ohp_project

2024-04-02

```
library(pacman)
p_load(tidyverse, knitr, ggplot2, ggstats, coefplot, haven)

#loading in the data
ohp_data = read_dta("ohp.dta")

ohp_data = na.omit(ohp_data)

#treatment: selected in the lottery
#ohp_all_ever_survey: ever enrolled in Medicaid from 1st notif. date
```

The primary difference between treatment and ohp_all_ever_survey lies within the nature of the Oregon Health Care plan's lottery system. treatment indicates that the individual was selected in the lottery, and ohp_all_ever_survey indicates whether or not the individual was ever on Medicaid following the announcement of the study. Functionally, it is important to distinguish the differences to specify the impact of the OHP's random lottery. If treated, we can include the impact of health insurance in our analysis. ohp_all_ever_survey is useful in identifying others who may be impacted by Medicaid, but not directly related to the natural experiment from OHP.

```
#subset data for control group
ohp_data$treatment = as.numeric(as.character(unclass(ohp_data$treatment)))

grouped_data <- ohp_data %>%
  group_by(treatment)

control_group <- grouped_data %>%
  filter(treatment == 0)

treatment_group = grouped_data %>%
  filter(treatment == 1)

treatment_group = na.omit(treatment_group)

control_group = na.omit(control_group)
```

```
#find means across groups
average_age = mean(control_group$age_inp)
average_dep_pre = mean(control_group$dep_dx_pre_lottery)
average_edu = mean(control_group$edu_inp)
average_gender = mean(control_group$gender_inp)

#create dataframe for averages
averages_table = data.frame(
  Variable = c("Age", "Depression Diagnosis", "Education", "Gender"),
```

```
  Average = c(average_age, average_dep_pre, average_edu, average_gender)
)

#create table
control_averages = kable(averages_table, align = 'c', caption = "Averages of Specific Variables in Cont

print(control_averages)
```

```
##
##
## Table: Averages of Specific Variables in Control Group
##
## |       Variable       |  Average   |
## |:--------------------:|:----------:|
## |         Age          | 40.4609418 |
## | Depression Diagnosis | 0.3484765  |
## |      Education       | 2.2472761  |
## |        Gender        | 0.5710065  |
```

In order to determine that the OHP lottery was truly randomized, we can check the distribution of certain characteristics among both treatment and control groups. If the groups were truly randomized, the averages for a given variable should be similar across both groups.

```
#finding differnce in means
#age, pre lottery depression, education, gender
age_regr <- lm(age_inp ~ treatment, data = ohp_data)
coeff_age_regr = age_regr$coefficients
summary_age_regr = summary(age_regr)
std_age_regr = summary_age_regr$coef[, "Std. Error"]

dep_regr = lm(dep_dx_pre_lottery ~ treatment, data = ohp_data)
coeff_dep_regr = dep_regr$coefficients
summary_dep_regr = summary(dep_regr)
std_dep_regr = summary_dep_regr$coef[, "Std. Error"]

edu_regr = lm(edu_inp ~ treatment, data = ohp_data)
coeff_edu_regr = edu_regr$coefficients
summary_edu_regr = summary(edu_regr)
std_edu_regr = summary_edu_regr$coef[, "Std. Error"]

#insert average gender regression here!!
gend_regr = lm(gender_inp ~ treatment, data = ohp_data)
coeff_gend_regr = gend_regr$coefficients
summary_gend_regr = summary(gend_regr)
std_gend_regr = summary_gend_regr$coef[, "Std. Error"]

#create new table w/coefficients
#defining coefficient values
coef_age = coeff_age_regr[2]
coef_depr_pre = coeff_dep_regr[2]
coef_edu = coeff_edu_regr[2]
coef_gender = coeff_gend_regr[2]
```

```r
#averages_coef_table
averages_coef_table = data.frame(
  Variable = c("Age", "Prior Depression", "Education Level", "Gender"),
  Average = c(average_age, average_dep_pre, average_edu, average_gender),
  Coefficiencts = c(coef_age, coef_depr_pre, coef_edu, coef_gender)
)

#create new table w/standard errors
#define standard error values
std_age = std_age_regr[2]
std_dep = std_dep_regr[2]
std_edu = std_edu_regr[2]
std_gend = std_gend_regr[2]

#averages_coef_std_table
averages_coef_std_table = data.frame(
  Variable = c("Age", "Prior Depression", "Education Level", "Gender"),
  Average = c(average_age, average_dep_pre, average_edu, average_gender),
  Coefficiencts = c(coef_age, coef_depr_pre, coef_edu, coef_gender),
  StandardErrors = c(std_age, std_dep, std_edu, std_gend)
)

averages_coef_table
```

```
##           Variable    Average Coefficiencts
## 1              Age 40.4609418   0.360818604
## 2 Prior Depression  0.3484765  -0.015312068
## 3  Education Level  2.2472761   0.022701952
## 4           Gender  0.5710065  -0.006890903
```

```r
averages_coef_std_table
```

```
##           Variable    Average Coefficiencts StandardErrors
## 1              Age 40.4609418   0.360818604    0.219668203
## 2 Prior Depression  0.3484765  -0.015312068    0.008910657
## 3  Education Level  2.2472761   0.022701952    0.017145544
## 4           Gender  0.5710065  -0.006890903    0.009320498
```

