TOOL

Using Simulation to Test a Hypothesis

Data-driven discoveries usually occur when you observe a sample statistic that is larger than you would expect due to random chance. For example, you might find that 70% of patients who take a new experimental drug are cured of a disease, in contrast to the only 30% of people who are cured when they take the current drug treatment. When you find a sample statistic that indicates a new discovery, you should always quantify the uncertainty associated with your results before you generalize your conclusion to a larger population. This will help you assess if your conclusion is real, or just the result of the randomness inherent in taking a small sample from a large population. Data scientists often quantify uncertainty with simulation-based hypothesis tests.

Use this tool as a general guide to testing hypotheses with a simulation-based approach in R.

Step 1: Specify null (H_o) and alternative (H_A) hypotheses. The H_o is the *status quo*, or what you would expect based on your current understanding. The H_A is a new discovery or finding.

Step 2: Construct a null distribution of the sample statistic. Do this by simulating many samples under the assumption that the H_o is correct, and visualizing the distribution of sample statistics with a histogram.

Step 3: Plot the observed sample statistic on the histogram and calculate the p-value. Both the histogram and the p-value indicate the chance of seeing a sample statistic of the size you observed in your sample if the H_o is correct.

Step 4: Use the cut-off value to make a decision. If the p-value is less than the cut-off value, reject the H_o and select the H_A. If the p-value is greater than the cut-off value select the H_o.

Using R With This Tool

The portions of this tool with a grey background are code text you can use to do the examples included in this tool. You can also modify them to use with your own data. In these examples:

- Commands are the lines of code that don't begin with a pound sign (#). Type these lines into R to carry out the command.
- Commented text begins with one pound sign and explains what the code does.
- Code output begins with two pound signs.

Using R to Test a Hypothesis

The steps and code below demonstrate using simulations to test a hypothesis about a Randomized Controlled Trial. Suppose you are interested in determining whether a new treatment for a disease works better than an old treatment for the same disease.

Here is the table of the results of this Randomized Controlled Trial:

	Patients	Worked	Didn't Work	% Success
New	52	27	25	51.9
Old	51	22	29	43.1
Total	103	49	54	47.6

The observed sample statistic from these data is the percent success for the new treatment minus the percent success of the old treatment, which is 8.8%. This sample statistic could indicate that the new treatment is better than the old treatment, but we should check this by testing the uncertainty around this result.

Step 1:

H_o: The new treatment and the old treatment **work equally well** in the population.

H_A: The new treatment **works better** than the old treatment in the population.

Step 2: Construct the null distribution that assumes both new and old treatments have the same chance of success. Based on the table of results above, the total % success is 47.6%. Simulate many samples and record the difference between the success rates of new and old treatments (sample statistic) on each sample. Visualize this null distribution in a histogram using the code below:

```
set.seed(1) # set seed for reproducibility

# Set up this scenario:
outcome = c("Worked", "Did not Work") # Possible outcomes
nsim = 100000                         # Number of iterations
store_p_diff = rep(0, nsim)           # Vector to store results

p_new = 27/52 # Proportion of success with new treatment
p_old = 22/51 # Proportion of success with old treatment
p_all = (22+27)/(51+52)  # Total proportion of success
```

