Problem Set 3

Daniel Koslovsky

2025-05-03

```
#PSET3

#Consultations:
#- I consulted with Chat-GPT 4o on how to estimate the regression with clustered
#standard errors in question 3, when lm_robust could not run it. See attached
#transcript.

library(fixest)
library(modelsummary)
library(haven)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become error
```

```
library(estimatr)
library(stringr)
library(did)


#Question 1
ftp_ar <- read_dta("ftp_ar.dta")
ftp_ar <- data.frame(ftp_ar)

ftp_srv <- read_dta("ftp_srv.dta")
ftp_srv <- data.frame(ftp_srv)

#merge and clean survey and administrative data
ftp_merge <- right_join(ftp_ar, ftp_srv, by = "sampleid")

ftp_merge_clean <- ftp_merge |>
  select(-ends_with(".y")) |>
  rename_with(~ sub("\\.x$", "", .), ends_with(".x"))
```

```r
#create tlyes
ftp_merge_tlyes <- ftp_merge_clean |>
  mutate(tlyes = case_when(
    fmi2 == 1 ~ 1,
    fmi2 == 2 | fmi2 == 8 ~ 0,
    is.na(fmi2) ~ NA_real_,
  )
  ) |>
  filter(!is.na(tlyes))

avg_qemp_post <- ftp_merge_tlyes |>
  summarise(across(starts_with("empq"), mean, na.rm= TRUE)) |>
  select(-empq20)
```

```
## Warning: There was 1 warning in `summarise()`.
## i In argument: `across(starts_with("empq"), mean, na.rm = TRUE)`.
## Caused by warning:
## ! The `...` argument of `across()` is deprecated as of dplyr 1.1.0.
## Supply arguments directly to `.fns` through an anonymous function instead.
##
##   # Previously
##   across(a:b, mean, na.rm = TRUE)
##
##   # Now
##   across(a:b, \(x) mean(x, na.rm = TRUE))
```

```r
avg_qemp_pre <- ftp_merge_tlyes |>
  summarise(across(starts_with("emppq"), mean, na.rm = TRUE))

print(avg_qemp_post)
```

```
##       empq1     empq2     empq3     empq4     empq5     empq6     empq7
## 1 0.3159269 0.3472585 0.3977372 0.4099217 0.3951262 0.4194952 0.4577894
##       empq8     empq9    empq10    empq11    empq12    empq13    empq14
## 1 0.4873803 0.4795474 0.4821584 0.4847694 0.4882507 0.4917319 0.4986945
##      empq15    empq16    empq17    empq18    empq19
## 1 0.5004352 0.4978242 0.5152306 0.5117493 0.5221932
```

```r
print(avg_qemp_pre)
```

```
##   emppq10     emppq9    emppq8    emppq7    emppq6    emppq5    emppq4
## 1       0 0.06875544 0.1627502 0.2375979 0.2497824 0.2489121 0.2419495
##      emppq3    emppq2    emppq1
## 1 0.2489121 0.2619669 0.3028721
```

```r
count_qemp_post <- ftp_merge_tlyes |>
  summarise(across(starts_with("empq"), ~sum(!is.na(.)))) |>
  select(-empq20)

count_qemp_pre <- ftp_merge_tlyes |>
  summarise(across(starts_with("emppq"), ~sum(!is.na(.))))
```

```
print(count_qemp_post)
```

```
##   empq1 empq2 empq3 empq4 empq5 empq6 empq7 empq8 empq9 empq10 empq11 empq12
## 1  1149  1149  1149  1149  1149  1149  1149  1149  1149   1149   1149   1149
##   empq13 empq14 empq15 empq16 empq17 empq18 empq19
## 1   1149   1149   1149   1149   1149   1149   1149
```

```
print(count_qemp_pre)
```

```
##   emppq10 emppq9 emppq8 emppq7 emppq6 emppq5 emppq4 emppq3 emppq2 emppq1
## 1    1149   1149   1149   1149   1149   1149   1149   1149   1149   1149
```

The longest pre-period we could analyze is 10 quarters, as we have observations for every individual in every pre-period quarter.

