Gerber and Green (2012) Chapter 13 Problem 1

Margaret Moor and Alexander Coppock, Yale University

January 30, 2018

This script shows how to conduct the randomization inference procedure in Gerber and Green (2012) Chapter 13 Problem 1 three different ways: using the `ri2` package, using the `ri` package, and by hand with a loop.

Chapter 13 Problem 1

Middleton and Rogers report the results of an experiment in which ballot guides were mailed to randomly assigned precincts in Oregon prior to the 2008 November election. The guides were designed to encourage voters to support certain ballot measures and oppose others. The dataset contains election results for 65 precincts, each of which contains approximately 550 voters. The outcome measure is the number of net votes won by the sponsors of the guide across the four ballot measures that they endorsed or opposed. The treatment is scored 0 or 1, depending on whether the precinct was assigned to receive ballot guides. A prognostic covariate is the average share of the vote cast for Democratic candidates in 2006.

(a) Estimate the average treatment effect, and illustrate the relationship between treatment and outcomes graphically using an individual values plot.
(b) Interpret the graph in part (a).
(c) Use randomization inference to test whether the apparent difference-in-means could have occurred by chance under the sharp null hypothesis of no treatment effect for any precinct. Interpret the results.
(d) Suppose it were the case that when randomly assigning precincts, the authors used the following screening procedure: no random allocation was acceptable unless the average 2006 Democratic support score in the treatment group was within 0.5 percentage points of the average 2006 Democratic support score in the control group. Do all subjects have the same probability of being assigned to the treatment group? If not, re-estimate the ATE, weighting the data as described in Box 4.5. Redo your hypothesis test in part (c) subject to this restriction on the randomization. Interpret the results.

SHOWN BELOW

```
# Data from http://isps.yale.edu/FEDAI
library(haven)
data13.1 <- read_dta("datasets/13.1.dta")

# Number of sims the same for both methods
sims <- 1000
```

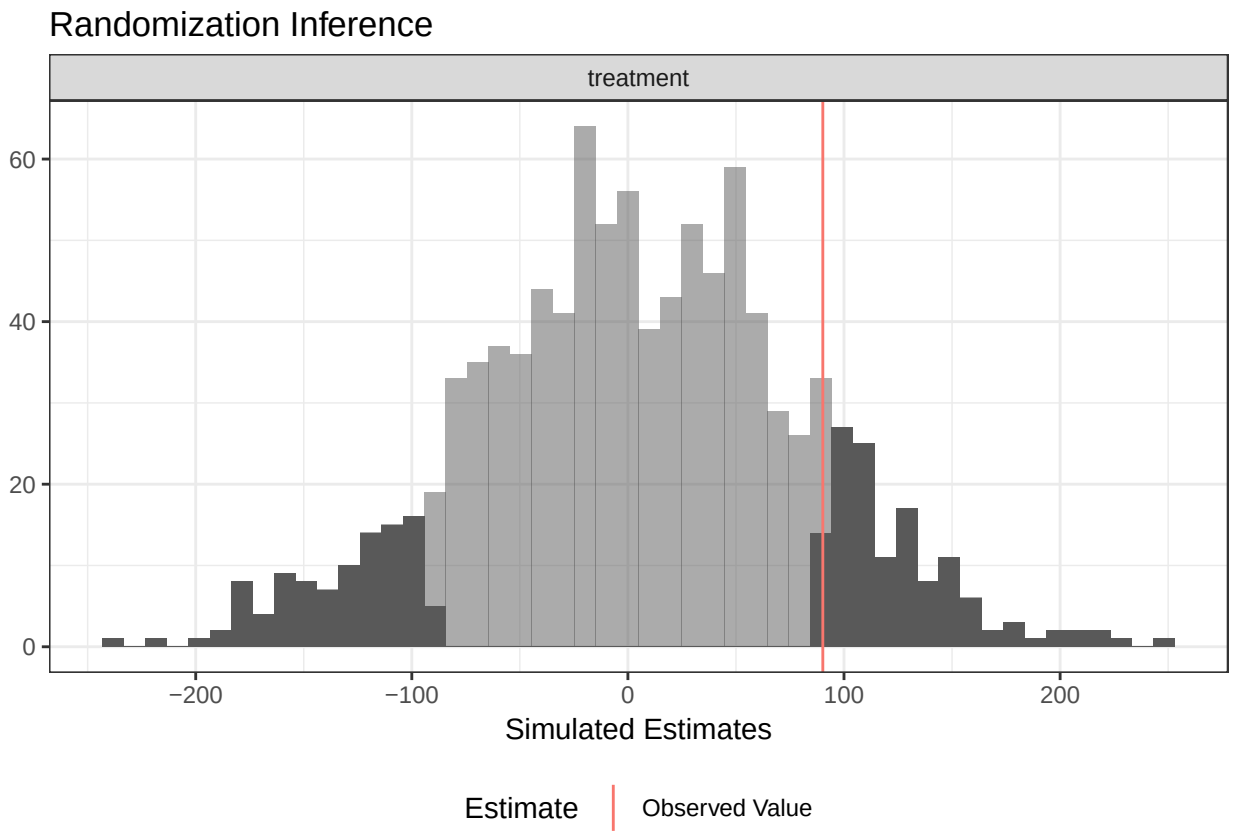
In ri2

```
library(ri2)

# Declare randomization procedure
declaration <- declare_ra(N = 65, m = 48)

# Conduct Randomization Inference
ri2_out <- conduct_ri(formula = relevant_measures_net ~ treatment,
                      declaration = declaration,
                      assignment = "treatment",
                      sharp_hypothesis = 0,
```

```
                      data = data13.1)

summary(ri2_out)
```

```
##   coefficient estimate two_tailed_p_value null_ci_lower null_ci_upper
## 1   treatment 90.20098              0.234     -155.4986      148.1394
```

```
plot(ri2_out)
```




```
# Part D
```

