Boseong Yun - Problem Set 4

Boseong Yun

11/17/2020

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

# Install and load packages --------------------------------------
packages <- c(
  "tidyverse",
  "haven",
  "readxl",
  "knitr",
  "broom",
  "dendroTools",
  "zoo",
  "patchwork",
  "plm",
  "lmtest",
  "car",
  "coefplot",
  "ggthemes",
  "lfe",
  "Hmisc",
  "stargazer",
  "fixest",
  "jtools"
)

# Change to install = TRUE to install the required packages--------------------------------------
pacman::p_load(packages, character.only = TRUE, install = FALSE)

# Load dataset ----------------------------------------------------
ftp_ar <- read_dta("ftp_ar.dta")
ftp_srv <- read_dta("ftp_srv.dta")

# Data Preparation ----------------------------------------------------
merged <- ftp_ar %>%
  inner_join(ftp_srv, by = "sampleid") %>%
  rename_at(vars(ends_with(".x")), funs(str_replace(., "\\..$",""))) %>%
  select(-ends_with(".y"))
```

```
## Warning: `funs()` is deprecated as of dplyr 0.8.0.
## Please use a list of either functions or lambdas:
##
##   # Simple named list:
##   list(mean = mean, median = median)
```

```
##
##   # Auto named with `tibble::lst()`:
##   tibble::lst(mean, median)
##
##   # Using lambdas
##   list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_warnings()` to see where this warning was generated.
```

1. Find the means of quarterly employment for each quarter from 10 quarters prior to RA to 19 quarters following RA (recall from the documentation that the quarter of random assignment, which would naturally be designated as quarter 0, is actually denoted as quarter 1). What is the longest pre-period that you could analyze, ensuring that you have data for everyone in the sample? The longest post-period?

Answer: The longest pre-proid that I could analyze ensuring that I have data for everyone is emppq9 (9 quarters prior to intervention) and the longest post-period is empq17 (16 quarters after intervention) because there are missing values in emppp10, empq18, empq19, and empq20.

```r
# pdf pg 20. EMPQ1 - EMPQ20: Employed Quarters 1:20
# pdf pg 34. EMPPQ1 - EMPPQ10: employed pre-quarter qtr 1:10
ftp_ar %>%
  select(starts_with("emppq"), starts_with("empq")) %>%
  summarize_all(funs(mean(., na.rm = TRUE))) %>%
  round_df(digits = 5) %>%
  t() %>%
  kable(caption = "The Means of Quaterly Employment")
```

Table 1: The Means of Quaterly Employment

emppq10	0.00000
emppq9	0.05755
emppq8	0.13535
emppq7	0.22629
emppq6	0.26892
emppq5	0.27531
emppq4	0.27602
emppq3	0.29023
emppq2	0.29378
emppq1	0.32718
empq1	0.34813
empq2	0.37655
empq3	0.40497
empq4	0.42025
empq5	0.42984
empq6	0.44050
empq7	0.45542
empq8	0.47496
empq9	0.48632
empq10	0.48845
empq11	0.47496
empq12	0.47815
empq13	0.47851
empq14	0.48632
empq15	0.48703
empq16	0.48703

Table 1: The Means of Quaterly Employment

empq17	0.49449
empq18	0.48152
empq19	0.48861
empq20	0.48502

2. Reconfigure the data so you have one record per person per quarter. Compute mean employment rates by treatment status, before and after treatment. Treatment status is TLyes; the date of random assignment divides the sample period into before and after (consider the period of random assignment itself to be "after"). Were employment rates similar between the treatment and control groups prior to treatment?

Answer: No, the employment rates were not similar between the treatment and control groups prior to the treatment. The difference in means test shows that we reject the null hypothesis that the difference in means is equal to 0.