The longest post-period we could analyze is from the quarter of treatment (q1) to the 18th quarter after treatment (q19).

```
#Question 2
#Reconfigure data
ftp_long <- ftp_merge_tlyes |>
  pivot_longer(
    cols = matches("^(empq|emppq)\\d+$"),
    names_to = c("period", "quarter"),
    names_pattern = "(emp?p?q)(\\d+$)",
    values_to = "employment"
  ) |>
  mutate(
    period = ifelse(period == "empq", "post", "pre"),
    quarter = as.integer(quarter)
  ) |>
  select(sampleid, period, quarter, employment, e, tlyes, everything())

#compare pre and post employment for treatment and control
ftp_pre_post_emp_rate <- ftp_long |>
  group_by(period, tlyes) |>
  summarise(
    emp_rate = mean(employment, na.rm = TRUE)
  )
```

```
## `summarise()` has grouped output by 'period'. You can override using the
## `.groups` argument.
```

```
print(ftp_pre_post_emp_rate)
```

```
## # A tibble: 4 x 3
## # Groups:   period [2]
##   period tlyes emp_rate
##   <chr>  <dbl>    <dbl>
## 1 post       0    0.417
## 2 post       1    0.492
## 3 pre        0    0.190
## 4 pre        1    0.212
```

	(1)
tlyes × post	0.055***
	(0.009)
Num.Obs.	34 153
R2	0.360
R2 Adj.	0.338
R2 Within	0.001
R2 Within Adj.	0.001
AIC	34 413.5
BIC	44 362.6
RMSE	0.39
Std.Errors	Heteroskedasticity-robust
FE: sampleid	X
FE: qtrs_combined	X

+ p <0.1, * p <0.05, ** p <0.01, *** p <0.001

Yes the employment rates were similar, the employment rate for the treatment group before treatment was 21.2% and for the control group it was 19%.

```
#Question 3
#Create continuous quarter variable and post dummy variable
dd_prep <- ftp_long |>
  mutate(
    qtrs_combined = ifelse(period == "post", quarter - 1, quarter * -1),
    post = ifelse(period == "post", 1, 0)
  )


#a
dd_regression <- feols(employment ~ tlyes * post | sampleid + qtrs_combined,
                       data = dd_prep,
                       se = "hetero")
```

```
## NOTE: 317 observations removed because of NA values (LHS: 317).

## The variables 'tlyes' and 'post' have been removed because of collinearity (see $collin.var).
```

```
modelsummary(dd_regression,
             coef_omit = "sampleid|qtrs_combined",
             stars = TRUE
             )
```

```
#b
dd_regression_clustered <- feols(employment ~ tlyes * post | sampleid + qtrs_combined,
                                 data = dd_prep,
                                 cluster = ~sampleid)
```

	(1)
tlyes × post	0.055**
	(0.018)
Num.Obs.	34 153
R2	0.360
R2 Adj.	0.338
R2 Within	0.001
R2 Within Adj.	0.001
AIC	34 413.5
BIC	44 362.6
RMSE	0.39
Std.Errors	by: sampleid
FE: sampleid	X
FE: qtrs_combined	X

+ p <0.1, * p <0.05, ** p <0.01, *** p <0.001

```
## NOTE: 317 observations removed because of NA values (LHS: 317).
## The variables 'tlyes' and 'post' have been removed because of collinearity (see $collin.var).
```

```
modelsummary(dd_regression_clustered,
             coef_omit = "sampleid|qtrs_combined",
             stars = TRUE)
```

The standard errors increase because there is high intra-group correlation in the employment rate (i.e. a person is very likely to be employed this quarter if they were employed last quarter, and the same for being unemployed), which is artificially lowering the non-clustered standard errors.

We should use clustered standard errors because the large change in the size standard errors indicates that the error terms may not be independent without accounting for intra-group correlation.

c) The coefficient can be interpreted as treatment causes a 5.3% increase in the employment rate.

```
#Question 4
pre_trends_prep <- dd_prep |>
  filter(qtrs_combined < 0)


#a
pre_treatment_trends <- feols(employment ~ tlyes | qtrs_combined,
                              data = pre_trends_prep,
                              cluster = ~sampleid)

modelsummary(pre_treatment_trends,
             stars = TRUE)
```

The coefficient for tlyes is not statistically significant, so we fail to reject the null hypothesis that the treatment and control groups had parallel trends in employment prior to treatment.