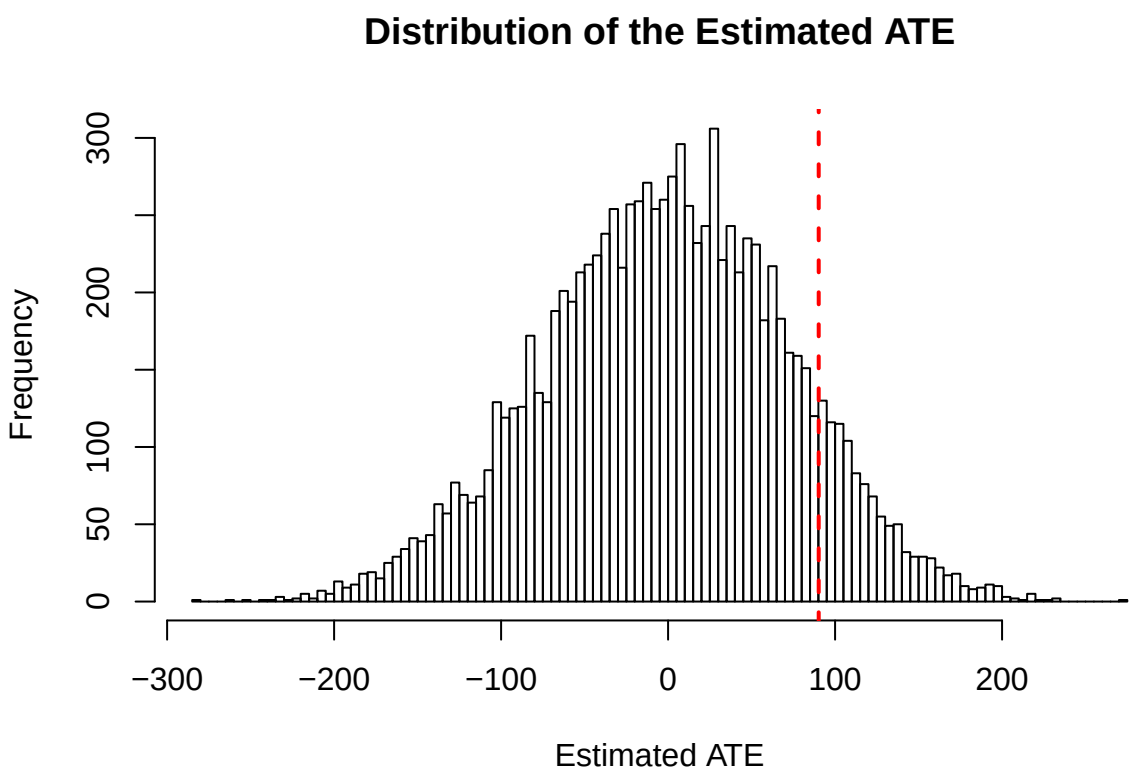
In ri

```
library(ri)

# Part C

# all possible permutations
perms <- genperms(data13.1$treatment)
```

```
## Too many permutations to use exact method.
## Defaulting to approximate method.
## Increase maxiter to at least 1867897112363099 to perform exact estimation.
```

```
# probability of treatment
probs <- genprobexact(data13.1$treatment)
# estimate the ATE
```

```
ate <- estate(data13.1$relevant_measures_net, data13.1$treatment, prob = probs)

## Conduct Sharp Null Hypothesis Test of Zero Effect for Each Unit

# generate potential outcomes under sharp null of no effect
Ys <- genouts(data13.1$relevant_measures_net, data13.1$treatment, ate = 0)
# generate sampling dist. under sharp null
distout <- gendist(Ys, perms, prob = probs)
# display characteristics of sampling dist. for inference
ri_out <- dispdist(distout, ate)
```




```
ri_out
```

```
## $two.tailed.p.value
## [1] 0.2166
##
## $two.tailed.p.value.abs
## [1] 0.2263
##
## $greater.p.value
## [1] 0.1083
##
## $lesser.p.value
## [1] 0.892
##
## $quantile
##      2.5%     97.5%
## -148.6939  139.1103
##
## $sd
```

```
## [1] 73.68862
##
## $exp.val
## [1] -1.269967
```