```
# Creatin the TLyes variable by limiting to those who responded explicitly------------------------------
data <- merged %>%
  mutate(TLyes = ifelse(fmi2 == 1, 1,
                        ifelse(fmi2 == 2, 0, NA)))

# Reconfiguring the data ---------------------------------------------------------------------------
data_rec <- data %>%
  select(TLyes, starts_with("emppq"), starts_with("empq")) %>%
  pivot_longer(-TLyes) %>%
  mutate(when = ifelse(str_detect(name, "emppq") == TRUE, "before", "after"))


data_rec %>%
  filter(!is.na(TLyes)) %>%
  group_by(when, TLyes) %>%
  dplyr::summarize(mean = mean(value, na.rm = TRUE)) %>%
  kable(caption = "Mean Employment Rates by Treatment Status and Treatment Period")
```

```
## `summarise()` regrouping output by 'when' (override with `.groups` argument)
```

Table 2: Mean Employment Rates by Treatment Status and Treatment Period

when	TLyes	mean
after	0	0.4299208
after	1	0.4922737
before	0	0.1917808
before	1	0.2115616

```
# creating treat and control dataframes for before-treatment periods
data_rec2_treat <- data_rec %>%
  filter(when == "before" & TLyes == 1)

data_rec2_control <- data_rec %>%
  filter(when == "before" & TLyes == 0)

# Source: https://stackoverflow.com/questions/52811684/running-a-two-sample-t-test-with-unequal-sample-
```

```r
t.test(data_rec2_treat$value, data_rec2_control$value, alternative = "two.sided")
```

```
##
##  Welch Two Sample t-test
##
## data:  data_rec2_treat$value and data_rec2_control$value
## t = 2.4072, df = 7745.6, p-value = 0.0161
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.003672262 0.035889218
## sample estimates:
## mean of x mean of y
## 0.2115616 0.1917808
```

3. (a) Run a DD regression. The dependent variable should be quarterly employment status, and the explanatory variables should include person dummies, quarter dummies, and an interaction between TLyes and a post-treatment dummy. Be sure to restrict the sample period to the quarters you identified in question 1.

   (b) Now run the same regression, clustering the standard errors by sampleid. Explain why the standard errors change the way they do. Which standard errors should you report?

   (c) How do you interpret the coefficient on the interaction between TLyes and the post-treatment dummy?

Asnwer: The standard errors increase when we cluster the standard errors by the sampleid. This is because clustering reduces the amount of information used to estimate the variance and thereby increases the standard errors. Also, we could suspect correlation in cluster. I should report the standard errors clustered by sampleid because we want to account for this corrleation. The interpreation of the coefficient on the interaction between TLyes and the post-treatment dummy is that the treatment increases the average employment rates by approximately 4.3 percent for those who are treated after the treatment period.

```r
# person dummies, quarter dummies, and an interaction between TLyes and a post-treatment dummy---------

# person dummies = sampleid / quarter dummies = quarter

# Creating before-RA dataframe--------------------------------------
before <- data %>%
  select(sampleid, TLyes, starts_with("emppq")) %>%
  pivot_longer(-c(sampleid, TLyes)) %>%
  separate(name, into = c("type", "qtr"), sep = 5) %>%
  mutate(post_treat = 0)

# Creating after-RA dataframe--------------------------------------
after <- data %>%
  select(sampleid, TLyes, starts_with("empq")) %>%
  pivot_longer(-c(sampleid, TLyes)) %>%
  separate(name, into = c("type", "qtr"), sep = 4) %>%
  mutate(post_treat = 1)

# Creating the combined dataframe--------------------------------------
data2 <- before %>%
  filter(!is.na(TLyes)) %>%
  bind_rows(after) %>%
  unite("quarter", type:qtr, sep = "") %>%
  filter(quarter %nin% c("emppq10", "empq18", "empq19", "empq20"))
```

```r
# Fixed Effects model using ---------------------------------------
# I have referenced the following website to specify functional forms
#: https://jbhender.github.io/Stats506/F18/GP/Group8.html
#: https://stackoverflow.com/questions/19017828/first-difference-linear-panel-model-variance-in-r-and-s


