Section 2

Randomization Inference

Sooahn Shin

GOV 2003

Sept 16, 2021

Overview

- Logistics:
  - **Pset 2 released!** Due at 11:59 pm (ET) on Sept 22

- Today's topics:
  1. Randomization inference (Design-based inference)
  2. Toy example: Donations encouragement (small/large sample)
  3. Inverting test to obtain CIs

Randomization inference

- Randomization inference (Design-based inference; permutation test)
  - Assignment mechanism: $\rightsquigarrow \Omega_0 = \{\boldsymbol{d} : \mathbb{P}(\boldsymbol{D} = \boldsymbol{d}) > 0\}$.
    - Bernoulli randomization $\rightsquigarrow$ use `rbinom(N, 1,.5)`
    - Completely randomized experiment $\rightsquigarrow$ use `ri::genperms()` or `sample()`
  - Sharp null hypothesis: $H_0 : \tau_i = Y_i(1) - Y_i(0) = const. \forall i$
    - $\rightsquigarrow$ We can fill out the missing potential outcomes
  - $\rightsquigarrow$ We can compute/approximate the distribution of test statistics $T(\boldsymbol{D}, \boldsymbol{Y})$ under the null (**randomization distribution**)

- Model-based inference
  - Assumes a distribution for potential outcomes

Toy example: Donations encouragement

- Setup:
  - N people
  - Encouragement by mail (0/1; D_i) → Donations to Harvard ($; Y_i)
  - Let Ω = set of 2^N treatment vectors (any N-vector of 0s and 1s).
- Suppose **complete randomization** has been implemented
  - $N = 6$ and $n_1 = \sum_i^N D_i = 3$.
  - $\rightsquigarrow \Omega_0 = \{\mathbf{d} \in \Omega : \sum_{i=1}^6 d_i = 3\} = \{(1,1,1,0,0,0),(1,1,0,1,0,0),\ldots\}$
- Test a sharp null of no effect: $H_0 : \tau_i = Y_i(1) - Y_i(0) = 0 \quad \forall i$.

Unit	Mailer D_i	Contr. Y_i	$Y_i(0)$	$Y_i(1)$
Jon	1	3	( )	3
Sansa	1	5	( )	5
Arya	1	0	( )	0
Robb	0	4	4	( )
Bran	0	0	0	( )
Rickon	0	1	1	( )

Randomization inference step-by-step

In a small sample,

1. Choose a sharp null hypothesis and a test statistic:
   - E.g.: $H_0 : \tau_i = 0$ for all i v. $H_1 : \tau_i \neq 0$ for some i
   - E.g.: absolute difference-in-means estimator
     $T_{\text{diff}} = \left| \frac{1}{n_1} \sum_{i=1}^{N} D_i Y_i - \frac{1}{n_0} \sum_{i=1}^{N} (1 - D_i) Y_i \right|$
2. Calculate observed test statistic: $T^{\text{obs}} = T(\mathbf{D}, \mathbf{Y})$.
3. List all the possible treatment vectors in Ω_0: $\{\widetilde{\mathbf{D}}_1, \ldots, \widetilde{\mathbf{D}}_K\}$ where $K = |\Omega_0|$
4. Calculate $\widetilde{T}_k = T(\widetilde{\mathbf{D}}_k, \mathbf{Y})$ for each k under the sharp null.
5. Observe the distribution of $\widetilde{T} = \{\widetilde{T}_1, \ldots, \widetilde{T}_K\}$.
6. Calculate the exact p-value: $p = \frac{1}{K} \sum_{k=1}^{K} \mathbb{I}(\widetilde{T}_k \geq T)$

1. Calculate observed test statistic

Unit	Mailer D_i	Contr. Y_i	$Y_i(0)$	$Y_i(1)$
Jon	1	3	( )	3
Sansa	1	5	( )	5
Arya	1	0	( )	0
Robb	0	4	4	( )
Bran	0	0	0	( )
Rickon	0	1	1	( )

$$T^{obs}_{diff} = |8/3 - 5/3| = 1$$

```r
y <- c(3, 5, 0, 4, 0, 1)
D <- c(1, 1, 1, 0, 0, 0)
T_obs <- abs(mean(y[D == 1]) - mean(y[D == 0]))
T_obs
```

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

2. Randomization distribution

- Possible treatment assignments $\{\widetilde{\mathbf{D}}_1, \ldots, \widetilde{\mathbf{D}}_{20}\}$
- Test statistics under the null $\widetilde{T} = \{\widetilde{T}_1(\widetilde{\mathbf{D}}_1, \mathbf{Y}), \ldots, \widetilde{T}_{20}(\widetilde{\mathbf{D}}_{20}, \mathbf{Y})\}$

Unit	Mailer $\widetilde{D}_1$	Contr. Y_i	$Y_i(0)$	$Y_i(1)$
Jon	**1**	3	(3)	3
Sansa	**1**	5	(5)	5
Arya	**0**	0	(0)	0
Robb	**1**	4	4	(4)
Bran	**0**	0	0	(0)
Rickon	**0**	1	1	(1)

$$\widetilde{T}_1 = |12/3 - 1/3| = 3.67$$

2. Randomization distribution

- Possible treatment assignments $\{\widetilde{\mathbf{D}}_1, \ldots, \widetilde{\mathbf{D}}_{20}\}$
- Test statistics under the null $\widetilde{T} = \{3.67, \ldots, \widetilde{T}_{20}(\widetilde{\mathbf{D}}_{20}, \mathbf{Y})\}$

