do — title: “Simulating_Under_Null_Hypothesis” author: “eCornell” date: “7/1/2021” output: html_document —

Scenario: Use this R Markdown file to help you **simulate the null distribution of a sample statistic**. In this simulation, you will assume that the null hypothesis, which is that both new and old treatments have 50% chance of success (recovery), is true.

This example uses ‘New’ to represent a new drug, and ‘Old’ to represent an old drug in a clinical trial example. Both treatments have two possible outcomes — either the treatment works (is successful) or does not work (is not successful).

Step 1: Set up your code and create colors.

```
# Set the seed so your results are reproducible. The first time you run code after using this command w
set.seed(1)

# Two possible outcomes - the individual recovered or did not recover.
outcome = c("Worked", "Did not Work")

# eCornell Hex Codes:
crimson = '#b31b1b'   # crimson
lightGray = '#cecece' # lightGray
darkGray = '#606366'  # darkGray
skyBlue = '#92b2c4'   # skyblue
gold = '#fbb040'      # gold
ecBlack = '#393f47'   # ecBlack
```

Step 2: Simulate the outcomes of patients under the new treatment.

Create a vector of 52 values sampled from the outcome vector you created above. These values represent whether the treatment caused an individual to recover or not.

```
result_new = sample(outcome, # Vector from which to draw samples
    52, # Number of Samples
    replace = TRUE, # Sample with replacement
    prob = c(0.5, 0.5)) # The probabilities of success and failure for each patient
result_new # Display the result_new vector
```

```
##  [1] "Did not Work" "Did not Work" "Worked"       "Worked"       "Did not Work"
##  [6] "Worked"       "Worked"       "Worked"       "Worked"       "Did not Work"
## [11] "Did not Work" "Did not Work" "Worked"       "Did not Work" "Worked"
## [16] "Did not Work" "Worked"       "Worked"       "Did not Work" "Worked"
## [21] "Worked"       "Did not Work" "Worked"       "Did not Work" "Did not Work"
## [26] "Did not Work" "Did not Work" "Did not Work" "Worked"       "Did not Work"
## [31] "Did not Work" "Worked"       "Did not Work" "Did not Work" "Worked"
## [36] "Worked"       "Worked"       "Did not Work" "Worked"       "Did not Work"
## [41] "Worked"       "Worked"       "Worked"       "Worked"       "Worked"
## [46] "Worked"       "Did not Work" "Did not Work" "Worked"       "Worked"
## [51] "Did not Work" "Worked"
```

```
p_new_sim = mean(result_new == "Worked") # Calculate the proportion of times the New treatment worked
p_new_sim # Display the proportion of times the new treatment worked
```

```
## [1] 0.5384615
```

Step 3: Simulate the outcomes of 52 patients under the old treatment.

Use the same method you used in step 2, but select a sample of 51 individuals.

```
result_old = sample(outcome, 51, replace = TRUE, prob = c(0.5, 0.5))
p_old_sim = mean(result_old == "Worked")
p_old_sim
```

```
## [1] 0.4117647
```

Step 4: Calculate the sample statistic.

The sample statistic is the difference in proportion of times that each treatment was successful.

```
p_diff = p_new_sim - p_old_sim
p_diff
```

```
## [1] 0.1266968
```

Step 5: Run a simulation.

