-

Assignment 2

Shraddha Ganesh

2024-12-09

```
knitr::opts_chunk$set(
    echo = TRUE,
    message = FALSE,
    warning = FALSE
)
install.packages("readxl", repos = "http://cran.us.r-project.org")
```

```
## package 'readxl' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\shrad\AppData\Local\Temp\Rtmp8W3PVM\downloaded_packages
```

```
install.packages("haven", repos = "http://cran.us.r-project.org")
```

```
## package 'haven' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\shrad\AppData\Local\Temp\Rtmp8W3PVM\downloaded_packages
```

```
install.packages("knitr",repos = "http://cran.us.r-project.org")
```

```
## package 'knitr' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\shrad\AppData\Local\Temp\Rtmp8W3PVM\downloaded_packages
```

```
install.packages("sandwich",repos = "http://cran.us.r-project.org")
```

```
## package 'sandwich' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\shrad\AppData\Local\Temp\Rtmp8W3PVM\downloaded_packages
```

```
install.packages("lmtest",repos = "http://cran.us.r-project.org")
```

```
## package 'lmtest' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\shrad\AppData\Local\Temp\Rtmp8W3PVM\downloaded_packages
```

```
install.packages("kableExtra",repos = "http://cran.us.r-project.org")
```

```
## package 'kableExtra' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\shrad\AppData\Local\Temp\Rtmp8W3PVM\downloaded_packages
```

```
install.packages("tinytex",repos = "http://cran.us.r-project.org")
```

```
## package 'tinytex' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\shrad\AppData\Local\Temp\Rtmp8W3PVM\downloaded_packages
```

```
library(readxl)
library(haven)
library(knitr)
library(sandwich)
library(lmtest)
library(kableExtra)
library(tinytex)
```

```
knitr::opts_chunk$set(
    echo = TRUE,
    message = FALSE,
    warning = FALSE
)
data = read_excel("C:/Users/shrad/Downloads/RestaurantExperimentCleaned.xlsx")
```

QUESTION-1

1. Type of restaurant chosen - Chain
2. Outcome variable chosen - Reservations

Importing data set = "data"

1. Extracting data for chained restaurant = "chain_res"
2. Extracting data for control group = "con_grp"
3. Extracting data for treatment group (alternate algorithm) = "alt_alg_grp".

```
knitr::opts_chunk$set(
    echo = TRUE,
    message = FALSE,
    warning = FALSE
)
data = read_excel("C:/Users/shrad/Downloads/RestaurantExperimentCleaned.xlsx")

chain_res = subset(data, restaurant_type == "chain")
con_grp = subset(data, treatment == 0)
alt_alg_grp = subset(data, treatment == 2)
```

Performing T-Test -> Extracting P-Value "p_value".

```
knitr::opts_chunk$set(
    echo = TRUE,
    message = FALSE,
    warning = FALSE
)
p_value = t.test(alt_alg_grp$reservations, con_grp$reservations)$p.value
```

Calculating Standard Error

1. Calculating Standard Deviation for control group (reservations) = "con_sd"
2. Calculating Standard Deviation for treatment group 2 (reservations) = "alt_sd"
3. Calculating Sample Size for control group (reservations) = "con_ss"
4. Calculating Sample Size for treatment group 2(reservations) = "alt_ss"
5. Calculating Standard Error = "std_error"

```
knitr::opts_chunk$set(
    echo = TRUE,
    message = FALSE,
    warning = FALSE
)
con_sd = round(sd(con_grp$reservations),2)
alt_sd = round(sd(alt_alg_grp$reservations),2)
con_ss = nrow(con_grp)
alt_ss = nrow(alt_alg_grp)
std_error = round(sqrt((con_sd^2/con_ss) + (alt_sd^2/alt_ss)),2)
```

Calculating Confidence Interval

1. Calculating mean of reservations in control group = "con_mean"
2. Calculating mean of reservations in treatment group 2 = "alt_mean"
3. Calculating the difference in mean = "mean_diff"
4. Calculating upper confidence interval = "ci_upper"
5. Calculating lower confidence interval = "ci_lower"

```
knitr::opts_chunk$set(
    echo = TRUE,
    message = FALSE,
    warning = FALSE
)
con_mean = round(mean(con_grp$reservations, na.rm = TRUE),2)
alt_mean = round(mean(alt_alg_grp$reservations, na.rm = TRUE),2)
mean_diff = alt_mean - con_mean
ci_upper = round((mean_diff + 1.96 * std_error),2)
ci_lower = round((mean_diff - 1.96 * std_error),2)
con_interval = c(ci_lower,ci_upper)
```

