Randomization Inference

An Introduction

Dr. Travis G. Coan

T.Coan@exeter.ac.uk

Assume that you have data from an experiment with a binary treatment and continuous outcome variable.

How would you analyze the "treatment effect" using these data?

What if I told you that the data come from a convenience sample?

What if I told you that the data come from a convenience sample and there are only $n = 8$ observations?

A "reasoned basis for inference"

The logic of randomization inference

Assume that we conducted a uniform randomized experiment with $n = 6$ subjects where half the subjects receive treatment. Prior to assigning the treatment, we thus have random vector of *potential* treatment assignments:

$$\mathbf{T} = \begin{bmatrix} T_1 \\ \vdots \\ T_n \end{bmatrix} \tag{1}$$

And suppose that the *observed* treatment assignment is as follows:

$$\mathbf{T}^{\mathsf{T}} = [\, 0, 0, 1, 0, 1, 1 \,]$$

Where $T = 1$ implies receiving treatment and $T = 0$ implies assignment to the control condition.

The logic of randomization inference

Assume further that we collect data on a single outcome variable, y, and can define a **test statistic**, S, as follows:

$$S = f(y, \mathbf{T}) \tag{2}$$

That is, a statistic which is a function of the outcome and treatment assignment. For instance, the difference in means across the treatment and control conditions fits this criteria:

$$\Delta = \bar{y}_{T=1} - \bar{y}_{T=0}$$

However, this could be *any* function of outcome and treatment assignment; this offers considerable flexibility when choosing a statistic.

Create example data in **R**

It is helpful to have a concrete empirical example. In **R**, you could run the following:

```
# Create sample data
set.seed(12345)
T <- c(0,0,1,0,1,1)
u <- rnorm(6, mean=0, sd=15)
y <- 50 + 10*T + u

# Define a function for the test statistic
S <- function(y, T){
  X <- cbind(y,T)
  return(mean(subset(X, T == 1, select = y)) -
           mean(subset(X, T == 0, select = y)))
}
```

The logic of randomization inference

So our data would look like:

Subject	y	T
$i = 1$	58.8	0
$i = 2$	60.6	0
$i = 3$	58.4	1
$i = 4$	43.2	0
$i = 5$	69.1	1
$i = 6$	32.7	1

And thus our **test statistic** would be:

$$S = \Delta = 54.3 - 54.2 = -0.81$$

How do we define the causal effect of treatment?

The potential outcomes framework

Using the so-called "Rubin Causal Model," the causal effect of treatment is defined in terms of the **potential outcome** under treatment and control for each subject:

$$Y_i(1) - Y_i(0)$$

This quantity is unobservable leads to Holland's **fundamental problem of causal inference**: at most, we can only observe one of these potential outcomes.

Causal inference and missing data

Subject	y	T	$Y(0)$	$Y(1)$
$i = 1$	58.8	0	58.8	?
$i = 2$	60.6	0	60.6	?
$i = 3$	58.4	1	?	58.4
$i = 4$	43.2	0	43.2	?
$i = 5$	69.1	1	?	69.1
$i = 6$	32.7	1	?	32.7

The **sharp** null hypothesis

In the randomization inference framework, one focuses directly on the **sharp null hypothesis**:

$$H_0\text{: } Y_i(1) - Y_i(0) = 0 \text{ for all } i$$

This is the null hypothesis that experimentalist are typically interested in: "no causal effect" of the treatment.

Notice what "no effect" implies: subject i's potential outcomes under treatment and control are constant. Put differently, the observed value of y is assumed constant regardless of the treatment a subject receives. So one can "flip" from treatment to control without impacting y.

The **sharp** null hypothesis

For this *particular* randomization, we can thus impute the missing potential outcome under the **sharp null hypothesis**:

Subject	y	T	$Y(0)$	$Y(1)$
$i = 1$	58.8	0	58.8	58.8
$i = 2$	60.6	0	60.6	60.6
$i = 3$	58.4	1	58.4	58.4
$i = 4$	43.2	0	43.2	43.2
$i = 5$	69.1	1	69.1	69.1
$i = 6$	32.7	1	32.7	32.7

Defining the **exact** null distribution

While all potential outcomes are constant under the sharp null, our test statistic will still vary based on who receives the treatment. Note that we can capture *all* of the possible ways that our test statistic could vary under the sharp null hypothesis by doing the following:

1. Permuting all possible random assignment vectors using the typical formula for a combination:

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} \tag{3}$$

2. Calculate the test statistic, S, using each potential assignment vector and the (fixed) vector of outcomes, y.

Permute all possible outcomes

$$\Omega = \begin{pmatrix} \begin{bmatrix}1\\1\\1\\0\\0\\0\end{bmatrix} & \begin{bmatrix}1\\1\\0\\1\\0\\0\end{bmatrix} & \cdots\cdots & \begin{bmatrix}1\\0\\0\\1\\0\\1\end{bmatrix} & \begin{bmatrix}1\\0\\0\\0\\1\\1\end{bmatrix} \\ \begin{bmatrix}0\\1\\1\\1\\0\\0\end{bmatrix} & \begin{bmatrix}0\\1\\1\\1\\0\\0\end{bmatrix} & \cdots\cdots & \begin{bmatrix}0\\0\\1\\0\\1\\1\end{bmatrix} & \begin{bmatrix}0\\0\\0\\1\\1\\1\end{bmatrix} \end{pmatrix} \tag{4}$$

Note: $\binom{6}{3} = 20$ potential random treatment vectors.

