

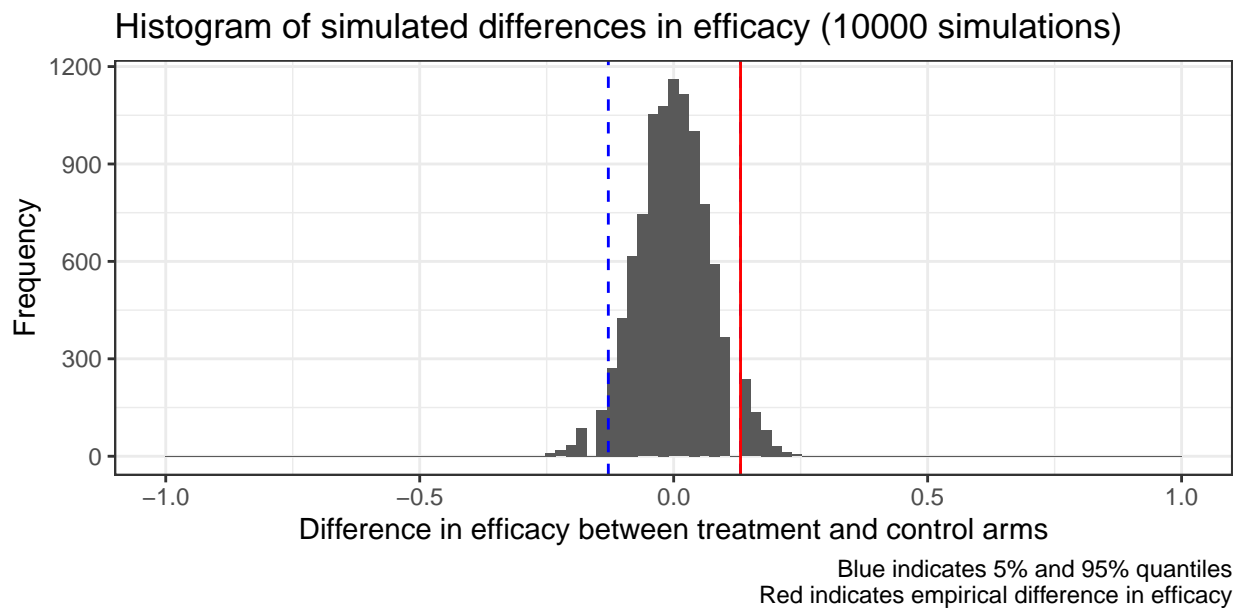
BIOST 544: Homework 2

Department of Biostatistics @ University of Washington

October 22, 2024

1. Suppose we are comparing a new treatment to standard-of-care. We run a randomized clinical trial (with basic simple randomization), and would like to assess if treatment is more effective than control (using a binary outcome — response vs non-response). Suppose efficacy is assessed by the difference in response proportion between new-treatment and standard-of-care.

Write a function that takes in data from a clinical trial run as above (a matrix with two columns: `tx` for treatment assignment, and outcome, with `outcome=1` indicating response), and runs a permutation/re-randomization test to evaluate if the data are consistent with the hypothesis that standard-of-care is at least as effective as new-treatment.



The histogram displays a permutation test assessing treatment efficacy using example data. The observed difference in efficacy, estimated by the proportion of respondents in the treatment and control arms, is 0.132. After 10^4 re-randomizations of treatment assignments, we evaluate the likelihood of observing this statistic under the null hypothesis of identical efficacy. With a right-sided α of 5%, the p-value of 0.027 suggests marginal evidence that the treatment is more effective than the standard of care.

See function `perm_test()` and its many helper functions in the Code Appendix at the end of this report.

2. Consider the following adaptive randomization scenario: Suppose we have pre-clinical evidence that a new treatment is much more effective than standard-of-care. Given this, it may not be ethical to randomize a ton of patients to standard-of-care. However, preclinical evidence is not always to be believed.

We decide to run a trial in which patients are enrolled one-at-a-time (sequentially). Each time a patient is enrolled, they are randomized to new treatment/control and then outcome is observed (all

before the next patient is enrolled). Furthermore, we change our randomization probabilities after each outcome is observed using the scheme in Figure 1– where p_{new} is our randomization probability for the new-treatment arm; with $\text{success}_{\text{new}}$ the number of successes on the new-treatment arm (up to that point in the trial); $\text{failures}_{\text{old}}$ the number of failures on the control-arm; and totalpatients the total number of patients enrolled (up to that point in the trial).