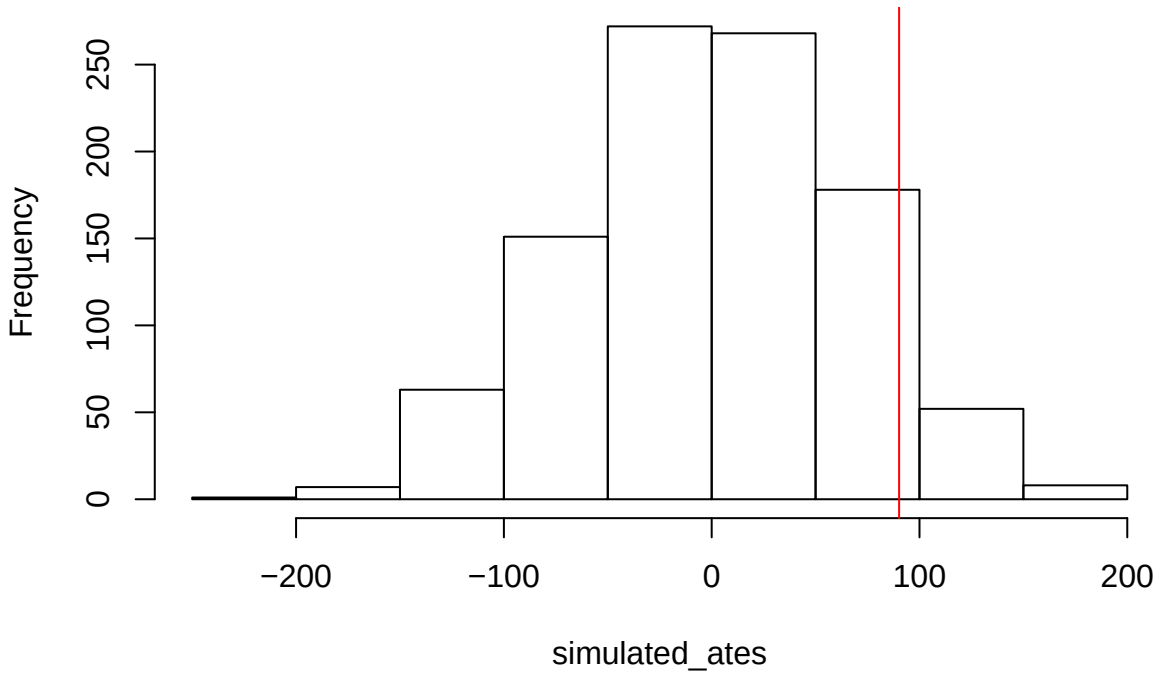
```
# Part D
```

By hand

```
# Part C

library(randomizr)
N <- 65
observed_ate <- with(data13.1, mean(relevant_measures_net[treatment == 1]) - mean(relevant_measures_net

simulated_ates <- rep(NA, sims)

for (i in 1:sims){
  data13.1$Z_sim <- complete_ra(N)
  simulated_ates[i] <- with(data13.1, mean(relevant_measures_net[Z_sim == 1]) - mean(relevant_measures_
}

p_two_tailed <- mean(abs(simulated_ates) >= abs(observed_ate))
p_upper <- mean(simulated_ates >= observed_ate)
p_lower <- mean(simulated_ates <= observed_ate)

c(observed_ate, p_two_tailed, p_upper, p_lower)
```

```
## [1] 90.20098  0.17100  0.08200  0.91800
```

```
hist(simulated_ates, breaks = 10)
abline(v = observed_ate, col = "red")
```




```r
# Part D

rand_mat <- matrix(nrow = N, ncol = sims, data = NA)

for (i in 1:sims) {
  data13.1$Z_sim <- complete_ra(N)
  diff = with(data13.1, mean(dem_perf_06[Z_sim == 1]) - mean(dem_perf_06[Z_sim == 0]))

  # if balance is "good", record rand.
  if (diff < 0.5) {
    rand_mat[, i] <- data13.1$Z_sim
  }

}

prob_assignment <- rowMeans(rand_mat, na.rm = TRUE)

data13.1$w <- with(data13.1, 1/(treatment*prob_assignment + (1-treatment)*(1-prob_assignment)))

fit <- lm(relevant_measures_net ~ treatment, weights = w, data = data13.1)
observed_ate2 <- coef(fit)[2]

simulated_ates2 <- rep(NA, sims)

for (i in 1:sims){
  if(any(is.na(rand_mat[,i]))){}
  else{
    data13.1$Z_sim <- rand_mat[,i]
    fit_sim <- lm(relevant_measures_net ~ Z_sim, data13.1)
```

```
    simulated_ates[i] <-coef(fit_sim)[2]
  }
}

p_two_tailed <- mean(abs(simulated_ates) >= abs(observed_ate))
p_upper <- mean(simulated_ates >= observed_ate)
p_lower <- mean(simulated_ates <= observed_ate)

hist(simulated_ates, breaks = 10)
abline(v = observed_ate2, col = "red")
```