# Q1. Using the plm function
fe_twoway <- plm(value ~ TLyes*post_treat,
    index = c("sampleid", "quarter"),
    model = "within",
    effect = "twoway", # twoway for the fixed effects model
    data = data2)

# Q2. clustering the standard errors by the sampleid
# I have referenced the following website to cluster
# https://stats.stackexchange.com/questions/10017/standard-error-clustering-in-r-either-manually-or-in-p
# this shows multiwaycov library (not being supported in the current ver)
#
# http://www.richard-bluhm.com/clustered-ses-in-r-and-stata-2/
# shows that Stata uses HC1
#
# :https://stats.stackexchange.com/questions/205604/cluster-definition-in-vcovhc/205607
# this shows that the vocvHC uses the grouping variables specified in the plm function

# Q2. clutser by sampleid
fe_group <- coeftest(
  fe_twoway,
  vcov = vcovHC(fe_twoway, type = "HC1", cluster = "group")
  ) %>%
  tidy()

# Q3.
tidy(fe_twoway) %>%
  bind_rows(fe_group) %>%
  mutate(cluster_sampleid = c("No", "Yes")) %>%
  round_df(digits = 5) %>%
  kable(caption = "Quarterly Employment Rates")
```

Table 3: Quarterly Employment Rates

term	estimate	std.error	statistic	p.value	cluster_sampleid
TLyes:post_treat	0.05005	0.01064	4.70288	0.00000	No
TLyes:post_treat	0.05005	0.01998	2.50472	0.01226	Yes

4. (a) Construct a test for parallel pre-treatment trends. Do you reject the null hypothesis?
(b) Plot the relevant results.
(c) Explain why this test is neither necessary nor sufficient for the DD estimator to identify the ATT.

Answer: No, the partial-F test shows that we fail to reject the null hypothesis. This test is neither necessary nor sufficient for the DD estimator to identify the ATT because this is a proxy test. Specifically, one cannot directly test the parallel trend assumption becauase it involves counterfactual outcomes that we cannot observe. Therefore, this test is neither necessary nor sufficient for the DD estimator to identify the ATT.

```r
# Creating quarter dummies for testing pre-trends
data_pre <- data2 %>%
  mutate(
    emppq9 = ifelse(quarter == "emppq9", 1, 0),
    emppq8 = ifelse(quarter == "emppq8", 1, 0),
    emppq7 = ifelse(quarter == "emppq7", 1, 0),
    emppq6 = ifelse(quarter == "emppq6", 1, 0),
    emppq5 = ifelse(quarter == "emppq5", 1, 0),
    emppq4 = ifelse(quarter == "emppq4", 1, 0),
    emppq3 = ifelse(quarter == "emppq3", 1, 0),
    emppq2 = ifelse(quarter == "emppq2", 1, 0),
    emppq1 = ifelse(quarter == "emppq1", 1, 0),
  )

# Constructing a test for parallel pre-treatment trends: the period immediately
# before the treatment is going to be the base period
mod_pre <- plm(value ~ TLyes:post_treat + TLyes:emppq2 + TLyes:emppq3 +
              TLyes:emppq4 + TLyes:emppq5 + TLyes:emppq6 + TLyes:emppq7 +
              TLyes:emppq8 + TLyes:emppq9,
    index = c("sampleid", "quarter"),
    effect = "twoways",
    model = "within",
    data = data_pre)