As patients are enrolled, randomized, and outcomes are recorded, p_{new} will change. This adaptation strategy was proposed and discussed (in more generality) in Wei and Durham (1978). It is known as a randomized play the winner rule. At the end of the trial, we assess effectiveness of treatment by looking at the difference between proportion of successes on the new treatment, and successes on the control, calculated on all the patients from the trial.

$$p_{\text{new}} \leftarrow \frac{1 + 3(\text{success}_{\text{new}} + \text{failures}_{\text{old}})}{2 + 3 \cdot \text{totalPatients}}$$

Figure 1: Formula for randomization probabilities.

- (a) Write a function that will take in data from a trial run as above (i.e., a matrix with one column `tx` of treatment assignments, one column `outcome` of binary outcomes, and one column `order` of enrollment order) and run a re-randomization test. The test should assess if the observed difference in response proportions is consistent with the hypothesis that standard-of-care is at least as effective as the new-treatment.

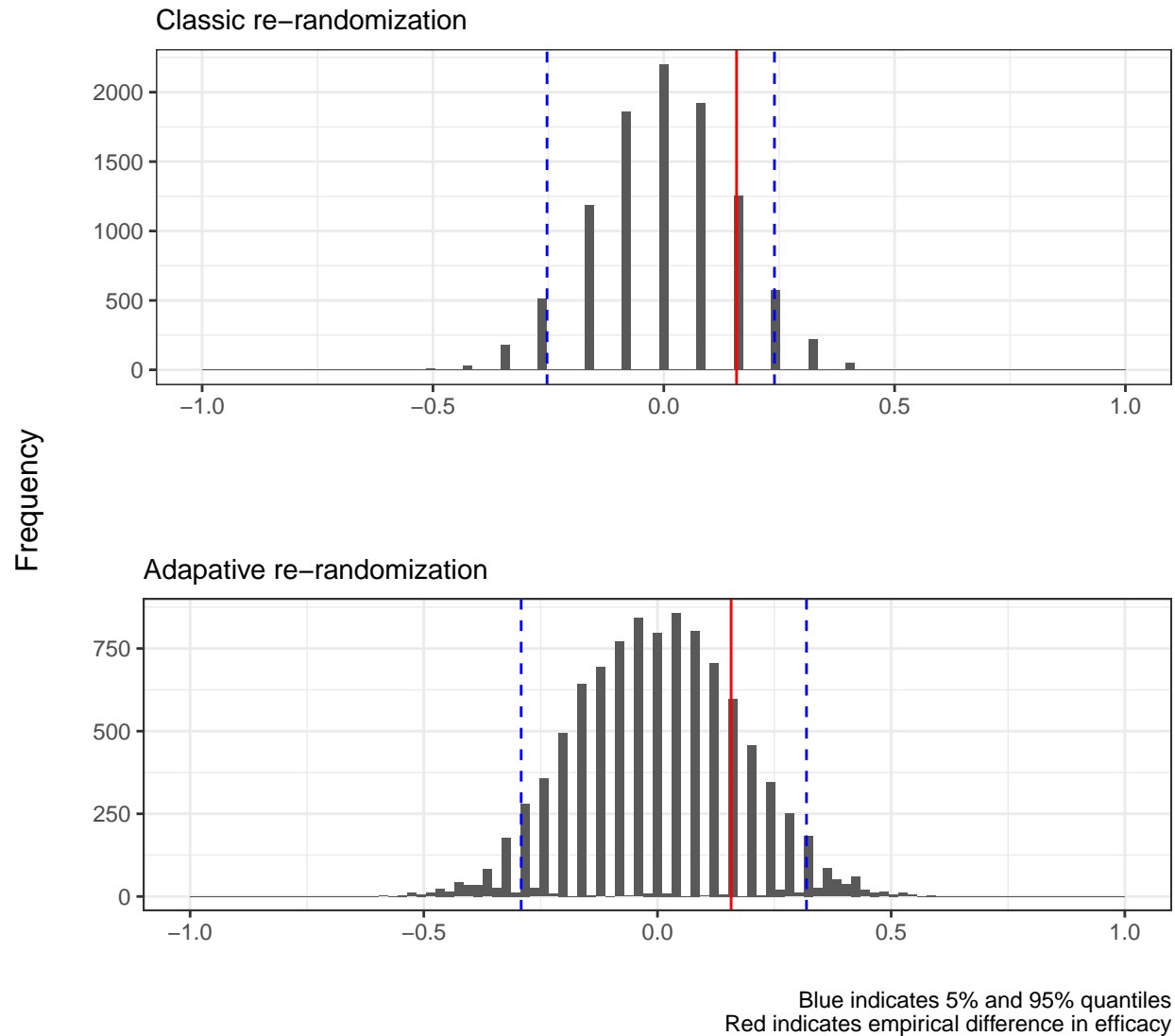
See function `permute_treatment()`, in the Code Appendix at the end of this report, which has a parameter that toggles adaptive and classic re-randomization of treatment.

