Project 1: Using Simulations To Test Hypotheses

Ken Wood

7/24/2024

When companies want to make improvements to their website or other customer interface, they need to be careful that these improvements result in a better customer experience. A common way to test which of two versions of their website is better is with A/B testing. A/B testing compares two versions of the same digital content to determine which performs better on a set of metrics.

In Part One of the course project, you will calculate the results for an A/B test conducted by a software company that wants to determine which of two web page layouts leads to a higher click rate. In this case, click rate is a metric that measures the proportion of visitors who click on a link to a featured product. Currently, the software company uses layout A, but if layout B has a higher click rate, they will switch to layout B.

To perform the A/B test, the software company runs the following experiment: Layout A is presented to a random set of 100 website visitors, and layout B is presented to a different random set of 100 visitors. The results indicate that 62 of the 100 visitors who were shown layout A clicked on the featured product, whereas 74 of the 100 visitors who were shown layout B clicked on the featured product. Now, you need to evaluate the uncertainty around this finding to determine whether this result seems real, or whether it is due to random chance.

Step 1

Write the null and alternative hypotheses you are evaluating for the A/B test.

H_0 : There is no difference in the click rates generated by Layout A as compared with Layout B.

H_a :: The click rates generated by Layout B are higher than those generated by Layout A.

Step 2

What is the observed sample statistic from the company's A/B test?

$CR(LayoutA) = 62/100 = 0.62$ $CR(LayoutB) = 74/100 = 0.74$

sample_statistic $= CR(LayoutB) - CR(LayoutA) = 0.74 - 0.62 = 0.12$

Step 3

What are the implications of a Type I error (false positive) and a Type II error (false negative) in this context?

If we incorrectly reject H_0 and accept H_a, then we would incur a Type I error. The implication is that we believe Layout B will generate a higher click rate than Layout A when, in fact, the difference in click rates between the two layouts is not statistically significant.

If we incorrectly accept H_0 and reject H_a, then we would incur a Type II error. The implication is that we believe the difference in click rates between Layout A and Layout B is not statistically significant when, in fact, the difference in click rates between the two layouts is statistically significant.

Step 4

Complete the following starter code to construct the null distribution of the sample statistic (click rate of layout B – click rate of layout A). Complete the five lines that have the comment # COMPLETE after them.

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

# Simulate null distribution of statistic
# Both layouts A and B have same chance of success: (74+62)/(100+100) = 68%
set.seed(1)

outcome = c("Clicked", "Did not Click")
nsim = 100000
store_p_diff = rep(0, nsim)

p_new =  62/100  # Observed click rate of Layout A
p_old =  74/100  # Observed click rate of Layout B
p_all =  (62+74)/(100+100) # Combined click rate under the null hypothesis = 68%

for (i in 1:nsim){
 result_new =  sample(outcome, 100, replace = TRUE, prob = c(p_all, 1-p_all))
 p_new_sim = mean(result_new == "Clicked")

 result_old = sample(outcome, 100, replace = TRUE, prob = c(p_all, 1-p_all))
 p_old_sim = mean(result_old == "Clicked")

