Assignment 3

Shraddha Ganesh

2025-01-04

Installing and Loading Relevant Packages

```
install.packages("dplyr", repos = "http://cran.us.r-project.org")
install.packages("sandwich", repos = "http://cran.us.r-project.org")
install.packages("haven", repos = "http://cran.us.r-project.org")
install.packages("gridExtra", repos = "http://cran.us.r-project.org")
install.packages("grid", repos = "http://cran.us.r-project.org")
install.packages("DT", repos = "http://cran.us.r-project.org")
install.packages("stargazer", repos = "http://cran.us.r-project.org")
install.packages("knitr", repos = "http://cran.us.r-project.org")
install.packages("sjPlot", repos = "http://cran.us.r-project.org")

library(sjPlot)
library(knitr)
library(stargazer)
library(dplyr)
library(sandwich)
library(lmtest)
library(haven)
library(gridExtra)
library(grid)
library(DT)
```

Question-1

Importing the Data set.

```
data = read_dta("C:/Users/shrad/Downloads/Lowe_Cric01.dta")
set.seed(123)
column_vars = c(
 "othfriends_balancerecode", "OthCasteClus", "MeanJatiOth",
 "bl_lastyr_work", "bl_played_past", "bl_field_test_catch_tot",
 "bl_age", "bl_bat_test_46_tot", "bl_bowl_test_speed_max",
 "bl_volunteer", "bl_school_yn")
```

Randomly Picking 5 variables from the column list

```
selected_columns = sample(column_vars, 5)
```

```
# Setting The Cluster Variable
cluster_var = data$team_cluster_full
data_sampled = data[sample(nrow(data), 800), ]

Defining The Balance Table
balance_table = data.frame(
 Variable = character(),
 Mean_Diff = numeric(),
 SE = numeric(),
 N = numeric(),
 stringsAsFactors = FALSE)

 Looping through every selected column
for (col_var in selected_columns) {
 Check if the column is numeric, if not, skip it
 if (is.numeric(data_sampled[[col_var]])) {
  Calculate the Dean difference and standard deviation
  mean_diff = mean(data_sampled[[col_var]], na.rm = TRUE)
  sd_diff = sd(data_sampled[[col_var]], na.rm = TRUE)
  Calculating the number of observations after excluding NA's
  n_obs = sum(!is.na(data_sampled[[col_var]]))
  balance_table = rbind(
   balance_table,
   data.frame(
    Variable = col_var,
    Mean_Diff = mean_diff,
    SE = sd_diff,
    N = n_obs
   )
  )
 } else {
  warning(paste("Skipping non-numeric column:", col_var))
 }
}

Printing The Table
balance_table_clean = balance_table
rownames(balance_table_clean) = NULL
table_theme = ttheme_default(
 core = list(fg_params = list(fontsize = 12),
          background = "white",
          border = "black",
          lwd = 1),


colhead = list(fg_params = list(fontsize = 14, fontface = "bold")),
 rowhead = list(fg_params = list(fontsize = 12))
)
table_grob = tableGrob(balance_table_clean, theme = table_theme)
```

```
grid.text("Balance Table for Selected Variables",
      gp = gpar(fontsize = 16, fontface = "bold"),
      y = unit(1, "npc") - unit(2, "lines"))
grid.draw(table_grob)
```

Balance Table for Selected Variables

Variable	Mean_Diff	SE	N
MeanJatiOth	3.9255725	1.2518220	800
bl_school_yn	0.7675000	0.4226903	800
OthCasteClus	0.4354062	0.2514185	800
bl_field_test_catch_tot	4.9250000	1.1910702	800
bl_volunteer	0.4500000	0.4978049	800

QUESTION 2

```
data1 = read_dta("C:/Users/shrad/Downloads/oakland_analysis_cleaned.dta")


data1$delta_bp = as.numeric(as_factor(data1$post_bp)) - as.numeric(as_factor(data1$pre_bp))
data1$delta_bmi = as.numeric(as_factor(data1$post_bmi)) - as.numeric(as_factor(data1$pre_bmi))
data1$delta_dia = as.numeric(as_factor(data1$post_dia)) - as.numeric(as_factor(data1$pre_dia))
data1$delta_cho = as.numeric(as_factor(data1$post_cho)) - as.numeric(as_factor(data1$pre_cho))
data1$delta_flu = as.numeric(as_factor(data1$post_flu)) - as.numeric(as_factor(data1$pre_flu))
outcomes <- list(
 list(pre = "pre_bp", post = "post_bp", delta = "delta_bp"),
 list(pre = "pre_bmi", post = "post_bmi", delta = "delta_bmi"),
 list(pre = "pre_dia", post = "post_dia", delta = "delta_dia"),
 list(pre = "pre_cho", post = "post_cho", delta = "delta_cho"),
 list(pre = "pre_flu", post = "post_flu", delta = "delta_flu"))

panel_a_models = lapply(outcomes, function(x) {
 lm(as.formula(paste(x$pre, "~ black_dr + sl5 + sl10")), data = data1)
})
panel_b_models = lapply(outcomes, function(x) {
 lm(as.formula(paste(x$post, "~ black_dr + sl5 + sl10")), data = data1)
})
panel_c_models = lapply(outcomes, function(x) {
 lm(as.formula(paste(x$delta, "~ black_dr + sl5 + sl10")), data = data1)
})
all_models = c(panel_a_models, panel_b_models, panel_c_models)

stargazer(
 all_models,
```

```
 type = "text",
 title = "Replication of Table 3: Pre, Post, and Delta Regressions",
 dep.var.labels.include = FALSE,
 column.labels = c("BP", "BMI", "Diabetes", "Cholesterol", "Flu Vaccination"),
 covariate.labels = c("Black Doctor", "$5 Incentive", "$10 Incentive"),
 add.lines = list(
  c("Panel", rep("A: Pre-Consultation", 5), rep("B: Post-Consultation", 5), rep("C: Delta (Post - Pre)", 5))
 ),
 omit.stat = c("f", "ser"),
 digits = 3
)
```