	(1)
tlyes	0.022
	(0.015)
Num.Obs.	11 490
R2	0.052
R2 Adj.	0.052
R2 Within	0.001
R2 Within Adj.	0.001
AIC	11 054.8
BIC	11 135.6
RMSE	0.39
Std.Errors	by: sampleid
FE: qtrs_combined	X

+ p <0.1, * p <0.05, ** p <0.01, *** p <0.001

```
#b
pre_trends_plot_df <- pre_trends_prep |>
  group_by(tlyes, qtrs_combined) |>
  summarize(employment = mean(employment))
```

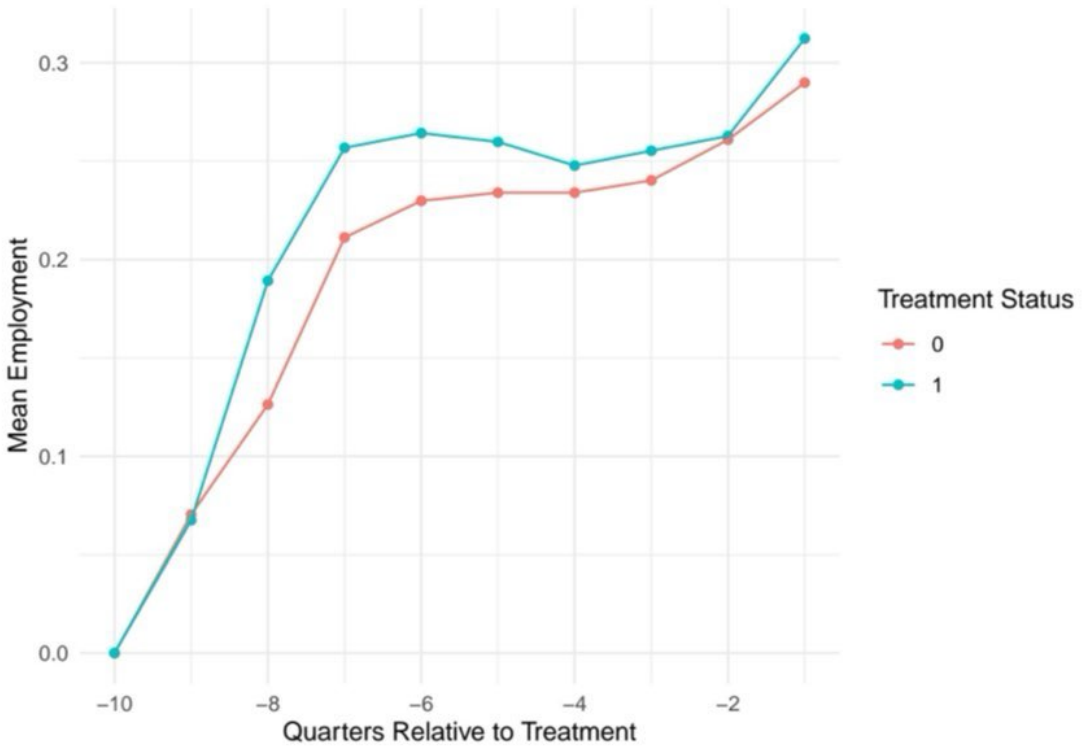
```
## `summarise()` has grouped output by 'tlyes'. You can override using the
## `.groups` argument.
```

```
print(pre_trends_plot_df)
```

```
## # A tibble: 20 x 3
## # Groups:   tlyes [2]
##    tlyes qtrs_combined employment
##    <dbl>         <dbl>      <dbl>
##  1     0           -10     0
##  2     0            -9     0.0704
##  3     0            -8     0.126
##  4     0            -7     0.211
##  5     0            -6     0.230
##  6     0            -5     0.234
##  7     0            -4     0.234
##  8     0            -3     0.240
##  9     0            -2     0.261
## 10     0            -1     0.290
## 11     1           -10     0
## 12     1            -9     0.0676
## 13     1            -8     0.189
## 14     1            -7     0.257
## 15     1            -6     0.264
## 16     1            -5     0.260
## 17     1            -4     0.248
```

```
## 18     1            -3     0.255
## 19     1            -2     0.263
## 20     1            -1     0.312
```

```
ggplot(data = pre_trends_plot_df,
       mapping = aes(x = qtrs_combined, y = employment, color = factor(tlyes), group = tlyes)
       ) +
  geom_point () +
  geom_line() +
  labs(x = "Quarters Relative to Treatment",
       y = "Mean Employment",
       color = "Treatment Status") +
  scale_x_continuous(breaks = seq(min(pre_trends_plot_df$qtrs_combined),
                                  max(pre_trends_plot_df$qtrs_combined),
                                  by = 2)) +
  theme_minimal()
```