- (b) Read in the data “HW2-adaptive-trial.txt” from the website; and, using your function from 2a, evaluate if the data are consistent with the hypothesis that standard-of-care is at least as effective as the new treatment.

Using a permutation test on the trial data, the observed efficacy difference, estimated by the proportion of respondents in the treatment/control arms, is 0.158. Based on 10^4 adaptive re-randomizations (see Figure 1), we assess the likelihood of this statistic under the null hypothesis of identical efficacy. With a right-sided α of 5%, the p-value of 0.204 indicates insufficient evidence that the treatment is more effective than the standard of care.

3. Now suppose a collaborator provided the data in “HW2-adaptive-trial.txt”, but forgot to say that they were generated using an adaptive trial. In this case we would likely accidentally analyze our data using the function written in (1). How does the sampling distribution of $\hat{\pi}_{\text{new}} - \hat{\pi}_{\text{old}}$ generated from the simple permutations in (1) compare to the sampling distribution generated using our adaptive re-randomization from the function in (2a)?

Histogram of simulated differences in efficacy (10000 simulations)



Difference in efficacy between treatment and control arms

The observed difference in efficacy, estimated by the proportion of respondents in the treatment/control arms, is 0.158. Based on 10^4 re-randomizations, we test for identical efficacy between the new treatment and standard of care. With a right-sided α of 5%, the p-value of 0.085 indicates insufficient evidence that the treatment is more effective.

This non-adaptive test reaches the same conclusion as the adaptive test, but its p-value suggests the observation is less consistent with the null hypothesis compared to the adaptive test's p-value.

End of report. Code appendix begins on the next page.

