RCTs PSET 2

Matthew Zhao

2023-11-05

1 Randomization and Balance Checks

1.1

Brune et al. (2021) asks what is the impact of offering a deferred savings program to workers on saving/investment outcomes and what are the mechanisms which affect take-up and use.

1.2

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

# set seed
set.seed(2090820)

# import data
df_complete <- read.csv('data/brune_et_al_data.csv')

# complete randomization
n <- df_complete %>% nrow()

# n draws from uniform(0,1), essentially assigning a random number to each worker
# when combined with rank
df_complete$rand <- runif(n, 0, 1)
df_complete$rank <- rank(df_complete$rand)

# assigns top half of random rank to control, bottom to treatment
df_complete$treat <- ifelse(df_complete$rank <= (n / 2), 1, 0)

table(df_complete$treat)
```

```
##
##   0   1
## 435 435
```

1.3

```r
# table of means with each row as a variable and T/C as columns
complete_means <- df_complete %>%
  group_by(treat) %>%
  summarise(female_w = mean(female_w),
            d_married_w = mean(d_married_w),
            worker_age_w = mean(worker_age_w),
            plucker_w = mean(plucker_w),
            sav_all_tot_tot_w = mean(sav_all_tot_tot_w),
            .groups = 'keep') %>%
  pivot_longer(cols=!treat,names_to = 'name', values_to = 'value') %>%
  mutate(value = round(value,2)) %>%
  pivot_wider(names_from = treat, names_prefix = 'treat_',
              values_from = value)

# table of standard deviations
complete_sds <- df_complete %>%
  group_by(treat) %>%
  summarise(female_w = sd(female_w),
            d_married_w = sd(d_married_w),
            worker_age_w = sd(worker_age_w),
            plucker_w = sd(plucker_w),
            sav_all_tot_tot_w = sd(sav_all_tot_tot_w),
            .groups = 'keep') %>%
  pivot_longer(cols=!treat,names_to = 'name', values_to = 'value') %>%
  mutate(value = round(value,2)) %>%
  pivot_wider(names_from = treat, names_prefix = 'sd_treat_',
              values_from = value)

# table of t stats and p values for diff in means between T & C
complete_stats <- df_complete %>%
  select(treat, female_w, d_married_w,
         worker_age_w, plucker_w, sav_all_tot_tot_w) %>%
  pivot_longer(cols=!treat,
               names_to='name',values_to='value') %>%
  group_by(name) %>%
  summarise(tstat = t.test(value ~ treat)$statistic,
            pval = t.test(value ~ treat)$p.value,
            .groups='keep')

# creating balance table from above tables
complete_bal <- complete_means %>%
  left_join(complete_sds,
            by='name') %>%
  left_join(complete_stats,
            by='name')

# export as latex
complete_bal_tab <- xtable(complete_bal)
align(complete_bal_tab) <- xalign(complete_bal_tab)
display(complete_bal_tab) <- xdisplay(complete_bal_tab)
print(complete_bal_tab, include.rownames=FALSE)
```

name	treat_0	treat_1	sd_treat_0	sd_treat_1	tstat	pval
female_w	0.37	0.33	0.48	0.47	1.21	0.23
d_married_w	0.69	0.71	0.46	0.46	-0.52	0.61
worker_age_w	39.56	39.47	10.70	10.85	0.13	0.90
plucker_w	0.78	0.75	0.41	0.43	1.20	0.23
sav_all_tot_tot_w	31968.41	32160.47	52691.10	49432.78	-0.06	0.96

1.4

```
# stratified randomization

# clear environment
rm(list = ls())

# set seed
set.seed(2090820)

# import data
df_strat <- read.csv('data/brune_et_al_data.csv')

n <- df_strat %>% nrow()
df_strat$rand <- runif(n, 0, 1)

# within each division, assign worker to treatment if random draw from
# uniform[0,1] is greater than the bottom third of the draws of that division
# to produce a 2:1 T/C allocation
df_strat <- df_strat %>%
  group_by(division) %>%
  mutate(treat_strat =
           ifelse(rand <= quantile(rand,c(0.33)), 0, 1)) %>%
  ungroup()

df_strat %>%
  count(division, treat_strat) %>%
  pivot_wider(names_from = treat_strat,
              names_prefix = 'treat_',
              values_from = n)
```

```
## # A tibble: 11 x 3
##    division treat_0 treat_1
##       <int>   <int>   <int>
##  1        1      28      56
##  2        2      12      25
##  3        3      24      48
##  4        4      35      72
##  5        5      33      65
##  6        7      26      51
##  7        8      30      59
##  8        9      27      55
##  9       10      26      51
## 10       11      28      55
## 11       12      21      43
```

1.5