# Constructing the test---------------------------------------------------
coeftest(mod_pre, vcov = vcovHC(mod_pre, type = "HC1", cluster = "group")) %>%
  tidy() %>%
  round_df(digits = 5) %>%
  kable(caption = "Pre-treatment Trends")
```

Table 4: Pre-treatment Trends

term	estimate	std.error	statistic	p.value
TLyes:post_treat	0.05012	0.02885	1.73759	0.08230
TLyes:emppq2	-0.02215	0.02573	-0.86094	0.38928
TLyes:emppq3	-0.01048	0.02918	-0.35919	0.71945
TLyes:emppq4	-0.00703	0.03130	-0.22458	0.82231
TLyes:emppq5	-0.00872	0.03388	-0.25725	0.79698
TLyes:emppq6	0.01771	0.03396	0.52130	0.60216
TLyes:emppq7	0.02390	0.03394	0.70400	0.48144
TLyes:emppq8	0.04126	0.03394	1.21585	0.22406
TLyes:emppq9	-0.03379	0.03317	-1.01852	0.30844

```r
# Partial F-Test: from aod package----------------------------------------------------
aod:::wald.test(b = coef(mod_pre),
                Sigma = vcovHC(mod_pre, type = "HC1", cluster = "group"),
                Terms = 2:9)
```

```
## Wald test:
## ----------
##
## Chi-squared test:
## X2 = 13.1, df = 8, P(> X2) = 0.11
```

```r
# Partial F-test: from lfe package-------------------------------------------------------------
fe_pre <- felm(value ~ TLyes:post_treat + TLyes:emppq2 + TLyes:emppq3 + TLyes:emppq4 +
                 TLyes:emppq5 +  TLyes:emppq6 + TLyes:emppq7 + TLyes:emppq8 + TLyes:emppq9 |
                 sampleid + quarter | 0 | sampleid, data = data_pre)

# Saving the name of the coefficients
var_name <- names(coef(fe_pre))[-1]

# Running waldtest from lfe package
waldtest(fe_pre, var_name)
```

```
##             p          chi2           df1           p.F             F           df2
##    0.1079413    13.1153676     8.0000000     0.1094483     1.6394209 1030.0000000
## attr(,"formula")
## ~`TLyes:emppq2` | `TLyes:emppq3` | `TLyes:emppq4` | `TLyes:emppq5` |
##     `TLyes:emppq6` | `TLyes:emppq7` | `TLyes:emppq8` | `TLyes:emppq9`
## <environment: 0x7f9b3e7b0a90>
```

```r
# Plot ---------------------------------------------------------------------------
coeftest(mod_pre, vcov = vcovHC(mod_pre, type = "HC1", cluster = "group")) %>%
  tidy(conf.int = TRUE) %>%
  mutate(quarters = c(0, -(2:9))) %>%
  ggplot(aes(x = quarters, y = estimate)) +
   geom_pointrange(aes(
    ymin = conf.low,
    ymax = conf.high)
    ) +
  geom_line() +
  geom_hline(yintercept = 0, color = "black") +
  geom_vline(xintercept = -1, color = "red") +
  scale_x_continuous(breaks = seq(-9, 0), limits = c(-9, 0)) +
  labs(
    title = "Coefficient Plots for Employment Status",
    subtitle = "(negative quarters indicate pre-intervention quarters)",
    x = "Quarters",
    y = "Esimtates"
  ) +
    theme_stata() +
  annotate(label = "base period (t-1)", "text", x = -1, y = -0.05, color = "red") +
  annotate(label = "Post-Treatment Effect =>", "text", x = -1.5, y = 0.095)
```

5. Now estimate period-specific treatment effects that vary freely over the post-treatment period and plot the estimates. Are the period-specific estimates significant? If not, does this concern you? Explain.

Answer: Not all of the period-specific treatment effects are siginificant. As the coefficient plot shows, there are many quarters where the confidence intervals of their coefficients include zero. This can be a little bit concerning because it means that the treatment effects were statistically siginificant only for two quarters, *empq9, empq10*, at 0.05 sigificance level. That is, it could be inappropriate to say that the intervention was effective for the whole range of quarters from the constant treatment effect when in fact there were two quarters where period-specific treatments were statistically significant. This can have policy implications depending on the costs and duration of policies. If you are generally interested in the treatment effect apart from period-specific treatment effects, however, you can be less concerned about it.

