PLSC 30600 - Lab 7 - Differences-in-differences

02/18/2023

Paglayan (2019) - "Public-Sector Unions and the Size of Government"

This lab will have you work through implementing and analyzing a differences-in-differences design with staggered adoption. We'll start with the common two-way fixed effects specification, highlight some of its problems, and illustrate some newer methods that have been developed to address them.

The dataset for this analysis comes from

> Paglayan, Agustina S. "Public-Sector Unions and the Size of Government." American Journal of Political Science 63.1 (2019): 21-36.

The paper looks at the effect of the rollout of mandatory teacher collective bargaining laws among U.S. states on education policy. It focuses on the impact of these laws on four outcomes related to investment in education and education quality: student teacher ratio, teacher salary and per-pupil expenditures. It uses a difference-in-differences design to estimate these treatment effects. However, because mandatory collective bargaining laws were not all implemented at the same time, the study is a staggered-rollout DiD design.

The relevant variables we'll need from the data are:

- `year` - School year
- `State` - State
- `CBrequired_SY` - Is collective bargaining required in this school year (D_{it})
- `YearCBrequired` - In what year is collective bargaining required?
- `studteachratio` - Student teacher ratio
- `lnavgteachsal` - Annual teacher salary (log, 2010 dollars)
- `lnppexpend` - Per-pupil current expenditures (log 2010 dollars)
- `lnnonwageppexpend` - Per-pupil non-wage current expenditures (log, 2010 dollars)

Let's start by loading in the complete dataset

```
union <- read_dta("Paglayan Dataset.dta")
```

The analysis primarily focuses on the period from 1959 to 1997, so let's subset down to those times

```
union <- union %>% filter(year >= 1959 & year <= 1997)
View(union)
```

Visualizing treatment patterns

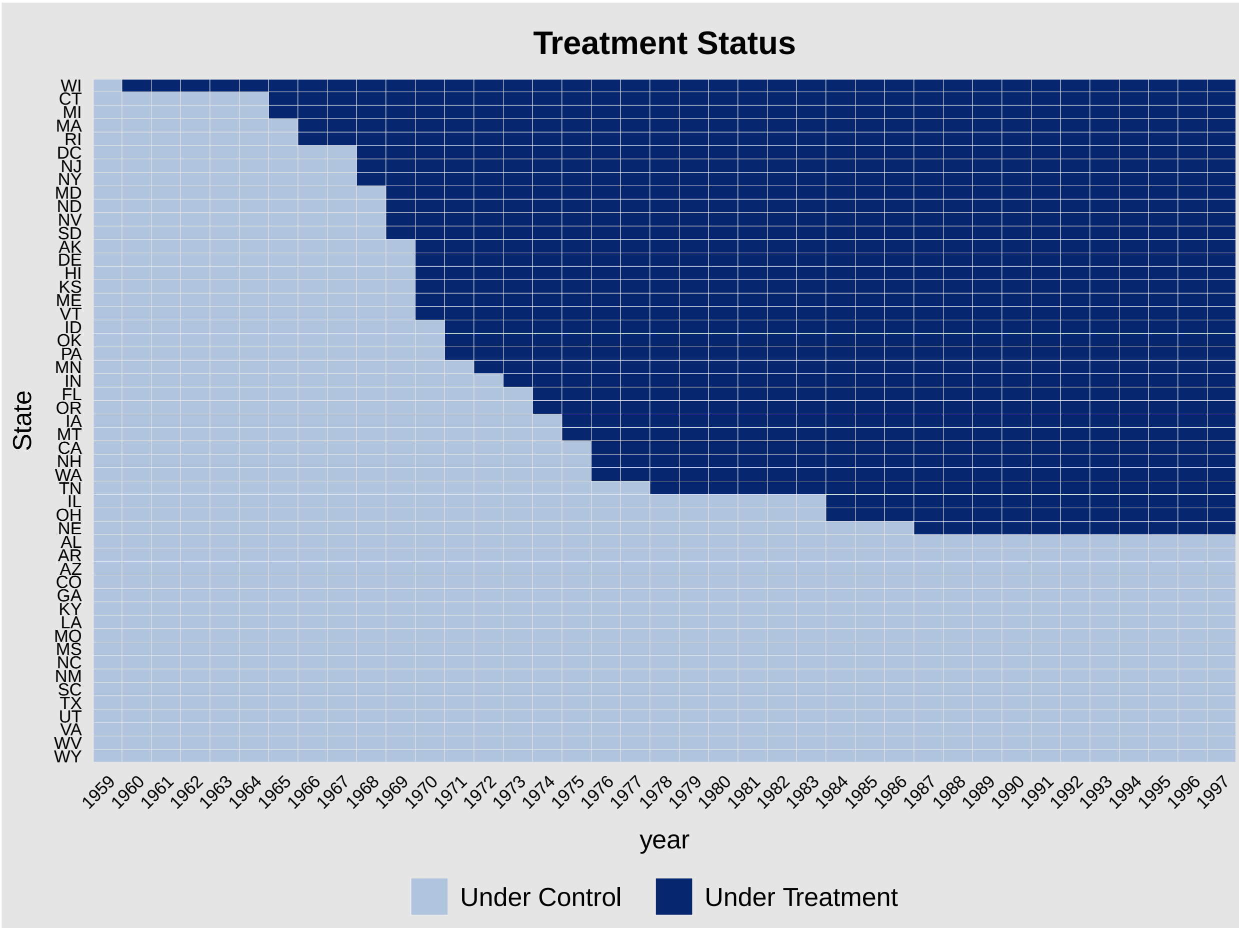
It's important to understand the timing of treatment patterns whenever looking at a staggered roll-out design: how many units ever initiate treatment, how many units are "never-treated", how staggered is the treatment (are there a lot that take treatment at the same time or are there many different timing-groups)?