Printing Output (P-Value, Standard Error, and Confidence Interval)

```
knitr::opts_chunk$set(
    echo = TRUE,
    message = FALSE,
    warning = FALSE
)
print(paste("P-Value:", p_value))
```

```
## [1] "P-Value: 0"
```

```
print(paste("Standard Error:", std_error))
```

```
## [1] "Standard Error: 0.1"
```

```
print(paste("Confidence Interval:", con_interval[1],"-",con_interval[2]))
```

```
## [1] "Confidence Interval: 7.52 - 7.92"
```

QUESTION 2

WITHOUT CLUSTERING

1. Importing Data Set
2. Sub-setting Control Group
3. Sub-setting Treatment Group

```
knitr::opts_chunk$set(
    echo = TRUE,
    message = FALSE,
    warning = FALSE
)
data2 = read_dta("C:/Users/shrad/Downloads/baltab_data_a02.dta")
treatment_data = subset(data2, treat == 1)
control_data = subset(data2, treat == 0)
```

Extracting data for Panel A - Teacher Attendance

1. School open

```
knitr::opts_chunk$set(
    echo = TRUE,
    message = FALSE,
    warning = FALSE
)

mean_treatment_open = round(mean(treatment_data$open, na.rm = TRUE),2)
mean_control_open = round(mean(control_data$open, na.rm = TRUE),2)
mean_diff_open = mean_treatment_open - mean_control_open
std_error_open = round(sd(treatment_data$open, na.rm = TRUE) /
                          sqrt(length(treatment_data$open)),2)

t_test_open = t.test(treatment_data$open, control_data$open)
p_value_open = round(t_test_open$p.value, 3)
n_treatment_open = length(treatment_data$open)
n_control_open = length(control_data$open)
```

Extracting data for Panel B - Student participation

1. Number of students present

```r
knitr::opts_chunk$set(
    echo = TRUE,
    message = FALSE,
    warning = FALSE
)

mean_treatment_students = round(mean(treatment_data$students, na.rm = TRUE),2)
mean_control_students = round(mean(control_data$students, na.rm = TRUE),2)
mean_diff_students = mean_treatment_students - mean_control_students
std_error_students = round(sd(treatment_data$students, na.rm = TRUE) /
                                   sqrt(length(treatment_data$students)),2)

t_test_students = t.test(treatment_data$students, control_data$students)
p_value_students = round(t_test_students$p.value, 3)
n_treatment_students = length(treatment_data$students)
n_control_students = length(control_data$students)
```

Extracting data for Panel D - Teacher performance measures

1. Percentage of children sitting within classroom
2. Percent of teachers interacting with students
3. Blackboards utilized

```r
knitr::opts_chunk$set(
    echo = TRUE,
    message = FALSE,
    warning = FALSE
)

mean_treatment_inside = round(mean(treatment_data$inside, na.rm = TRUE),2)
mean_control_inside = round(mean(control_data$inside, na.rm = TRUE),2)
mean_diff_inside = mean_treatment_inside - mean_control_inside
std_error_inside = round(sd(treatment_data$inside, na.rm = TRUE) /
  sqrt(length(treatment_data$inside)),2)

t_test_inside = t.test(treatment_data$inside, control_data$inside)
p_value_inside = round(t_test_inside$p.value, 3)
n_treatment_inside = length(treatment_data$inside)
n_control_inside = length(control_data$inside)

mean_treatment_interact = round(mean(treatment_data$interact_kids,
                                      na.rm = TRUE),2)
mean_control_interact= round(mean(control_data$interact_kids, na.rm = TRUE),2)
mean_diff_interact = mean_treatment_interact - mean_control_interact
std_error_interact = round(sd(treatment_data$interact_kids, na.rm = TRUE) /
  sqrt(length(treatment_data$interact_kids)),2)

t_test_interact = t.test(treatment_data$interact_kids, control_data$interact_kids)
p_value_interact = round(t_test_interact$p.value, 3)
n_treatment_interact = length(treatment_data$interact_kids)
n_control_interact = length(control_data$interact_kids)

mean_treatment_bbused= round(mean(treatment_data$bbused, na.rm = TRUE),2)
mean_control_bbused = round(mean(control_data$bbused, na.rm = TRUE),2)
mean_diff_bbused = mean_treatment_bbused - mean_control_bbused
std_error_bbused = round(sd(treatment_data$bbused, na.rm = TRUE) /
  sqrt(length(treatment_data$bbused)),2)

t_test_bbused = t.test(treatment_data$bbused, control_data$bbused)
p_value_bbused = round(t_test_bbused$p.value, 3)
n_treatment_bbused = length(treatment_data$bbused)
n_control_bbused = length(control_data$bbused)
```