c) The pre-trends regression and plot both show that the sample passes the parallel trends test for the pre-treatment data. This means that we cannot reject the parallel trends assumption. However, it does not imply that the parallel trends assumption necessarily holds true for the post-treatment data. Thus, it does not imply that we have accurately estimated the ATT.

```
#Question 5

event_study_prep <- dd_prep |>
```

```
  filter(qtrs_combined > -2)

event_study_model <- feols(employment ~ tlyes * i(qtrs_combined) | sampleid,
                           data = event_study_prep,
                           cluster = ~sampleid)
```

```
## NOTE: 317 observations removed because of NA values (LHS: 317).
```

```
## The variables 'tlyes' and 'tlyes:qtrs_combined::19' have been removed because of collinearity (see $
```

```
modelsummary(event_study_model,
             stars = TRUE,
             coef_omit = "^qtrs_combined::")
```

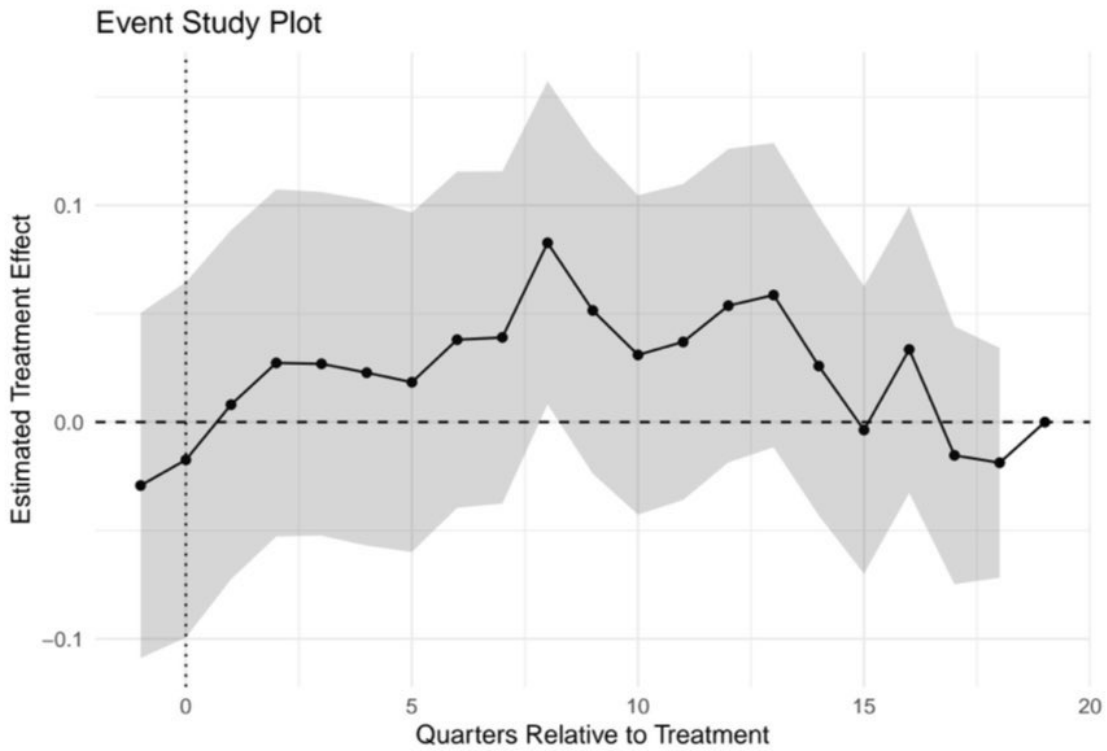
```
rel_times <- unique(event_study_prep$qtrs_combined)
rel_times <- sort(rel_times)

vcov_mat <- vcov(event_study_model)

treatment_effects <- data.frame(
  qtrs_combined = rel_times
) |>
  mutate(
    term = paste0("tlyes:qtrs_combined::", qtrs_combined),
    treatment_effect = ifelse(term %in% names(coef(event_study_model)),
                              coef(event_study_model)[term],
                              0),
    std_error = ifelse(term %in% rownames(vcov_mat),
                       sqrt(diag(vcov_mat)[term]),
                       NA),
    ci_upper = treatment_effect + 1.96 * std_error,
    ci_lower = treatment_effect - 1.96 * std_error
  )


ggplot(treatment_effects, aes(x = qtrs_combined, y = treatment_effect)) +
  geom_point() +
  geom_line() +
  geom_ribbon(aes(ymin = ci_lower, ymax = ci_upper), alpha = 0.2) +
  geom_hline(yintercept = 0, linetype = "dashed") +
  geom_vline(xintercept = 0, linetype = "dotted") +
  labs(
    title = "Event Study Plot",
    x = "Quarters Relative to Treatment",
    y = "Estimated Treatment Effect"
  ) +
  theme_minimal()
```