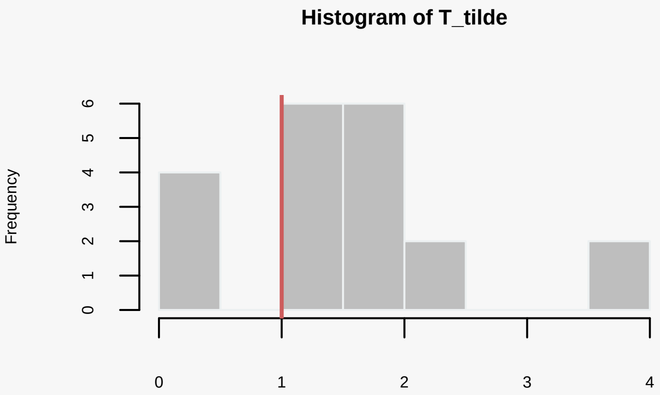
Unit	Mailer $\widetilde{D}_{20}$	Contr. Y_i	$Y_i(0)$	$Y_i(1)$
Jon	**0**	3	(3)	3
Sansa	**0**	5	(5)	5
Arya	**0**	0	(0)	0
Robb	**1**	4	4	(4)
Bran	**1**	0	0	(0)
Rickon	**1**	1	1	(1)

$$\widetilde{T}_{20} = |5/3 - 8/3| = 1$$

2. Randomization distribution

```r
D_bold <- ri::genperms(D)
K <- ncol(D_bold)
T_tilde <- rep(NA, times = K)
for (i in 1:K) {
  D_tilde <- D_bold[, i]
  T_tilde[i] <- abs(mean(y[D_tilde == 1]) - mean(y[D_tilde == 0]))
}
```

3. P-value




```r
# p-value
mean(T_tilde >= T_obs)
```

```
## [1] 0.8
```

Randomization inference step-by-step

In a large sample,

1. Choose a sharp null hypothesis and a test statistic:
2. Calculate observed test statistic: $T^{\text{obs}} = T(\mathbf{D}, \mathbf{Y})$.
3. **Too many possible treatment vectors in Ω_0 → take K samples!**
4. Calculate $\widetilde{T}_k = T(\widetilde{\mathbf{D}}_k, \mathbf{Y})$ for each k under the sharp null.
5. Observe the distribution of $\widetilde{T} = \{\widetilde{T}_1, \ldots, \widetilde{T}_K\}$.
6. Calculate the p-value: $p = \frac{1}{K}\sum_{k=1}^{K} \mathbb{I}(\widetilde{T}_k \geq T)$

Choosing test statistics

- Difference in means
- Rank statistic
  - when we have many outliers
  - $\rightsquigarrow$ `wilcox.test()` for rank-sum statstic
- $S = \sum_{i=1}^{N} D_i Y_i(1)$
  - when Y_i is binary, Fisher's exact test (recall Lady Tasting Tea)
  - $\rightsquigarrow$ `fisher.test()`
- Using absolute values under the sharp null of no effect
  - $\rightsquigarrow$ testing against a two-sided alternative hypothesis

$$H_0 : \tau_i = 0 \; \forall i \qquad H_1 : \tau_i \neq 0 \text{ for some } i$$

Confidence Intervals: Inverting the Test

- For a sharp null $\tau_i = Y_i(1) - Y_i(0) = \tau_0 \quad \forall i$, we can conduct the test and calculate the p-value.
- Repeat the above with different values of τ_0
- 95% CI: The range of τ_0 that we cannot reject the null at the 0.05 level.

Example Code for Inverting the Test

```
# Data
Yi <- large_sample$factor # Observed outcome
Di <- recode(large_sample$canvass, `Placebo` = 0, `Full Intervention` = 1)
N <- length(Yi);  n1 <- sum(Di); n0 <- sum(1-Di)
# Pick candiate taus on a grid
tau_cand <- seq(-0.5, 0.5, by = 0.01)
save_pval <- rep(NA, length(tau_cand)) # to save the p-value below

# 1. Calculate the observed statistics
T_obs <- sum(Di*Yi)/n1 - sum((1-Di)*Yi)/n0
```

Example Code for Inverting the Test

```
# TODO 2: Create function for computing p-value given tau and observed stat
your_fun <- function(tau, t_obs, n_sim = 1000) { # Input: tau, observed sta
  # TODO 2-1: Calculate Yi(1) using Yi, Di, and tau
  Y1 <- NULL
  # TODO 2-2: Calculate Yi(0) using Yi, Ti, and tau
  Y0 <- NULL
  Ttilde_ls <- rep(NA, n_sim)
  # Simulation:
  for (s in 1:n_sim) {
    # TODO 2-3: Randomly sample treatment vectors
    Dtilde_s <- NULL
    # TODO 2-4: For each treatment vector, compute the statistics
    Ttilde_ls[s] <- NULL
  }
  # 2-5: Calculate and return the p-value
  pval <- 2 * min(mean(Ttilde_ls >= t_obs), mean(Ttilde_ls <= t_obs))
  return(pval)
}
```

Example Code for Inverting the Test

```r
# TODO 3: Loop over each candidate tau
set.seed(123)
for (t in 1:length(tau_cand)) {
  save_pval[t] <- your_fun(tau_t, T_obs)
}

# 4. Obtain the upper / lower bound of 95% CI
lb <- tau_cand[min(which(save_pval >= 0.025))]
ub <- tau_cand[max(which(save_pval >= 0.025))]

# TODO 5: Print the 95% CI (lb, ub)
```

Visualization of p-values from test inversion