Tabulating the information extracted

```
knitr::opts_chunk$set(
    echo = TRUE,
    message = FALSE,
    warning = FALSE
)

table = data.frame(
  Variable = c("Panel A: School open ",
               "Panel B: Number of students present",
               "Panel D: Percentage of children sitting within classroom",
               "Panel D: Percent of teachers interacting with students",
               "Panel D: Blackboards utilized"),
  Treatment_Mean = c(mean_treatment_open, mean_treatment_students,
                     mean_treatment_inside, mean_treatment_interact,
                     mean_treatment_bbused),
  Control_Mean = c(mean_control_open, mean_control_students,
                   mean_control_inside, mean_control_interact,
                   mean_control_bbused),
  Mean_Difference = c(mean_diff_open, mean_diff_students,
                      mean_diff_inside, mean_diff_interact,
                      mean_diff_bbused),
  Standard_Error = c(std_error_open, std_error_students,
                     std_error_inside, std_error_interact,
                     std_error_bbused),
  Sample_Treatement = c(n_treatment_open,n_treatment_students,
                        n_treatment_inside,n_treatment_interact,
                        n_treatment_bbused),
  Sample_Control = c(n_control_open,n_control_students,
                        n_control_inside,n_control_interact,
                        n_control_bbused),
  P_Value = c(p_value_open, p_value_students, p_value_inside,
             p_value_interact, p_value_bbused)
)

kable(
  table,
  caption = "<center>Balance Table for Treatment and Control Groups</center>",
  digits = 2
)
```

Table:

Balance Table for Treatment and Control Groups

Variable	Treatment_Mean	Control_Mean	Mean_Difference	Standard_Error	Sample_Treatement	Sample_Control	P_Value
Panel A: School open	0.78	0.58	0.20	0.01	1616	1535	0.00
Panel B: Number of students present	10.50	7.90	2.60	0.25	1616	1535	0.00
Panel D: Percentage of children sitting within classroom	0.57	0.42	0.15	0.01	1616	1535	0.00
Panel D: Percent of teachers interacting with students	0.43	0.33	0.10	0.01	1616	1535	0.00
Panel D: Blackboards utilized	0.92	0.93	-0.01	0.01	1616	1535	0.35

WITH CLUSTERING

Because we do not have variables like School_id, i am clustering the standard errors on the variable “time_code”.

Extracting data for Panel A - Teacher Attendance

1. School open

```
knitr::opts_chunk$set(
    echo = TRUE,
    message = FALSE,
    warning = FALSE
)

mean_treatment_open = round(mean(treatment_data$open, na.rm = TRUE),2)
mean_control_open = round(mean(control_data$open, na.rm = TRUE),2)
mean_diff_open = mean_treatment_open - mean_control_open
model_open = lm(open ~ treat, data = data2)
cluster_se_open = vcovCL(model_open, cluster = ~time_code)
se_open_clustered= round(sqrt(diag(cluster_se_open))[2], 2)


t_test_open <- coeftest(model_open, vcov = cluster_se_open)
p_value_open <- round(t_test_open[2, 4], 3)
```

Extracting data for Panel B - Student participation

1. Number of students present

```
knitr::opts_chunk$set(
    echo = TRUE,
    message = FALSE,
    warning = FALSE
)

mean_treatment_students = round(mean(treatment_data$students, na.rm = TRUE),2)
mean_control_students = round(mean(control_data$students, na.rm = TRUE),2)
mean_diff_students = mean_treatment_students - mean_control_students
model_students = lm(students ~ treat, data = data2)
cluster_se_students = vcovCL(model_students, cluster = ~time_code)
se_students_clustered = round(sqrt(diag(cluster_se_students))[2], 2)

t_test_students= coeftest(model_students, vcov = cluster_se_students)
p_value_students = round(t_test_students[2, 4], 3)
```

Extracting data for Panel D - Teacher performance measures

1. Percentage of children sitting within classroom
2. Percent of teachers interacting with students
3. Blackboards utilized

```
knitr::opts_chunk$set(
    echo = TRUE,
    message = FALSE,
    warning = FALSE
)

mean_treatment_inside = round(mean(treatment_data$inside, na.rm = TRUE),2)
mean_control_inside = round(mean(control_data$inside, na.rm = TRUE),2)
mean_diff_inside = mean_treatment_inside - mean_control_inside
model_inside = lm(inside ~ treat, data = data2)
cluster_se_inside = vcovCL(model_inside, cluster = ~time_code)
se_inside_clustered = round(sqrt(diag(cluster_se_inside))[2], 2)
t_test_inside = coeftest(model_inside, vcov = cluster_se_inside)
p_value_inside = round(t_test_inside[2, 4], 3)

mean_treatment_interact = round(mean(treatment_data$interact_kids,
                                      na.rm = TRUE),2)
mean_control_interact = round(mean(control_data$interact_kids, na.rm = TRUE),2)
mean_diff_interact = mean_treatment_interact - mean_control_interact
model_interact = lm(interact_kids ~ treat, data = data2)
cluster_se_interact = vcovCL(model_interact, cluster = ~time_code)
se_interact_clustered = round(sqrt(diag(cluster_se_interact))[2], 2)
t_test_interact= coeftest(model_interact, vcov = cluster_se_interact)
p_value_interact = round(t_test_interact[2, 4], 3)


mean_treatment_bbused = round(mean(treatment_data$bbused, na.rm = TRUE),2)
mean_control_bbused = round(mean(control_data$bbused, na.rm = TRUE),2)
mean_diff_bbused = mean_treatment_bbused - mean_control_bbused
model_bbused = lm(bbused ~ treat, data = data2)
cluster_se_bbused = vcovCL(model_bbused, cluster = ~time_code)
se_bbused_clustered = round(sqrt(diag(cluster_se_bbused))[2], 2)
t_test_bbused = coeftest(model_bbused, vcov = cluster_se_bbused)
p_value_bbused = round(t_test_bbused[2, 4], 3)
```

