

Problem Set 2 – Solutions

Research Design for Causal Inference

Due: April 14, 2013

PART I – GERBER & GREEN CHAPTER 3 EXERCISES

Excercise 1:

- (a) A standard error is the standard deviation of a sampling distribution. See Box 3.2 in FEDAI for formulas for both.
- (b) Randomization inference starts from the “sharp null hypothesis” of no effects for any subject. Under this null, you assume that treatment assignment did not matter and therefore construct a “sampling” or “randomization” distribution by using the original randomization procedure to simulate permutations of treatment and control assignments. The statistical significance of the observed outcome is then calculated by seeing how many of these simulated potential outcomes produce ATE estimates at least as large as the observed ATE.
- (c) A 95% confidence interval uses the observed data to describe the range of outcomes that have a 95% probability of encompassing the true outcome.
- (d) Complete random assignment pools all experimental subjects together and randomly assigns them to experimental conditions without regard for any attributes or prior groupings. Block random assignment groups subjects who share specific characteristics and assigns conditions *within* those blocks. Cluster random assignment takes subjects who enter the study in existing groups (e.g., classrooms, teams, etc.) and randomly assigns the intact groups to conditions.
- (e) Balanced designs ensure precise and narrow confidence interval calculations as well as unbiased difference of means estimates in cluster-randomized studies. (Note: There may be other advantages – did you notice any?)

Exercise 5:

```

#
# Create table 2.1 again (note you had code to do this from pset 1)
y.0 <- c(10, 15, 20, 20, 10, 15, 15)
y.1 <- c(15, 15, 30, 15, 20, 15, 30)

# One way to do this is with a ``for'' loop.
# First, I'll create an empty vector with seven items called
# ``diff.means'' to store the results:

diff.means <- rep(NA, 7)

# Now, I'll use the loop to cycle through the treatment and control
# values. For each iteration (which I call ``i'' here), I put one unit
# into treatment and remove it from control. Then, I calculate the
# average control outcome and subtract it from treatment, storing the
# result in my ``diff.means'' vector.

for(i in 1:length(diff.means)){
  treat <- y.1[i]
  control <- y.0[-i]
  diff.means[i] <- mean(treat) - mean(control)
  rm(treat); rm(control)
}

# Here are the difference-in-means for all seven randomizations
diff.means

## [1] -0.8333333  0.0000000 15.8333333  0.8333333  4.1666667  0.0000000
## [7] 15.0000000

```

Exercise 6: *(Bonus: verify/implement G & G's calculation of the 95% CI for this estimate using statistical software.)*

```

# Read in the dataset:
d <- read.csv(
  "http://aaronshaw.org/teaching/2015/causal/data/clingsmith_et_al.csv")

# The treatment variable is "success" indicating whether they won the
# lottery ("1") or not ("0").

```

```

table(d$success)

##
##    0    1
## 448 510

# The outcome variable is "views" which is a summary score ranging
# from -12 -- 12:
summary(d$views)

##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -12.000   0.000   2.000   2.121   3.000  12.000

# Make your life easier. Drop all the rest of the variables:
d <- d[,c("success", "views")]

d$success <- as.logical(d$success) # Logical variables.
# Aside: Logical variables are bad for computational performance!

### Now, here we go with the actual calculations:
# Step 0: use set.seed() to make your analysis reproducible:
set.seed(20150419)

# Step 1: Create a treatment assignment function that uses blocking:
treat.assign <- function(treat){
  treat.vector <- sample(treat)
  return(treat.vector)
}

# Step 2: Generate a randomization distribution by replicating
# treatment assignment 10,000 times:
rand.dist <- replicate(10000, treat.assign(d$success))

# Always a good idea to check the class of a new object:
class(rand.dist)

## [1] "matrix"

# Let's make that a data frame:
rand.dist <- data.frame(rand.dist)