Code Appendix

```
# clear environment
rm(list = ls())

# setup options
knitr::opts_chunk$set(echo = FALSE, message = FALSE)
options(knitr.kable.NA = '-', digits = 3)
labs <- setdiff(knitr::all_labels(), c("setup", "llm_appendix", "allcode"))

# load relevant packages
library(dplyr)      # data frame manipulation
library(ggplot2)    # plotting
library(gridExtra)  # grid of plots

## =====
## Question 1
## =====

## functions that analyze clinical trial data and run a permutation test to evaluate if the data is con

# helper function to compute a summary statistic of interest
calc_summary_stat <- function(data) {
  # split cases and controls
  data.cases = subset(data, tx==1)
  data.controls = subset(data, tx==0)
  # compute difference in efficacy between cases and controls
  summary.stat = mean(data.cases$outcome) - mean(data.controls$outcome)
  return(summary.stat)
}

# helper function to re-randomize assigned treatment
permute_treatment <- function(data, adaptive=FALSE) {
  perm.data = data
  # adaptive randomization (for Question 2)
  if (adaptive) {
    s_new = 0 # variables for adaptive treatment assignment formula
    f_old = 0
    totalP = 0
    # iterate over patients--in order--and re-assign treatment
    for (i in perm.data$order) {
      currentP = perm.data[i,] # current patient data
      p_new = (1+3*(s_new+f_old)) / (2+3*totalP) # probability of receiving treatment
      # assign current patient to treatment or control arm
      tx_new = rbinom(1, 1, p_new)
      currentP$tx = tx_new
      # update number of success among treated and failures among controls
      s_new = ifelse(currentP$tx & currentP$outcome, s_new+1, s_new)
      f_old = ifelse(!currentP$tx & !currentP$outcome, f_old+1, f_old)
      totalP = totalP+1 # increment number of assigned patients
      # insert updated patient data
      perm.data[i,] = currentP
    }
  }
}
```

```

}
# classical randomization (for Question 1)
else perm.data$tx = sample(data$tx) # re-randomize assigned treatments
# return data with re-randomized treatments
return(perm.data)
}

# helper function to simulate one permutation of the data
sim_one_trial <- function(data, adaptive=FALSE) {
  perm.data = permute_treatment(data, adaptive) # permute assigned treatment
  summary.stat = calc_summary_stat(perm.data)
  # return a summary statistic
  return(summary.stat)
}

# function to conduct a permutation test, with an option to use an adaptive treatment assignment formula
perm_test <- function(data, adaptive=FALSE) {
  # simulate a sampling distribution of a summary statistic with many permutations
  n.permutations = 10000
  permutations = replicate(n.permutations, sim_one_trial(data, adaptive))
  # compute the right-sided p-value of our empirical statistic
  empirical.stat = calc_summary_stat(data)
  pval = mean(permutations > empirical.stat)
  # return test results
  return(list(output=permutations, n=n.permutations,
             empirical.stat=empirical.stat, pval=pval))
}

## helper function to visualize a permutation test
plot_perm_test <- function(perm.test, plot=TRUE) {
  simulations = perm.test$output
  quantiles = quantile(simulations, probs=c(0.05, 0.95))
  title = paste("Histogram of simulated differences in efficacy (", perm.test$n,
               " simulations)", sep="")
  caption = "Blue indicates 5% and 95% quantiles\nRed indicates empirical difference in efficacy"
  # plot the observed statistic with the simulated sampling distribution
  gg = ggplot() +
    # sampling distribution with the 5% and 95% quantiles
    geom_histogram(aes(simulations), bins = 100) +
    geom_vline(xintercept=quantiles, linetype=2, col="blue") +
    # the observed statistic
    geom_vline(xintercept=perm.test$empirical.stat, col="red") +
    # pageantry
    labs(title=title, caption=caption) +
    xlab("Difference in efficacy between treatment and control arms") +
    ylab("Frequency") +
    xlim(-1,1) +
    theme_bw()
  if (plot) print(gg) # optional printing
  # return generated plot
  return(gg)
}

```

```

## conduct a permutation test of treatment efficacy using example data
# load example data
nsclc <- read.table("../data/nsclc-modified.txt")
test.data <- nsclc %>% select(tx=response,outcome=survival.past.400)

# simulate a sampling distributions of the summary statistic using permutations
set.seed(1)
perm.test <- perm_test(test.data)
# get empirical statistic and right-sided p-value
emp.stat <- perm.test$empirical.stat
pval <- perm.test$pval
# visually compare our observed value to the simulated sampling distribution
gg <- plot_perm_test(perm.test, plot=TRUE)

## =====
## Question 2
## =====

## see function `permute_treatment()`, which has a parameter that toggles adaptive
## or classic re-randomization of treatment

## conduct an adaptive re-randomization test of treatment efficacy
adptv <- read.delim("../data/HW2-adaptive-trial.txt", sep=",")
# simulate a sampling distributions of the statistic using permutations
set.seed(1)
perm.test.adapt <- perm_test(data=adptv, adaptive=TRUE)
# get empirical statistic and right-sided p-value
emp.stat.adapt <- perm.test.adapt$empirical.stat
pval.adapt <- perm.test.adapt$pval
# save plot of our observed value on the simulated sampling distribution
gg.adapt <- plot_perm_test(perm.test.adapt, plot=FALSE)

## =====
## Question 3
## =====

## conduct a classic re-randomization test of treatment efficacy
# simulate a sampling distributions of the summary statistic using permutations
set.seed(1)
perm.test <- perm_test(data=adptv) # "adaptive=FALSE" is a default argument
# get empirical statistic and right-sided p-value
emp.stat <- perm.test$empirical.stat
pval <- perm.test$pval
# save plot of our observed value on the simulated sampling distribution
gg <- plot_perm_test(perm.test, plot=FALSE)

## plot results of classic and adaptive re-randomization tests
gridExtra::grid.arrange(
  # ignore redundant titles
  gg + labs(title="", caption="", subtitle="Classic re-randomization") +
    xlab("") + ylab(""),
  gg.adapt + labs(title="", subtitle="Adapative re-randomization") +
    xlab("") + ylab(""),

```

```
top=gg$labels$title, left=gg$labels$y, bottom=gg$labels$x)
```

End of document.