Tabulating the information extracted

```
knitr::opts_chunk$set(
    echo = TRUE,
    message = FALSE,
    warning = FALSE
)

table1 = data.frame(
  Variable = c("Panel A: School open ",
               "Panel B: Number of students present",
               "Panel D: Percentage of children sitting within classroom",
               "Panel D: Percent of teachers interacting with students",
               "Panel D: Blackboards utilized"),
  Treatment_Mean = c(mean_treatment_open, mean_treatment_students,
                     mean_treatment_inside, mean_treatment_interact,
                     mean_treatment_bbused),
  Control_Mean = c(mean_control_open, mean_control_students,
                   mean_control_inside, mean_control_interact,
                   mean_control_bbused),
  Mean_Difference = c(mean_diff_open, mean_diff_students,
                      mean_diff_inside, mean_diff_interact,
                      mean_diff_bbused),
  Standard_Error = c(se_open_clustered, se_students_clustered,
                     se_inside_clustered, se_interact_clustered,
                     se_bbused_clustered),
  Sample_Treatement = c(n_treatment_open,n_treatment_students,
                        n_treatment_inside,n_treatment_interact,
                        n_treatment_bbused),
  Sample_Control = c(n_control_open,n_control_students,
                        n_control_inside,n_control_interact,
                        n_control_bbused),
  P_Value = c(p_value_open, p_value_students, p_value_inside,
              p_value_interact, p_value_bbused)
)

kable(
  table1,
  caption = "<center>Balance Table for Treatment and Control Groups</center>",
  digits = 2
)
```

Table:

Balance Table for Treatment and Control Groups

Variable	Treatment_Mean	Control_Mean	Mean_Difference	Standard_Error	Sample_Treatement	Sample_Control	P_Value
Panel A: School open	0.78	0.58	0.20	0.01	1616	1535	0
Panel B: Number of students present	10.50	7.90	2.60	0.06	1616	1535	0
Panel D: Percentage of children sitting within classroom	0.57	0.42	0.15	0.01	1616	1535	0
Panel D: Percent of teachers interacting with students	0.43	0.33	0.10	0.00	1616	1535	0
Panel D: Blackboards utilized	0.92	0.93	-0.01	0.00	1616	1535	0

COMMENTS

An interesting aspect of the experiment is the use of cameras for monitoring instead of traditional methods like inspections. This allows for continuous monitoring without directly interfering with teachers. The cameras are introduced to teachers, making them more accountable in an environment where absenteeism and disengagement are common. While knowing they're being monitored encourages teachers to improve their behavior, these changes might only be temporary, with teachers potentially returning to their old ways once the monitoring stops. While the study highlights the impact of external motivation on performance, it raises the question of how to foster long-term internal motivation.

QUESTION 3

Part-1: Describe what kind of data you would need to measure racial discrimination?

Response: To quantify racial disparities, we would need a variety of data points from job applications.We would require the resumes of all people who applied, along with the applicants' names, which may show their racial identity. In addition to their identities, we would want the results of each of these applications. Did they get a callback, interview, job offer, or rejection? For a thorough study, we would also need more information about the applicant, such as education level, job experience levels, and so on. We can then investigate if ethnic identities influence the outcome of a job application while keeping all other variables constant.

Part-2: The piece mentions that some firms perform worse than others in terms of inclusive recruitment practices. Assume that you are entrusted with the job of resurrecting one such firm's image. What would be your strategy?

Response: First, there should be a standard recruiting strategy for employing people, and regular audits to assure its efficacy. There should also be a diverse recruiting panel, in which persons with varied perspectives and identities participate in the recruitment process. To promote an open culture and a strong redressal system, the company can collaborate with local groups to offer diversity lectures and presentations. Additionally, internal employee-led diversity networks can be established.