```

```

# Just in case, drop any duplicate columns using the unique() function:
rand.dist <- unique(rand.dist, MARGIN = 2)

# Step 3: Create a function to calculate the difference in means for a
# single treatment assignment vector:
gen.difference.means <- function(treat.vector){
  y.treat <- d$views[treat.vector]
  y.control <- d$views[!treat.vector]
  ate <- mean(y.treat) - mean(y.control)
  return(ate)
}

# Step 4: Use the function to calculate the difference of means for
# all of the treatment assignment vectors:
ate.dist <- unlist(lapply(rand.dist, gen.difference.means))

# Step 5: Calculate the number and proportion of potential outcomes
# larger than the observed; one & two-tailed p-values, etc.

# Start by calculating the observed ATE
ate.obs <- mean(d$views[d$success]) - mean(d$views[!d$success])

### One-tailed test:
# Calculate the number of potential outcomes larger than the observed
# ATE:
table(ate.dist > ate.obs)

##
## FALSE TRUE
## 9986 14

# and the corresponding proportion (for "TRUE") gives you the p-value for
# the probability that an effect as large would be observed under the
# sharp null:
prop.table(table(ate.dist > ate.obs))

##
## FALSE TRUE
## 0.9986 0.0014

### Two-tailed test

```

```
# Use absolute values:
table(abs(ate.dist) > ate.obs)

##
## FALSE TRUE
## 9975    25

prop.table(table(abs(ate.dist) > ate.obs))

##
## FALSE TRUE
## 0.9975 0.0025
```

Excercise 12:

(a) # I'll do this one with R:

```
d <- data.frame(
  "school" = c("a", "b", "b", "c", "c", "c"),
  "classroom" = c("1", "1", "2", "1", "2", "3"),
  "y0" = rep(0,6),
  "y1" = c(0,1,1,2,2,2)
)

# Part a - the wording here is a little ambiguous, but I assume Gerber
# \& Green are looking for the true ATE
mean(d$y1-d$y0)

## [1] 1.333333
```

(b) Section 3.6.2 suggests that unequal cluster sizes can produce biased difference-in-means estimates. You could talk about this and/or calculate it directly. I'll opt for the latter approach:

```
# Since the average control outcome is zero no matter the school
# assigned to treatment, we can just work with the average control
# outcomes under the different potential random assignments:

y.a <- mean(d$y1[d$school == "a"])
```

```

y.b <- mean(d$y1[d$school == "b"])
y.c <- mean(d$y1[d$school == "c"])

mean(c(y.a, y.b, y.c))

## [1] 1

```

The outcome is biased (true ATE was larger). It is an example of a situation in which the potential outcomes covary with cluster size (the bigger the cluster, the bigger the estimated treatment effect!). The implication is that this is a worrisome situation we might avoid through balanced cluster size or the use of alternative estimators.

```

(c) # I'll check this for bias by calculating the mean of the
# potential outcomes:

mean(
  c(
    mean(d$y1[d$school == "a" | d$school == "b"]) - 0,
    mean(d$y1[d$school == 'c']) - 0)
)

## [1] 1.333333

# The design yields unbiased estimates (their mean is equal to the
# true ATE). The implications of this is that unequal cluster sizes can
# introduce bias in difference of means estimates. Gerber & Green
# recommend calculating difference in total outcomes as a result!

```

PART II – THE LADY TASTING TEA

In a very famous example that effectively invented Randomization Inference, Sir R.A. Fisher's (1935) provided the following description of a Lady Tasting Tea:

A lady declares that by tasting a cup of tea made with milk she can discriminate whether the milk or the tea infusion was first added to the cup.

To test The Lady's claim, he proposes to make six cups of tea – three of them “milk first” and three of them “tea first” – and to present the six cups to The Lady in random order. The Lady knows that this will be the study design, so she knows she will receive three cups of each type and that the order of the presentation will be random.