One very convenient package for doing this visualization is the `panelView` package which contains the `panelview` function. The documentation is quite extensive and supports plots of the outcome among different treatment timings as well as plots of the treatment - definitely check it out for yourself. Here we'll illustrate how to make a simple treatment timing plot

```
# Starting point of collective laws
union %>%
  group_by(State) %>%
  filter(CBrequired_SY == 1) %>%
  select(year, State, CBrequired_SY) %>%
  summarize(Start = min(year)) %>%
  arrange(Start)
```

```
## # A tibble: 34 x 2
##    State Start
##    <chr> <dbl>
##  1 WI     1960
##  2 CT     1965
##  3 MI     1965
##  4 MA     1966
##  5 RI     1966
##  6 DC     1968
##  7 NJ     1968
##  8 NY     1968
##  9 MD     1969
## 10 ND     1969
## # ... with 24 more rows
```

```
library(panelView)

# Our treatment at time t variable (D_{it}) is "CBrequired_SY"
# The syntax for the index argument is to pass the name of the "unit" variable and the name of the "tim
# by.timing sorts the units by treatment initiation
# axis.adjust makes it so we can read the years (tilts them 45 degrees)
panelview(data = union, D = "CBrequired_SY", index = c("State", "year"), axis.adjust=T, by.timing=T)
```



Note that Wisconsin has only one pre-treatment period and ends up being removed from the analysis (since we can't use it for the pre-treatment placebos). DC is also dropped because it's not a state.

```
# Drop WI and DC
union <- union %>% filter(State != "WI"&State!="DC")
```

Two-way fixed effects

The paper looks at four main outcomes:

- `studteachratio` - Student teacher ratio
- `lnavgteachsal` - Annual teacher salary (log, 2010 dollars)
- `lnppexpend` - Per-pupil current expenditures (log 2010 dollars)
- `lnnonwageppexpend` - Per-pupil non-wage current expenditures (log, 2010 dollars)

Let's start by just running two-way fixed effects (with all of its attendant issues and seeing what the estimates are)

```
# Student-teacher ratio
lm_robust(studteachratio ~ CBrequired_SY,
          fixed_effects = ~ as.factor(year) + as.factor(State),
          data=union,
          cluster=State)
```

```
##                Estimate Std. Error t value Pr(>|t|) CI Lower CI Upper   DF
## CBrequired_SY    0.952      0.381     2.5   0.0163    0.184     1.72 43.2
```

```
# Note about Log: Log-level -> A one unit change in X is associated with 100 * Beta % change in Y

# Annual teacher salary (log, 2010 dollars)
lm_robust(lnavgteachsal ~ CBrequired_SY,
          fixed_effects = ~ as.factor(year) + as.factor(State),
          data=union,
          cluster=State)
```

```
##                Estimate Std. Error t value Pr(>|t|) CI Lower CI Upper   DF
## CBrequired_SY  -0.0142     0.0161  -0.886    0.381  -0.0466   0.0182 43.2
```

```
# Log per-pupil expenditures
lm_robust(lnppexpend ~ CBrequired_SY,
          fixed_effects = ~ as.factor(year) + as.factor(State),
          data=union,
          cluster=State)
```

```
##                Estimate Std. Error t value Pr(>|t|) CI Lower CI Upper   DF
## CBrequired_SY  -0.0244     0.0242   -1.01     0.32  -0.0733   0.0245 43.2
```

```
# Log per-pupil non-wage expenditures
lm_robust(lnnonwageppexpend ~ CBrequired_SY,
          fixed_effects = ~ as.factor(year) + as.factor(State),
          data=union,
          cluster=State)
```

```
##                Estimate Std. Error t value Pr(>|t|) CI Lower CI Upper   DF
## CBrequired_SY  -0.0141     0.0266  -0.528      0.6  -0.0678   0.0396 43.2
```

One simple fix that we might want to make is to use an estimator that does not use future periods to de-bias our estimates of each group-time treatment effect. Essentially, we'd like to estimate the effect for each treatment initiation group for every possible time period and aggregate those estimates up to some form of "average". The Callaway and Sant'anna (2021) estimator does just this and is implemented in the `did` package in R.