This balance table is consistent with individuals having been randomly assigned to treatment and controls groups given the values of the coefficients. The each coefficient demonstrates a small difference in the values between control and treatment groups, which indicates that both groups have been successfully randomized.

```r
#estimate compliance rate for OHP experiment
compli_regr = lm(ohp_all_ever_survey ~ treatment, data = ohp_data)
summary_compli_regr = summary(compli_regr)
compli_rate = summary_compli_regr$coefficients["treatment", "Estimate"]
print(compli_rate)
```

```
## [1] 0.2506557
```

Compliance rate is likely 25%. This is conducted by regressing whether or not a household was ever enrolled in Medicare after treatment was assigned. The results show that while prior Medicare enrollment was low,

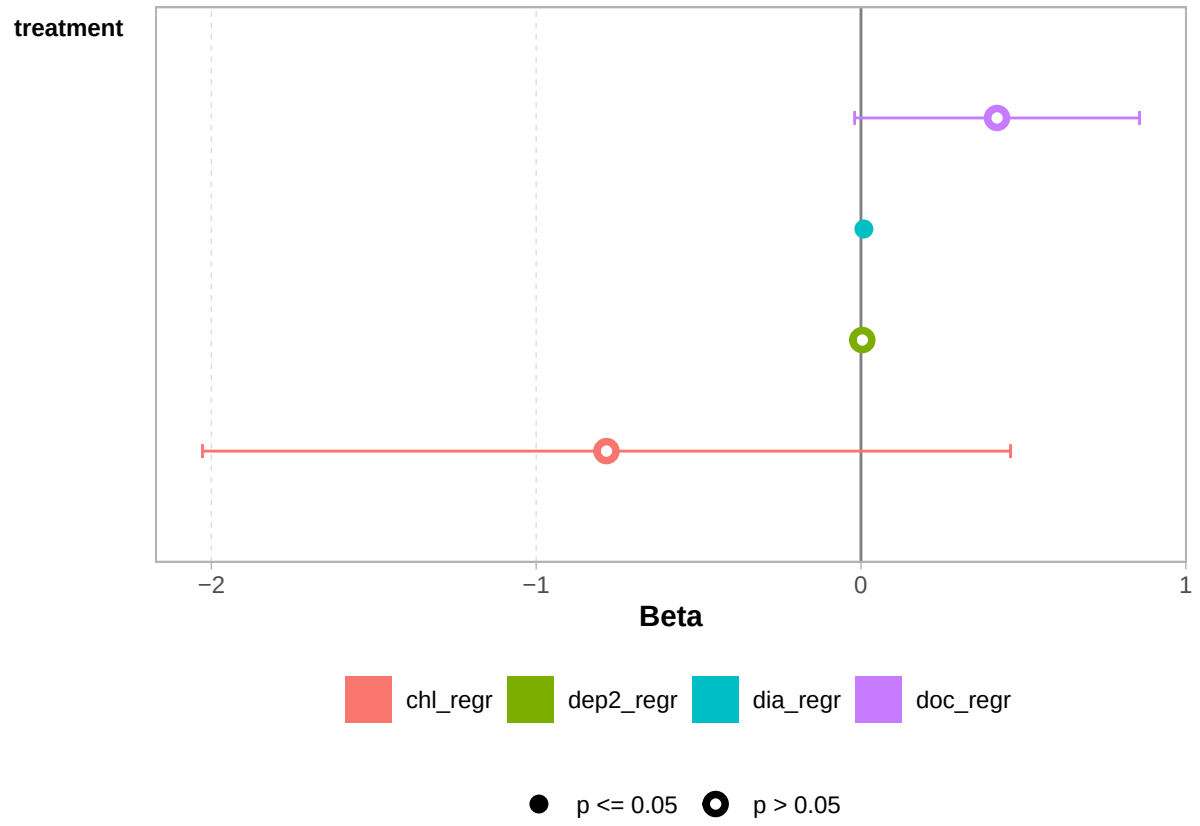
lottery successfully increased Medicare enrollment. Additionally, this effect is captured with high statistical significance, with P-values <2e-16.

```
#estimate ITT regression: pick post lottery variables for analysis
#variables include: cholesterol, depression post/lot, diabetes post/lot, number doc visits

chl_regr = lm(chl_inp ~ treatment, data = ohp_data)
dep2_regr = lm(dep_dx_post_lottery ~ treatment, data = ohp_data)
dia_regr = lm(dia_dx_post_lottery ~ treatment, data = ohp_data)
doc_regr = lm(doc_num_mod_inp ~ treatment, data = ohp_data)

#grab coefficient estimates from regressions
#cholesterol estimates
coeff_chl_regr = chl_regr$coefficients[1]
summary_chl_regr = summary(chl_regr)
std_chl_regr = summary_chl_regr$coef[, "Std. Error"]

#depression estimates
coeff_dep2_regr = dep2_regr$coefficients[1]
summary_dep2_regr = summary(dep2_regr)
std_dep2_regr = summary_dep2_regr$coef[, "Std. Error"]

#diabetes estimates
coeff_dia_regr = dia_regr$coefficients[1]
summary_dia_regr = summary(dia_regr)
std_dia_regr = summary_dia_regr$coef[, "Std. Error"]

#doctor visits estimates
coeff_doc_regr = doc_regr$coefficients[1]
summary_doc_regr = summary(doc_regr)
std_doc_regr = summary_doc_regr$coef[, "Std. Error"]

#create table for regression estimates

itt_regr_table = data.frame(
  Variable = c("Cholesterol", "Depression Diagnosis", "Diabetes Diagnosis", "Number of Doctor Visits"),
  Coefficients = c(coeff_chl_regr, coeff_dep2_regr, coeff_dia_regr, coeff_doc_regr),
  StandardErrors = c(std_chl_regr, std_dep2_regr, std_dia_regr, std_doc_regr)
)
print(itt_regr_table)
```

```
##                  Variable Coefficients StandardErrors
## 1             Cholesterol 205.88648507    0.458580665
## 2    Depression Diagnosis   0.04949215    0.634575315
## 3      Diabetes Diagnosis   0.01144968    0.003006766
## 4 Number of Doctor Visits   5.64727608    0.004160706
## 5             Cholesterol 205.88648507    0.001711837
## 6    Depression Diagnosis   0.04949215    0.002368807
## 7      Diabetes Diagnosis   0.01144968    0.161706877
## 8 Number of Doctor Visits   5.64727608    0.223766941
```