	(1)
tlyes × qtrs_combined = -1	−0.029
	(0.041)
tlyes × qtrs_combined = 0	−0.017
	(0.042)
tlyes × qtrs_combined = 1	0.008
	(0.041)
tlyes × qtrs_combined = 2	0.027
	(0.041)
tlyes × qtrs_combined = 3	0.027
	(0.040)
tlyes × qtrs_combined = 4	0.023
	(0.041)
tlyes × qtrs_combined = 5	0.018
	(0.040)
tlyes × qtrs_combined = 6	0.038
	(0.040)
tlyes × qtrs_combined = 7	0.039
	(0.039)
tlyes × qtrs_combined = 8	0.083*
	(0.038)
tlyes × qtrs_combined = 9	0.051
	(0.038)
tlyes × qtrs_combined = 10	0.031
	(0.038)
tlyes × qtrs_combined = 11	0.037
	(0.037)
tlyes × qtrs_combined = 12	0.054
	(0.037)
tlyes × qtrs_combined = 13	0.059
	(0.036)
tlyes × qtrs_combined = 14	0.026
	(0.035)
tlyes × qtrs_combined = 15	−0.004
	(0.034)
tlyes × qtrs_combined = 16	0.034
	(0.034)
tlyes × qtrs_combined = 17	−0.015
	(0.030)
tlyes × qtrs_combined = 18	−0.019
	(0.027)
Num.Obs.	23 812
R2	0.397
R2 Adj.	0.366
R2 Within	0.030



Most of the-period specific estimates are not significant. This concerns me because the interaction term was significant for the 2x2 model.

```
#Question 6
mean(treatment_effects$treatment_effect)
```

```
## [1] 0.02237008
```

The mean of the treatment effects from question 5 is less than half the size of the constant post-treatment effect from question 3. To test the hypothesis that they are the same I would calculate the standard error of the mean treatment effect from question 5 and then compare the 95% confidence intervals of the two estimates to see if they overlap.

Question 7 There may be heterogeneity in the treatment effect depending on at what point in calendar time the treatment was administered. If that is the case, then the OLS estimator has undesireable properties as the weighted average of all 2x2 DD estimates, such as negative weights and estimated effects changing from just adding more data.

```
table(dd_prep$rarelqt)
```

```
##
##     2     3     4
## 10980 14010  9480
```