Let's start with the full dataset, including the never-treated units - we'll focus on the "student-teacher ratio" outcome since that's the one where we get the biggest discrepancy in results

```
library(did)
```

```
## Warning: package 'did' was built under R version 4.2.2
```

```
# We pass in the first period when a unit takes treatment, coded as 0 for the never-treateds
union$YearTreated <- union$YearCBrequired
union$YearTreated[is.na(union$YearTreated)] <- 0

# First estimate each group-time att_gt
# Note: for some reason they want a numeric idname
```

```r
full_attgt <- att_gt(yname = "studteachratio",
                     tname = "year",
                     idname = "Stateid",
                     gname = "YearTreated",
                     data= union,
                     control_group = "notyettreated")
```

```
## Warning in pre_process_did(yname = yname, tname = tname, idname = idname, : Be aware that there are s
##   Check groups: 1965,1966,1968,1969,1971,1972,1973,1974,1975,1976,1978,1984,1987.

## Warning in att_gt(yname = "studteachratio", tname = "year", idname =
## "Stateid", : Not returning pre-test Wald statistic due to singular covariance
## matrix
```

```r
# Definitely read these warnings - it's hard to do inference with small groups!
# There are a lot of groups and a lot of group-time effects - it's kind of pointless to try to do infer
# we can also define the control group as the never treated observations or not yet treated, i.e., cont

# Callaway and Sant'anna give a few different ways - "group" will estimate an average for each group an
# other aggregation options
full_att <- aggte(full_attgt, type = "group")
full_att
```

```
##
## Call:
## aggte(MP = full_attgt, type = "group")
##
## Reference: Callaway, Brantly and Pedro H.C. Sant'Anna.  "Difference-in-Differences with Multiple Time
##
##
## Overall summary of ATT's based on group/cohort aggregation:
##    ATT    Std. Error     [ 95%  Conf. Int.]
##  0.619         0.332    -0.0319        1.27
##
##
## Group Effects:
##  Group Estimate Std. Error [95% Simult.  Conf. Band]
##   1965   0.1553      0.843       -1.870       2.180
##   1966  -0.1660      0.475       -1.307       0.975
##   1968   1.0125      0.771       -0.838       2.863
##   1969   1.5515      0.416        0.553       2.550 *
##   1970   0.4663      0.422       -0.548       1.481
##   1971   0.5103      1.004       -1.900       2.921
##   1972   1.3517      0.330        0.560       2.143 *
##   1973  -0.0701      0.289       -0.763       0.623
##   1974   0.6764      0.631       -0.839       2.192
##   1975   0.5172      0.310       -0.228       1.262
##   1976   1.2371      1.331       -1.959       4.433
##   1978  -0.2553      0.235       -0.820       0.310
##   1984  -0.1170      0.268       -0.761       0.527
##   1987   0.3821      0.235       -0.183       0.947
## ---
```

```
## Signif. codes: '*' confidence band does not cover 0
##
## Control Group:  Not Yet Treated,  Anticipation Periods:  0
## Estimation Method:  Doubly Robust
```

Maybe there's something to some of the early vs. late groups, but overall the estimated effect on student-teacher ratios is slightly positive but statistically insignificant. Though take note the difference with the original two-way FE result that was "significant"