It seems as though the ITT effect here was large, given that all outcomes variables were affected when given treatment.

```r
#graph w/estimates + std errors
plot1 = ggcoef_compare(list("chl_regr" = chl_regr, "dep2_regr" = dep2_regr, "dia_regr" = dia_regr, "doc_
plot1
```




```r
#treatment on the treated effect (ATET)
#ATET = ITT / effect of winning lottery
itt_chl = coeff_chl_regr/compli_rate
itt_dep2 = coeff_dep2_regr/compli_rate
itt_dia = coeff_dia_regr/compli_rate
itt_doc = coeff_doc_regr/compli_rate

#create new data frame --> new table
atet_effect_table = data.frame(
  Variable = c("Cholesterol", "Depression Diagnosis", "Diabetes Diagnosis", "Number of Doctors Visits")
  ATET_Effect = c(itt_chl, itt_dep2, itt_dia, itt_doc)
)
```

In order to calculate this estimate, I divided the coefficients from each variable (cholesterol, depression diagnosis, diabetes diagnosis, and number of doctors visits) by the compliance rate for the study. This outputs the given results.

The ATET results measure the impact of a treatment on the treated individuals, providing insight into how the treatment affects those who actually undergo the treatment, rather than considering the effect on the entire population including both treated and untreated individuals.

Worrying about attrition bias is a necessary component of long term studies. Firstly, defining attrition bias is important. Attrition bias represents the loss of any observations over time, stemming from failing to gather

data in some capacity. This can come from lack of desire to continue participation, forgetfulness, dropping out of the study, or even death.

In any long term data collection process, attrition bias can play a role in data gathering. The authors of Taubman et al. (2014) go so far as to address how potential attrition bias might be mitigated. This is clear evidence that there is likely attrition bias within the available data. Moreover, acknowledging how the duration of the study can affect attrition is a useful first step when considering the implications for our data.