x1)-median(x2)
```

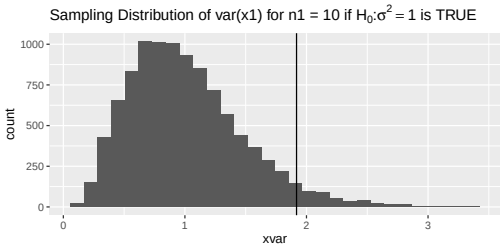
```
## [1] 0.6336835
```



$$H_0 : \sigma^2 = s_0^2 \quad | \quad s^2 = \frac{\sum(x_i - \bar{x})^2}{n-1} \quad | \quad H_0 : \sigma_1^2 = \sigma_2^2$$

```
n1 <- 10; N<-10000; simulated_xvars<-1:N
set.seed(130); x1 <- rnorm(mean=1,sd=1.5,n=n1
set.seed(42); for(i in 1:N){
  simulated_x <- rnorm(mean=0,sd=1,n=n1)#H_0
  # sample(c(0,1), size=n, replace=TRUE)
  simulated_xvars[i] <- var(simulated_x)
}; var(x1)
```

```
## [1] 1.918255
```




```
n2 <- 15; permutation_test_statistics<-1:N
set.seed(131); x2 <- rnorm(mean=0, sd=1.2, n=n
set.seed(43); for(i in 1:N){
  shuffled_xs <- sample(c(x1,x2),
                        size=n1+n2,replace=FALSE)
  tmp <- var(shuffled_xs[1:n1]) -
         var(shuffled_xs[(n1+1):(n1+n2)])
  permutation_test_statistics[i] <- tmp
}; var(x1)-var(x2)
```

```
## [1] 0.8407229
```