 p_diff = p_new_sim - p_old_sim
 store_p_diff[i] = p_diff

}
```

Step 5

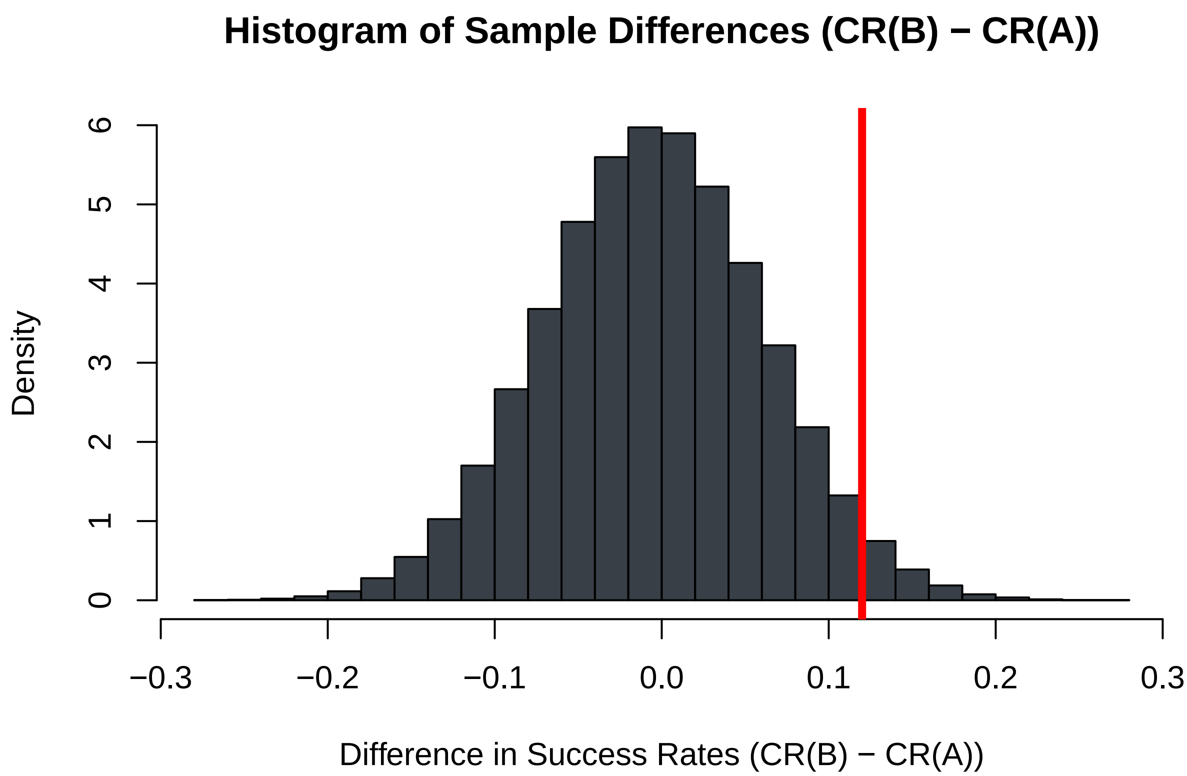
Calculate the mean and standard deviation of the null distribution of the sample statistic you created in Question 4.

```r
mean = mean(store_p_diff)
std_dev = sd(store_p_diff)
```

Step 6

Draw a histogram of the null distribution of the sample statistic. Plot the observed statistic over the histogram.

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

abline(v = 0.12, lwd = 4, col = "red")
```



Step 7

Calculate the p-value of the observed statistic. Interpret this p-value in the context of this problem. Based on the usual accepted cut-off value of 0.05, would you select the null hypothesis, or reject it in favor of the alternative hypothesis? Briefly explain your decision.

```
# calculate p-value
mean(store_p_diff > 0.12) # = 0.03
```

```
## [1] 0.02912
```

Since the p-value < 0.05, we would reject the null hypothesis, H_0. We conclude this because, if the null hypothesis were true (both Layout A and Layout B had the same 68% click rate in population), there would be only a 3% chance that the difference in click rates is 0.12 or higher. So it does look like there is strong evidence in favor of the alternative hypothesis, H_a.

Step 8

Suppose the company decides that they would like to choose layout B if layout B is at least 8% better than layout A. What are the chances of obtaining a false positive with this decision rule?

Looking at the power curve, for a signal value of 0.08, the corresponding power is approximately 0.3. Therefore, there is a 30% chance of obtaining a false positive.

Step 9

Suppose you want to control the false positive rate at 5%. What cut-off value should you use?

Looking at the power curve, there is a dotted horizontal line plotted for power = 0.05. The corresponding signal is 0.025 or 2.5%, which is our cut-off value.