```
# table of means with each row as a variable and T/C as columns
strat_means <- df_strat %>%
  group_by(treat_strat) %>%
  summarise(female_w = mean(female_w),
            d_married_w = mean(d_married_w),
            worker_age_w = mean(worker_age_w),
            plucker_w = mean(plucker_w),
            sav_all_tot_tot_w = mean(sav_all_tot_tot_w),
            .groups = 'keep') %>%
  pivot_longer(cols=!treat_strat,names_to = 'name', values_to = 'value') %>%
  mutate(value = round(value,2)) %>%
  pivot_wider(names_from = treat_strat, names_prefix = 'treat_',
              values_from = value)

# table of standard deviations
strat_sds <- df_strat %>%
  group_by(treat_strat) %>%
  summarise(female_w = sd(female_w),
            d_married_w = sd(d_married_w),
            worker_age_w = sd(worker_age_w),
            plucker_w = sd(plucker_w),
            sav_all_tot_tot_w = sd(sav_all_tot_tot_w),
            .groups = 'keep') %>%
  pivot_longer(cols=!treat_strat, names_to = 'name', values_to = 'value') %>%
  mutate(value = round(value,2)) %>%
  pivot_wider(names_from = treat_strat, names_prefix = 'sd_treat_',
              values_from = value)

# table of t stats and p values for diff in means between T & C
strat_stats <- df_strat %>%
  select(treat_strat, female_w, d_married_w,
         worker_age_w, plucker_w, sav_all_tot_tot_w) %>%
  pivot_longer(cols=!treat_strat,
               names_to='name',values_to='value') %>%
  group_by(name) %>%
  summarise(tstat = t.test(value ~ treat_strat)$statistic,
            pval = t.test(value ~ treat_strat)$p.value,
            .groups='keep')

# creating balance table from above tables
strat_bal <- strat_means %>%
  left_join(strat_sds,
            by='name') %>%
  left_join(strat_stats,
            by='name')

# export as latex
strat_bal_tab <- xtable(strat_bal)
align(strat_bal_tab) <- xalign(strat_bal_tab)
display(strat_bal_tab) <- xdisplay(strat_bal_tab)
print(strat_bal_tab, include.rownames=FALSE)
```

name	treat_0	treat_1	sd_treat_0	sd_treat_1	tstat	pval
female_w	0.33	0.35	0.47	0.48	-0.66	0.51
d_married_w	0.70	0.70	0.46	0.46	0.16	0.88
worker_age_w	39.29	39.63	11.02	10.64	-0.43	0.67
plucker_w	0.74	0.78	0.44	0.42	-1.06	0.29
sav_all_tot_tot_w	33270.55	31461.38	48699.33	52228.98	0.50	0.61

Survey CTO

2.1

See pdf appended to the end of the doc.

2.2

```
rand_treat_ids <- df_strat %>%
  filter(treat_strat == 1) %>%
  sample_n(10) %>%
  select(worker_id, treat_strat)
rand_control_ids <- df_strat %>%
  filter(treat_strat == 0) %>%
  sample_n(10) %>%
  select(worker_id, treat_strat)
rand_ids <- bind_rows(rand_treat_ids,
          rand_control_ids)
rand_ids
```

```
## # A tibble: 20 x 2
##    worker_id treat_strat
##        <int>       <dbl>
##  1      1901           1
##  2      2515           1
##  3      1140           1
##  4      2640           1
##  5      2059           1
##  6      1197           1
##  7       238           1
##  8       580           1
##  9      1562           1
## 10       517           1
## 11      2188           0
## 12        36           0
## 13      1949           0
## 14       649           0
## 15       442           0
## 16      2320           0
## 17       394           0
## 18      1856           0
## 19       656           0
## 20      1277           0
```

Below I provide code which processes the data from my test survey (I manually inputted the data of the above workers).

```
survey_df <- read.csv('data/survey_results.csv')

# share employed + avg inc for treatment and control
survey_df %>%
  left_join(rand_ids,
            by = 'worker_id') %>%
  group_by(treat_strat) %>%
  summarise(share_employed = mean(is_employed),
            avg_inc = mean(inc_last14),
            .groups = 'keep') %>%
  pivot_longer(cols = !treat_strat, names_to = 'name',
               values_to = 'value') %>%
  pivot_wider(names_from = treat_strat, names_prefix = 'treat_',
              values_from = value)
```

```
## # A tibble: 2 x 3
##   name           treat_0 treat_1
##   <chr>            <dbl>   <dbl>
## 1 share_employed     0.8       1
## 2 avg_inc        37850     15710
```