```
# I have utilized the following website to specify the options
#: https://stackoverflow.com/questions/28359491/r-plm-time-fixed-effect-model

data_whole <- data_pre %>%
  mutate(
    empq1 = ifelse(quarter == "empq1", 1, 0),
    empq2 = ifelse(quarter == "empq2", 1, 0),
    empq3 = ifelse(quarter == "empq3", 1, 0),
    empq4 = ifelse(quarter == "empq4", 1, 0),
    empq5 = ifelse(quarter == "empq5", 1, 0),
    empq6 = ifelse(quarter == "empq6", 1, 0),
    empq7 = ifelse(quarter == "empq7", 1, 0),
    empq8 = ifelse(quarter == "empq8", 1, 0),
    empq9 = ifelse(quarter == "empq9", 1, 0),
    empq10 = ifelse(quarter == "empq10", 1, 0),
```

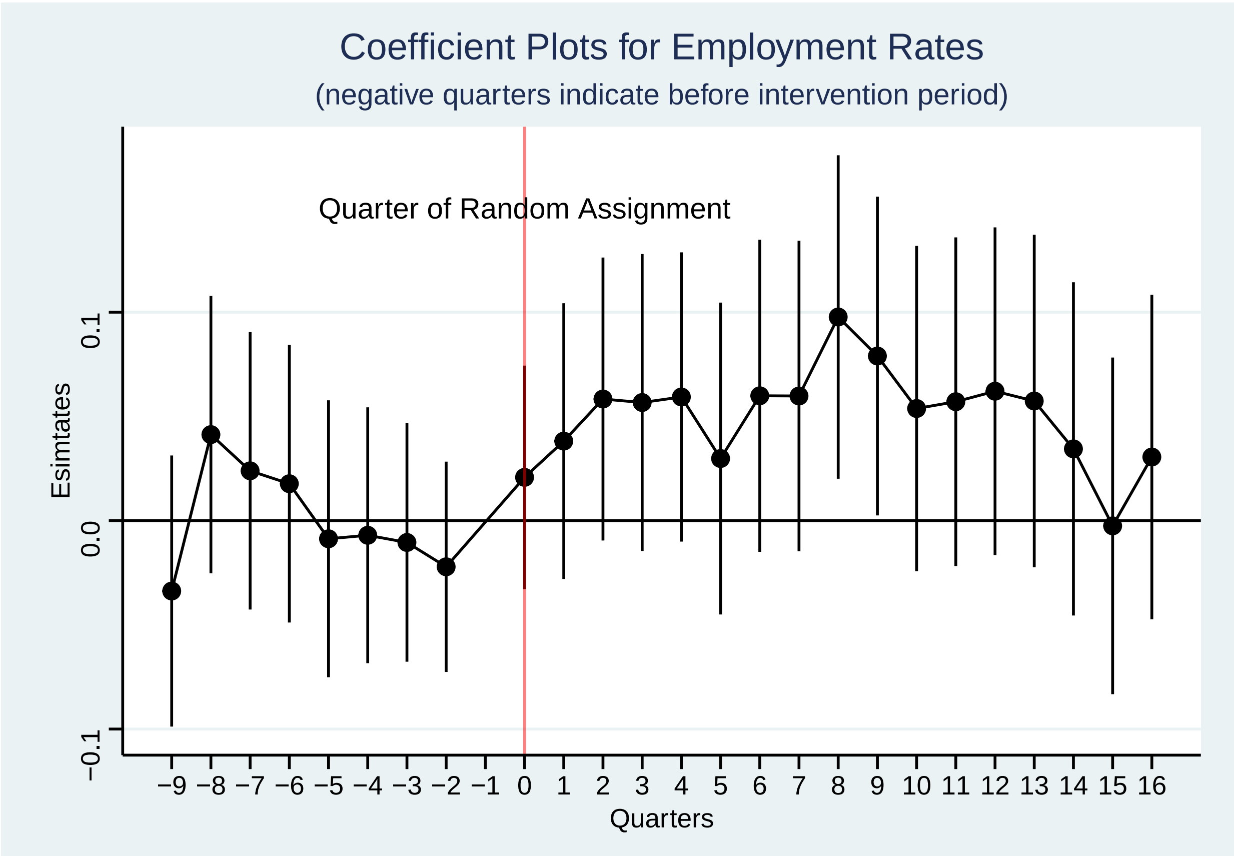
```
    empq11 = ifelse(quarter == "empq11", 1, 0),
    empq12 = ifelse(quarter == "empq12", 1, 0),
    empq13 = ifelse(quarter == "empq13", 1, 0),
    empq14 = ifelse(quarter == "empq14", 1, 0),
    empq15 = ifelse(quarter == "empq15", 1, 0),
    empq16 = ifelse(quarter == "empq16", 1, 0),
    empq17 = ifelse(quarter == "empq17", 1, 0),
  )