Step 10

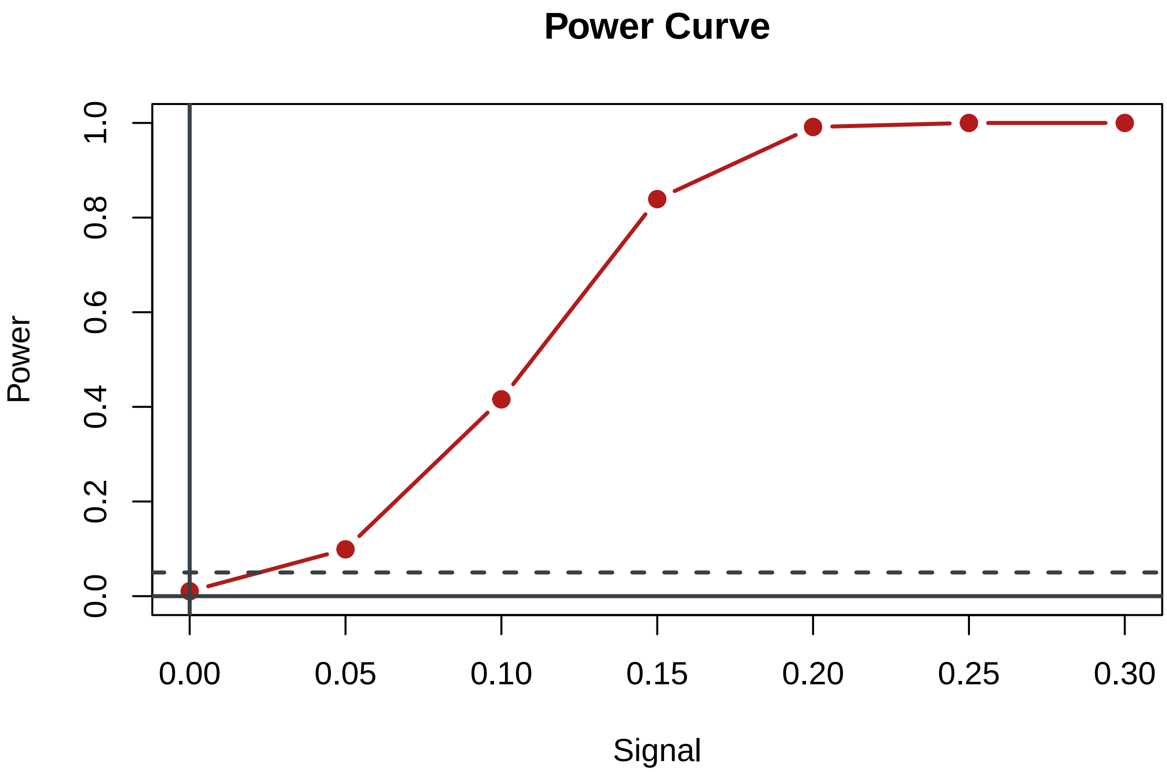
The following code chunk draws a power curve for these data. Run the code to create the power curve. Then, use the power curve to determine how often you would be able to detect that layout B is better than layout A under the following conditions: - you choose 11% as the cut-off value, and - layout B is 10% better than layout A.

```
# R function to calculate power for a desired signal level:
calc_power <- function(delta){

set.seed(1)
outcome = c("Clicked", "Did not Click")
nsim = 100000
store_p_diff = rep(0, nsim)

p_new = 74/100
p_old = 62/100
p_all = (74+62)/(100+100)

for (i in 1:nsim){
 result_new = sample(outcome, 100, replace = TRUE, prob = c(p_all+delta, 1-p_all-delta))
 p_new_sim = mean(result_new == "Clicked")

 result_old = sample(outcome, 1000, replace = TRUE, prob = c(p_all, 1-p_all))
 p_old_sim = mean(result_old == "Clicked")

 p_diff = p_new_sim - p_old_sim
 store_p_diff[i] = p_diff
}

return(mean(store_p_diff > 0.11))
}
# Use this function to calculate power for signal levels 5%, 10%, 15%, ...
delta_seq = seq(0, 0.3, by=0.05)
power_seq = rep(0, length(delta_seq))

for (i in 1:length(delta_seq)){
  delta = delta_seq[i]
  power_seq[i] = calc_power(delta = delta)
  print(paste('signal = ', delta, '; power = ', round(power_seq[i], 4)))
}
```

```
## [1] "signal =  0 ; power =  0.0103"
## [1] "signal =  0.05 ; power =  0.099"
## [1] "signal =  0.1 ; power =  0.4158"
## [1] "signal =  0.15 ; power =  0.839"
## [1] "signal =  0.2 ; power =  0.9915"
## [1] "signal =  0.25 ; power =  1"
## [1] "signal =  0.3 ; power =  1"
```

```
# Plot the power curve:
plot(delta_seq, power_seq, type = 'b', pch = 19, lwd = 2, col = crimson,
     xlab = 'Signal', ylab = 'Power', main = 'Power Curve',
     ylim = c(0,1), xlim = c(0, 0.3))
abline(h = 0.05, col = ecBlack, lwd = 2, lty = 2)
abline(h = 0, col = ecBlack, lwd = 2, lty = 1)
```

```r
abline(v = 0, col = ecBlack, lwd = 2, lty = 1)
```



We choose 11% as the cut-off value:

There is a 50% chance of finding that layout B is better than layout A with a cut-off rule that controls the chance of a false positive at 5%.

Layout B is 10% better than layout A.

With a signal = 0.1, there is a 42% of finding that Layout B is 10% better than layout A.

This is the end of Part One of the course project.

Don't forget to submit your work!