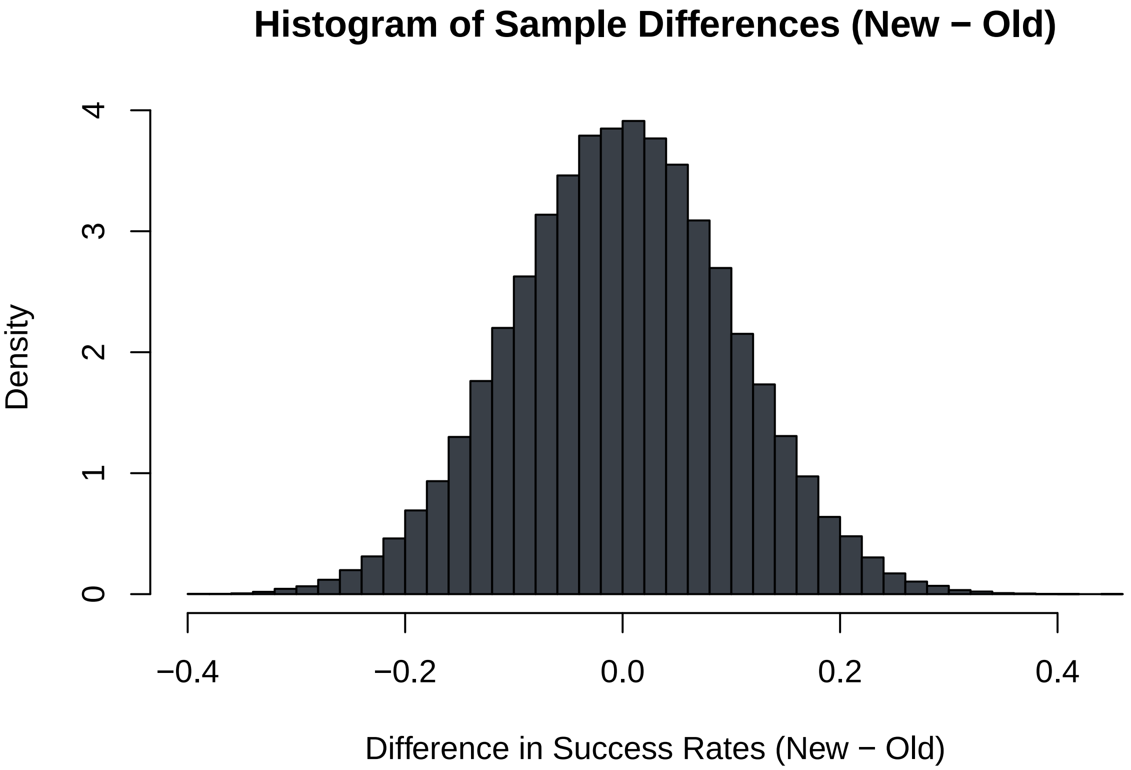
Now, run your simulation by repeating steps 2-4 many, many times.

```
# The number of times you'll repeat the simulation
nsim = 100000
# The vector in which to store the results of different simulations
store_p_diff = rep(0, nsim)

# Create a for loop to tell R to run your simulation 100,000 times
for (i in 1:nsim){

   # Step 2:
    result_new = sample(outcome, 52, replace = TRUE, prob =     c(0.5, 0.5))
    p_new_sim = mean(result_new == "Worked")

   # Step 3:
    result_old = sample(outcome, 51, replace = TRUE, prob =     c(0.5, 0.5))
    p_old_sim = mean(result_old == "Worked")

   # Step 4:
    p_diff = p_new_sim - p_old_sim
    store_p_diff[i] = p_diff

} # Close the for loop
```

Step 6: Visualize the distribution of sample statistics.

Visualize the distribution of sample statistics by creating a histogram.

```
hist(store_p_diff, breaks = 40, freq = FALSE, col = ecBlack,
     main = 'Histogram of Sample Differences (New - Old)',
     xlab = 'Difference in Success Rates (New - Old)')
```



Step 7: Interpret the simulation results.

Calculate the average sample statistic across all 100,000 simulation results.

```r
mean(store_p_diff)
```

```
## [1] -0.0002496908
```

We see that this histogram of sample statistics is centered around 0, but we also see that the difference is as large as 10% or higher in many simulations.

Calculate the number of times the sample statistic was more than +/- 10%. This means the New and Old treatments had success rates that were at least 10% different.

```r
mean(store_p_diff > 0.1 | store_p_diff < -0.1)
```

```
## [1] 0.32239
```

In about 32% of simulations (roughly 1 in 3 simulations), new and old treatment and new treatment had at least a 10% different success rate.

Calculate the standard deviation of the sample statistics.

```r
sd(store_p_diff)
```

```
## [1] 0.0983533
```

We see that on average the difference between the success rates of new and old treatment varied about 9.8% around 0, even though in truth (population) their success rates were the same.