# period-secific treatment effects
fe_period <- plm(value ~ TLyes:emppq2 + TLyes:emppq3 + TLyes:emppq4 +
                   TLyes:emppq5 + TLyes:emppq6 + TLyes:emppq7 + TLyes:emppq8 +
                   TLyes:emppq9 + TLyes:empq1 + TLyes:empq2 + TLyes:empq3 +
                   TLyes:empq4 + TLyes:empq5 + TLyes:empq6 + TLyes:empq7 +
                   TLyes:empq8 + TLyes:empq9 + TLyes:empq10 + TLyes:empq11 +
                   TLyes:empq12 + TLyes:empq13 + TLyes:empq14 + TLyes:empq15 +
                   TLyes:empq16 + TLyes:empq17,
    index = c("sampleid", "quarter"),
    model = "within",
    effect = "twoways",
    data = data_whole)

# Test Results
coeftest(fe_period, vcov = vcovHC(fe_period, type = "HC1", cluster = "group")) %>%
  tidy() %>%
  round_df(digits = 5) %>%
  kable(caption = "Quarter Employment Rates (Period-Speicfic)")
```

Table 5: Quarter Employment Rates (Period-Speicfic)

term	estimate	std.error	statistic	p.value
TLyes:emppq2	-0.02215	0.02574	-0.86069	0.38942
TLyes:emppq3	-0.01048	0.02919	-0.35909	0.71953
TLyes:emppq4	-0.00703	0.03131	-0.22451	0.82236
TLyes:emppq5	-0.00872	0.03389	-0.25718	0.79704
TLyes:emppq6	0.01771	0.03397	0.52115	0.60227
TLyes:emppq7	0.02390	0.03395	0.70379	0.48157
TLyes:emppq8	0.04126	0.03395	1.21548	0.22419
TLyes:emppq9	-0.03379	0.03318	-1.01822	0.30859
TLyes:empq1	0.02076	0.02735	0.75892	0.44791
TLyes:empq2	0.03814	0.03375	1.13029	0.25836
TLyes:empq3	0.05834	0.03463	1.68470	0.09206
TLyes:empq4	0.05666	0.03634	1.55912	0.11898
TLyes:empq5	0.05932	0.03540	1.67578	0.09379
TLyes:empq6	0.02979	0.03817	0.78044	0.43514
TLyes:empq7	0.05990	0.03820	1.56792	0.11691
TLyes:empq8	0.05979	0.03800	1.57333	0.11565
TLyes:empq9	0.09770	0.03959	2.46786	0.01360
TLyes:empq10	0.07897	0.03902	2.02387	0.04299
TLyes:empq11	0.05378	0.03981	1.35103	0.17670
TLyes:empq12	0.05705	0.04022	1.41842	0.15608
TLyes:empq13	0.06208	0.04010	1.54802	0.12163

term	estimate	std.error	statistic	p.value
TLyes:empq14	0.05739	0.04069	1.41049	0.15841
TLyes:empq15	0.03442	0.04078	0.84394	0.39871
TLyes:empq16	-0.00251	0.04120	-0.06101	0.95135
TLyes:empq17	0.03052	0.03972	0.76839	0.44226