Regression Table - Panel A

```
tab_model(
 panel_a_models[[1]], panel_a_models[[2]], panel_a_models[[3]],
 panel_a_models[[4]], panel_a_models[[5]],
 dv.labels = c("BP (Pre)", "BMI (Pre)", "Diabetes (Pre)", "Cholesterol (Pre)", "Flu Vaccination (Pre)"),
 title = "Panel A: Pre-Consultation",
 show.p = TRUE,
 show.ci = FALSE
)
```

Panel A: Pre-Consultation

	BP (Pre)		BMI (Pre)		Diabetes (Pre)		Cholesterol (Pre)		Flu Vaccination (Pre)	
Predictors	*Estimates*	*p*	*Estimates*	*p*	*Estimates*	*p*	*Estimates*	*p*	*Estimates*	*p*
(Intercept)	0.56	**<0.001**	0.52	**<0.001**	0.34	**<0.001**	0.33	**<0.001**	0.19	**<0.001**
Randomized to Black Doctor - CTO Survey	0.03	0.520	0.02	0.557	0.05	0.195	0.01	0.798	-0.01	0.812
Subsidy Level: $5 - CTO Survey	0.03	0.572	-0.06	0.227	0.09	0.076	0.07	0.153	0.19	**<0.001**
Subsidy Level: $10 - CTO Survey	-0.02	0.628	-0.01	0.857	0.03	0.545	-0.01	0.755	0.30	**<0.001**
Observations	637		637		637		637		637	
R^2 / R^2 adjusted	0.002 / -0.002		0.003 / -0.002		0.008 / 0.004		0.006 / 0.001		0.069 / 0.064	

Regression Table - Panel B

```
tab_model(
 panel_b_models[[1]], panel_b_models[[2]], panel_b_models[[3]],
 panel_b_models[[4]], panel_b_models[[5]],
```

```
 dv.labels = c("BP (Post)", "BMI (Post)", "Diabetes (Post)", "Cholesterol (Post)", "Flu Vaccination (Post)"),
 title = "Panel B: Post-Consultation",
 show.p = TRUE,
 show.ci = FALSE)
```

Panel B: Post-Consultation

	BP (Post)		BMI (Post)		Diabetes (Post)		Cholesterol (Post)		Flu Vaccination (Post)	
Predictors	*Estimates*	*p*	*Estimates*	*p*	*Estimates*	*p*	*Estimates*	*p*	*Estimates*	*p*
(Intercept)	0.71	**<0.001**	0.60	**<0.001**	0.37	**<0.001**	0.34	**<0.001**	0.18	**<0.001**
Randomized to Black Doctor - CTO Survey	0.11	**0.001**	0.16	**<0.001**	0.20	**<0.001**	0.26	**<0.001**	0.10	**0.008**
Subsidy Level: $5 - CTO Survey	0.04	0.283	0.02	0.678	0.11	**0.022**	0.07	0.168	0.22	**<0.001**
Subsidy Level: $10 - CTO Survey	-0.03	0.514	-0.01	0.819	0.05	0.249	-0.00	0.933	0.22	**<0.001**
Observations	637		637		637		637		637	
R^2 / R^2 adjusted	0.021 / 0.017		0.031 / 0.026		0.052 / 0.047		0.071 / 0.066		0.060 / 0.055	

Regression Table - Panel C

```
tab_model(
 panel_c_models[[1]], panel_c_models[[2]], panel_c_models[[3]],
 panel_c_models[[4]], panel_c_models[[5]],
 dv.labels = c("BP (Delta)", "BMI (Delta)", "Diabetes (Delta)", "Cholesterol (Delta)", "Flu Vaccination (Delta)"),
 title = "Panel C: Delta (Post - Pre)",
 show.p = TRUE,
 show.ci = FALSE)
```

Panel C: Delta (Post - Pre)

	BP (Delta)		BMI (Delta)		Diabetes (Delta)		Cholesterol (Delta)		Flu Vaccination (Delta)	
Predictors	*Estimates*	*p*	*Estimates*	*p*	*Estimates*	*p*	*Estimates*	*p*	*Estimates*	*p*
(Intercept)	0.15	**<0.001**	0.08	**0.010**	0.03	0.229	0.01	0.836	-0.01	0.842
Randomized to Black Doctor - CTO Survey	0.08	**0.017**	0.14	**<0.001**	0.15	**<0.001**	0.25	**<0.001**	0.11	**0.001**
Subsidy Level: $5 - CTO Survey	0.02	0.694	0.08	0.056	0.02	0.494	-0.00	0.962	0.03	0.485
Subsidy Level: $10 - CTO Survey	-0.00	0.939	-0.00	0.971	0.03	0.465	0.01	0.785	-0.08	**0.047**
Observations	637		637		637		637		637	
R^2 / R^2 adjusted	0.010 / 0.005		0.035 / 0.031		0.045 / 0.040		0.088 / 0.084		0.028 / 0.023	