Part a Suppose that The Lady correctly identifies whether milk or tea was added first for all six out of six cups presented to her in the experiment. We call the milk-first cups the treatment ($A_i = 1$) and The Lady's guess $Y_i = 1$ when she guesses milk-first. Use the number of correctly identified cups as the test statistic, formally represented by the following equation:

$$\sum_i A_i Y_i + (1 - A_i)(1 - Y_i) \quad (1)$$

What is the p-value for a test under the “sharp null hypothesis” that The Lady has no ability to discriminate?

Hint: Use your knowledge of the research design to calculate the total number of potential outcomes for the experiment and then to calculate the likelihood that The Lady would produce 6 out of 6 correct answers if she had been guessing randomly).

```
# First, calculate the total number of potential
# outcomes (which Rosenbaum and others sometimes call ``Omega'')

choose(6,3)

## [1] 20

# Now, enter the ``World of the Sharp Null Hypothesis'' -- Because there is
# exactly one of these 20 possible combinations that corresponds to a
# perfect set of (6 out of 6) responses, the probability of getting 6
# out of 6 correct answers had The Lady been guessing randomly is
# exactly:

1/20

## [1] 0.05

# There 's your p-value.
```

Part b Imagine that The Lady makes one mistake and therefore correctly identifies whether milk or tea was added first for four out of the six cups (revisit the study design if you're confused as to why one mistake yields two wrong answers). What is the p-value for a test under the sharp null hypothesis that she has no ability to discriminate under these conditions?

```

# Again, consider the ``World of the Sharp Null Hypothesis'' in which
# The Lady has no ability to discriminate between Tea or Milk-first. Our
# job is to calculate the likelihood that she got 4 (or more!) correct
# answers by guessing randomly. So let's enumerate the potential outcomes.

# In addition to the 1 combination in which she guessed all 6
# correctly, there are now several other conditions under which she
# might have gotten 4 out of 6 correct answers. She might have drawn 1
# incorrect tea-first cup (out of 3) and 1 incorrect milk-first cup (out
# of 3), meaning that there were ``3 choose 1'' options for each:

choose(3,1)

## [1] 3

# multiplying two of these together (to account for the tea-first and
# the milk-first cups) gives you the total number of combinations in
# which she might have gotten exactly 4 correct answers:

choose(3,1) * choose(3,1)

## [1] 9

# Adding the 9 combinations of 4 correct answers plus the one
# combination of 6 correct answers gives us 10 combinations (once again
# out of a possible 20 total potential outcomes), resulting in a
# p-statistic of:

10/20

## [1] 0.5

```

Part c Now, imagine that instead of using “fixed margins” for the randomization (i.e. requiring three milk-first cups as well as three tea-first cups) you were to conduct Fisher’s experiment using “binomial randomization” (i.e. randomly determining whether to add milk or tea first *for each cup*). For each cup, we’ll say that there is a probability $p = 1/2$ that it is milk-first and a $1 - p$ probability that it is tea-first. In this case, The Lady does not know the value of p , but she does know that the cups have been assigned under binomial randomization. If, under these conditions, The Lady makes one mistake, what is the p-

value for a test under the sharp null hypothesis of no effects?

Hint: To do this, you will want to consider that under binomial randomization in an experiment with N trials, the probability, Pr , that the number of “successes”, Y (in this case, correctly identified cups of tea), is equal to some value y is:

$$Pr(Y = y) = \binom{N}{y} p^y (1 - p)^{N-y} \quad (2)$$

Read $\binom{N}{y}$ as “N choose y.” Note that if you want to calculate $\binom{N}{y}$ you can probably do so easily by hand for small numbers, but as the numbers grow you may want to use the `choose()` function in R, a scientific calculator, or Wolfram Alpha to handle the calculations.

```
# Using the formula provided, the p-value under binomial randomization
# is:

choose(6,5) * (.5^5) * ((1-.5)^1)

## [1] 0.09375

# Quite a difference!
```