```r
# Creating a plot
coeftest(fe_period, vcov = vcovHC(fe_period, type = "HC1", cluster = "group")) %>%
  tidy(conf.int = TRUE) %>%
  mutate(quarters = c(-(2:9), (0:16))) %>%
  ggplot(aes(x = quarters, y = estimate)) +
   geom_pointrange(aes(
    ymin = conf.low,
    ymax = conf.high)
    ) +
  geom_line() +
  geom_hline(yintercept = 0, color = "black") +
  geom_vline(xintercept = 0, color = "red", alpha = 0.5) +
  scale_x_continuous(breaks = -9:16, limits = c(-9, 16)) +
  labs(
    title = "Coefficient Plots for Employment Rates",
    subtitle = "(negative quarters indicate before intervention period)",
    x = "Quarters",
    y = "Esimtates"
  ) +
  annotate(label = "Quarter of Random Assignment", "text", x = 0, y = 0.15) +
  theme_stata()
```



6. Compare the mean of the period-specific treatment effects from question 5 to the constant post-treatment effect from question 3. Are they similar? How would you test the hypothesis that they are the same?

Answer: In order to determine the similiarty, one would have to use a partial F test to see test the hypothesis that they are the same. As Terrence has kindly mentioned in this post, we are testing the hypothesis that they are smiliar by looking at whether the coefficients cancel out between the pooled estimates and period-specific estimates. Based on the similar framework, I need to test the hypothesis using a partial F-test. The results of the test show that we fail to reject the null hypothesis that they are similar.

```
# Getting Means -------------------------------------------------------
constant_mean <-
  coeftest(fe_twoway, vcovHC(fe_twoway, type = "HC1", cluster = "group")) %>%
  tidy() %>%
  .$estimate

period_specific_mean <-
  coeftest(fe_period, vcov = vcovHC(fe_period, type = "HC1", cluster = "group")) %>%
  tidy() %>%
  slice(9:25) %>%
  .$estimate %>%
  mean(.)


data.frame(
  "Constant Effect" = constant_mean,
  "Period-Specific Effect" = period_specific_mean
```

```r
) %>%
  kable(caption = "Post Treatment Effects")
```

Table 6: Post Treatment Effects

Constant.Effect	Period.Specific.Effect
0.0500453	0.0501226

```r
# Test -------------------------------------------------------

# Q3
fe_combined <- felm(value ~ TLyes*post_treat + TLyes:empq2 + TLyes:empq3 +
                      TLyes:empq4 + TLyes:empq5 +  TLyes:empq6 + TLyes:empq7 + TLyes:empq8 +
                      TLyes:empq9 + TLyes:empq10 +  TLyes:empq11 + TLyes:empq12 + TLyes:empq13 +
                      TLyes:empq14 + TLyes:empq15 + TLyes:empq16 + TLyes:empq17 |
                      sampleid + quarter | 0 | sampleid,  data = data_whole)


fe_vars <- names(coef(fe_combined))[-1:-3]

waldtest(fe_combined, fe_vars)
```

```
##            p         chi2          df1          p.F            F          df2
##    0.6893080   12.7725741   16.0000000    0.6887641    0.7982859 1030.0000000
## attr(,"formula")
## ~`TLyes:empq2` | `TLyes:empq3` | `TLyes:empq4` | `TLyes:empq5` |
##     `TLyes:empq6` | `TLyes:empq7` | `TLyes:empq8` | `TLyes:empq9` |
##     `TLyes:empq10` | `TLyes:empq11` | `TLyes:empq12` | `TLyes:empq13` |
##     `TLyes:empq14` | `TLyes:empq15` | `TLyes:empq16` | `TLyes:empq17`
## <environment: 0x7f9b28512a00>
```