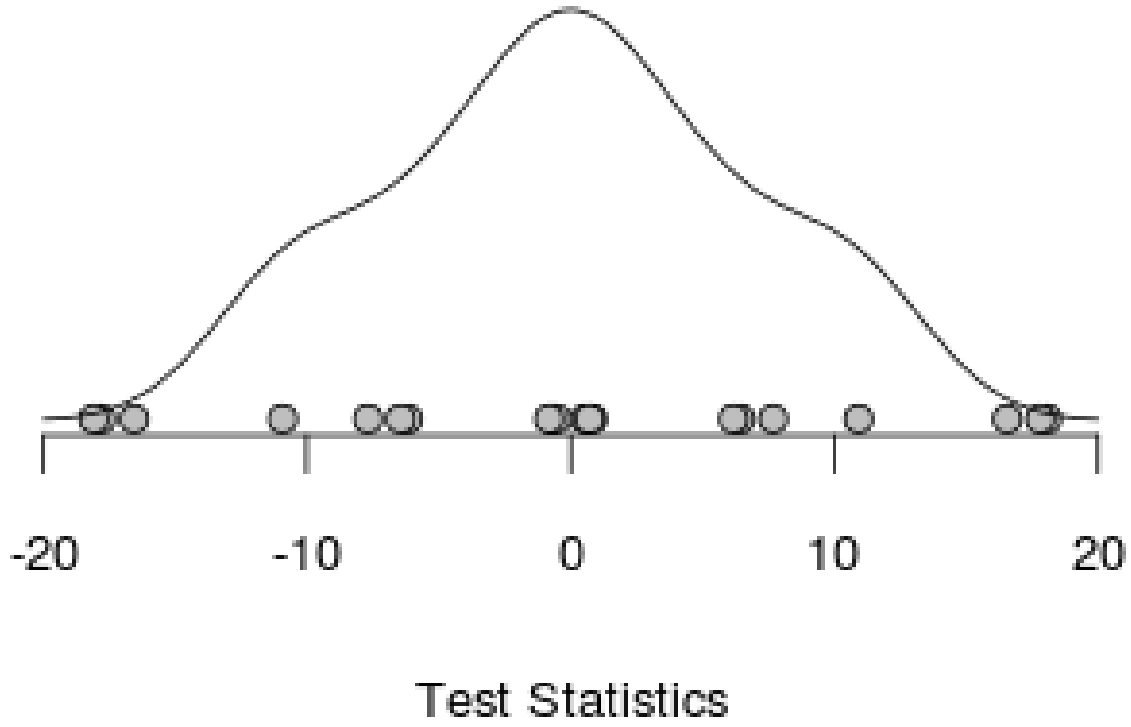
Calculate S for each vector (a little **R** code)

```
# Permute assignment indices
n <- 6
k <- 3
assign <- combn(seq(n), k)

# Define function to compute S under the null
compute.null <- function(y, assign.vec){
  n <- length(y)
  omega <- rep(0, times=n)
  omega[assign.vec] <- 1
  return(S(y, omega))
}

# Compute the null distribution
null.dist <- apply(assign, 2, compute.null, y=y)
```

The null distribution



Note: Randomization distribution of S. Kernel density as overlay.

The **exact** p-value

We can calculate the **exact p-value** directly from the null distribution. For a **one-sided** test (assuming a positive effect):

$$p = Pr(S \geq s_i | H_0) = \frac{\sum I((S \geq s_i))}{|\Omega|} \tag{5}$$

Where I represents the indicator function and $|...|$ indicates the cardinality of the set (i.e., the number of elements). We can easily adapt (5) to accommodate a one-sided test in the opposite direction or a two-sided.

Using **R** for exact p-value

```
# Calculate test statistic
test.stat <- S(y, T)

# Define function to calculate p-value (two-sided)
p.value <- function(null.dist, test.stat){
  left.side <- (sum(null.dist < test.stat)/
                  length(null.dist))
  right.side <- (sum(null.dist > test.stat)/
                   length(null.dist))
  two.side <- (sum(abs(null.dist) > abs(test.stat))/
                 length(null.dist))
  return(list(left=left.side, two.side=two.side,
              right.side=right.side))
}
```

Note: The **two-sided** p-value for our example is 0.70.

What about large n studies?

Calculating the full exact distribution becomes computationally infeasible in large n studies. How do we get around this problem? We can use a Monte Carlo based procedure with a suitably high number of resamples (typically, 1,000 to 5,000 resamples).

Example: Suppose we now have a sample of $n = 1000$ with 400 individuals assigned to treatment and 600 assigned to control. We could estimate the null distribution via the following sampling algorithm:

1. Hold the outcome variable, y, fixed at its observed value.
2. Randomly shuffle the treatment vector, T.
3. Estimate the test statistic, save, and repeat.

What about the size and uncertainty of effects?

Researchers are often interested in a point estimate for the treatment effect and a confidence interval for that point estimate. We can do this in the randomization inference framework by **inverting the test**.

We do need to add two additional assumptions: 1) we now need SUTVA and 2) we need to specify a **model of effects**. Typically, one assumes additive, constant effects:

$$Y_i(1) = Y_i(0) + \tau \tag{6}$$

Basically, we would conduct a series of hypothesis tests with non-zero values of τ and keep the values not rejected at the value of α we have chosen.

One other cool aspect of randomization inference

Parametric tests such may be fully nested in the randomization inference framework, provided that one can identify a suitable test statistic.

For instance, Cohen and Coan (2014) nest a parametric double-bounded dichotomous choice model of contingent valuation.