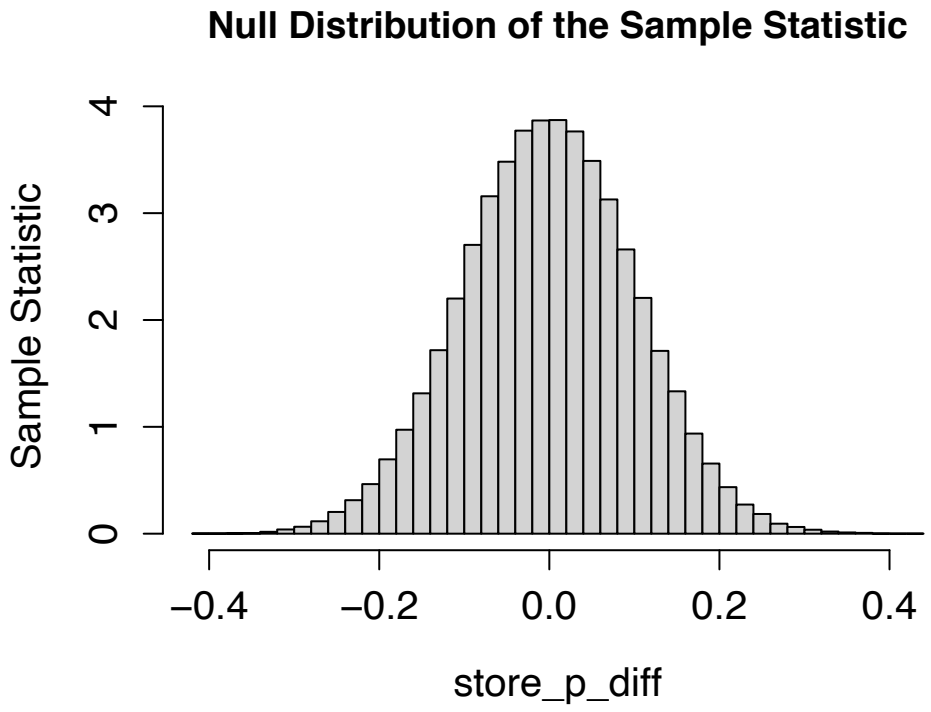
```
# Run simulation:
for (i in 1:nsim){ # Create a for loop

     # Simulate results of the NEW treatment, assuming the probability of
     # success is p_all:
  result_new = sample(outcome, 52, replace = TRUE, prob = c(p_all, 1-p_all))
  p_new_sim = mean(result_new == "Worked")
     # Simulates results of the OLD treatment, assuming the probability of
     # success is p_all:
   result_old = sample(outcome, 51, replace = TRUE, prob = c(p_all, 1-p_all))
   p_old_sim = mean(result_old == "Worked")

     # Calculate and store the sample statistic for this sample iteration:
    p_diff = p_new_sim - p_old_sim
    store_p_diff[i] = p_diff

}

# Draw the histogram:
hist(store_p_diff, breaks = 40, freq = FALSE,
    main = "Null Distribution of the Sample Statistic",
    ylab = "Sample Statistic", col = "lightgrey")
```

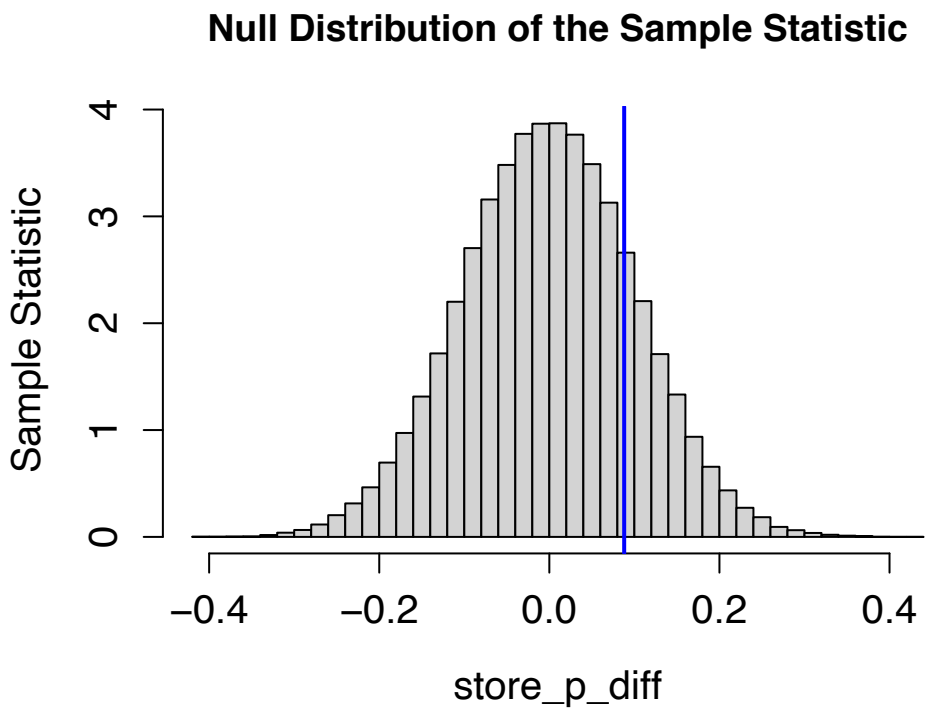


Step 3: Plot the sample statistic on the null distribution and calculate the p-value.

```
# Draw the histogram:
hist(store_p_diff, breaks = 40, freq = FALSE,
    main = "Null Distribution of the Sample Statistic",
    ylab = "Sample Statistic", col = "lightgrey")

# Plot the Observed Statistic:
abline(v = 0.088, lwd = 2, col = "blue")

# Calculate the p-value:
mean(store_p_diff > 0.088)
## [1] 0.18146
```



Step 4: Make a decision. Here, we'll use the standard cut-off value of p-value = 0.05, which keeps the false positive rate at 5%. Based on our simulation, the p-value is 0.18 (18%), which is higher than the cut-off value, so we are unable to reject the null hypothesis. Based on this result, we cannot reject the null hypothesis, so our data do not indicate that the new treatment works better than the old one.