```r
# Using pFtest

fe_after <- plm(value ~ TLyes*post_treat + TLyes:empq1 + TLyes:empq2 + TLyes:empq3 +
                  TLyes:empq4 + TLyes:empq5 +  TLyes:empq6 + TLyes:empq7 + TLyes:empq8 +
                  TLyes:empq9 + TLyes:empq10 +  TLyes:empq11 + TLyes:empq12 + TLyes:empq13 +
                  TLyes:empq14 + TLyes:empq15 + TLyes:empq16 + TLyes:empq17,
                index = c("sampleid", "quarter"),
                model = "within",
                effect = "twoway",
                data = data_whole)

fe_twoway <- plm(value ~ TLyes*post_treat,
    index = c("sampleid", "quarter"),
    model = "within",
    effect = "twoway", # twoway for the fixed effects model
    data = data2)


pFtest(fe_after, fe_twoway)
```

```
##
##  F test for twoways effects
```

```
##
## data:  value ~ TLyes * post_treat + TLyes:empq1 + TLyes:empq2 + TLyes:empq3 +  ...
## F = 0.80248, df1 = 16, df2 = 25733, p-value = 0.6844
## alternative hypothesis: significant effects
```

7. Return now to the model that imposes constant treatment effects during the post-treatment period. Estimate the effect of treatment using IV, where e is an instrument for TLyes. This is similar to what you did for problem set 2, only now the data are configured differently. Are the DD and IV estimates similar?

Answer: No, the DD and IV estimates are different from each other. Specifically, the IV estimates have higher esimates and standard errors than those of the DD estimates.

```r
# Creating before iv dataframe
before_iv <- data %>%
  select(sampleid, e, TLyes, starts_with("emppq")) %>%
  pivot_longer(-c(sampleid, TLyes, e)) %>%
  separate(name, into = c("type", "qtr"), sep = 5) %>%
  mutate(post_treat = 0)

# Creating after iv dataframe
after_iv <- data %>%
  select(sampleid, e, TLyes, starts_with("empq")) %>%
  pivot_longer(-c(sampleid, TLyes, e)) %>%
  separate(name, into = c("type", "qtr"), sep = 4) %>%
  mutate(post_treat = 1)

# Creating the combined iv dataframe
combined_iv  <- before_iv %>%
  bind_rows(after_iv) %>%
  unite("quarter", type:qtr, sep = "") %>%
  filter(quarter %nin% c("emppq10", "empq18", "empq19", "empq20")) %>%
  mutate(
    treat = TLyes * post_treat,
    iv = e * post_treat
  )


# fe3 mod
fe3_mod <- felm(value ~ TLyes*post_treat | sampleid + quarter | 0 | sampleid,  data = data2)


# Running a panel regression
iv_mod <- felm(value ~ 0 | sampleid + quarter | (treat ~ iv) | sampleid, data = combined_iv)
```

```r
stargazer(iv_mod, fe3_mod,
          title = "The Differences between IV and DD",
          column.labels = c("IV", "DD"))
```

% Table created by stargazer v.5.2.2 by Marek Hlavac, Harvard University. E-mail: hlavac at fas.harvard.edu
% Date and time: Wed, Dec 02, 2020 - 04:31:52

Table 7: The Differences between IV and DD

	Dependent variable:	
	value	
	IV	DD
	(1)	(2)
‘treat(fit)‘	0.152^{***}	
	(0.044)	
TLyes		
		(0.000)
post_treat		
		(0.000)
TLyes:post_treat		0.050^{**}
		(0.020)
Observations	26,806	26,806
R^2	0.357	0.359
Adjusted R^2	0.331	0.333
Residual Std. Error (df = 25749)	0.397	0.396
Note:	*p<0.1; **p<0.05; ***p<0.01	