What happens if we just use the "at-some-point treated" units and drop the never-treateds like in the original paper?

```
union_treated <-  union %>% filter(!is.na(YearCBrequired))
# First estimate each group-time att_gt
# Note: for some reason they want a numeric idname
treated_attgt <- att_gt(yname = "studteachratio",
                        tname = "year",
                        idname = "Stateid",
                        gname = "YearTreated",
                        data= union_treated,
                        control_group = "notyettreated")
```

```
## Warning in pre_process_did(yname = yname, tname = tname, idname = idname, : Be aware that there are s
##   Check groups: 1965,1966,1968,1969,1971,1972,1973,1974,1975,1976,1978,1984,1987.
```

```
## Warning in att_gt(yname = "studteachratio", tname = "year", idname =
## "Stateid", : Not returning pre-test Wald statistic due to singular covariance
## matrix
```

```
# Note that we *cannot* fit this with control_group = "nevertreated" because there aren't any never-tre

# Aggregate to get a single "ATT"
treated_att <- aggte(treated_attgt, type = "group")
```

```
## Warning in compute.aggte(MP = MP, type = type, balance_e = balance_e, min_e =
## min_e, : Simultaneous critical value is arguably 'too large' to be realible.
## This usually happens when number of observations per group is small and/or there
## is no much variation in outcomes.
```

```
treated_att
```

```
##
## Call:
## aggte(MP = treated_attgt, type = "group")
##
## Reference: Callaway, Brantly and Pedro H.C. Sant'Anna.  "Difference-in-Differences with Multiple Tim
##
##
## Overall summary of ATT's based on group/cohort aggregation:
##    ATT    Std. Error     [ 95%  Conf. Int.]
##  0.229         0.223     -0.209       0.666
##
```

```
##
## Group Effects:
##  Group Estimate Std. Error [95% Simult.  Conf. Band]
##   1965  -0.2753     0.7483     -2.16e+15    2.16e+15
##   1966  -0.4433     0.3462     -1.00e+15    1.00e+15
##   1968   1.0477     0.6569     -1.90e+15    1.90e+15
##   1969   1.0242     0.3795     -1.10e+15    1.10e+15
##   1970  -0.0774     0.5358     -1.55e+15    1.55e+15
##   1971   0.2355     0.8464     -2.44e+15    2.44e+15
##   1972   1.0872     0.1417     -4.09e+14    4.09e+14
##   1973   0.2038     0.1410     -4.07e+14    4.07e+14
##   1974  -0.0895     0.3616     -1.04e+15    1.04e+15
##   1975  -0.2337     0.1457     -4.21e+14    4.21e+14
##   1976   0.5871     0.8973     -2.59e+15    2.59e+15
##   1978   0.3239     0.1869     -5.40e+14    5.40e+14
##   1984  -0.3213     0.0661     -1.91e+14    1.91e+14
## ---
## Signif. codes: '*' confidence band does not cover 0
##
## Control Group:  Not Yet Treated,  Anticipation Periods:  0
## Estimation Method:  Doubly Robust
```

Again, somewhat similar result to the original paper - I think possibly driven by the change in the set of time periods for which we can estimate the treatment effects (we aren't estimate effects from the 1990s because there are no more "not-yet-treated" groups).

Overall though, be aware that Callaway and Sant'anna (2021) is a bit of a high-variance estimator because of how it selects the differencing groups. Try playing around with some of the new imputation estimators (e.g. Borusyak, Jaravel and Spiess (2021)) and see what you get!

"Event study" plots

Suppose we don't want to estimate the effect overall, but rather the effect for each time period after treatment . Suppose we also want to do some "placebo" tests (compare pre-treatment periods relative to a baseline "never-treated" group)

A very common approach to estimating per-period effects via difference-in-differences is a "dynamic" two-way fixed effects regression specification. We create a set of dummy variables for each unit based on the number of time periods between the year in the row and the year the unit takes treatment. These dummy variables will always be 0 for the never-treateds. We also need to leave one period out as our baseline comparison period

```r
# Make a variable for how many years away a unit is from its treatment time (0 is the time of treatment
union <- union %>% mutate(yearFromCB = year - YearCBrequired)
```

```r
# Let's see what happens with California. 1976 is when collective law is implemented.
california <- union %>%
  filter(State == "CA") %>%
  select(State, year, CBrequired_SY, yearFromCB)

View(california)
```

```r
# How many possible levels (among units that at some point take treatment)
table(union$yearFromCB)
```

```
##
## -28 -27 -26 -25 -24 -23 -22 -21 -20 -19 -18 -17 -16 -15 -14 -13 -12 -11 -10  -9
##   1   1   1   3   3   3   3   3   3   4   4   7   9  11  12  13  16  22  26  28
##  -8  -7  -6  -5  -4  -3  -2  -1   0   1   2   3   4   5   6   7   8   9  10  11
##  28  30  32  32  32  32  32  32  32  32  32  32  32  32  32  32  32  32  32  31
##  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31
##  31  31  29  29  29  29  29  29  28  28  25  23  21  20  19  16  10   6   4   4
##  32
##   2
```

```r
# Make the never-treateds "infinity" (ensure that their dummy will be dropped as well)
union$yearFromCB[is.na(union$yearFromCB)] <- Inf

# Make the dummy variables using the factor syntax - make -1 the reference period
union$yearFromCBFactor <- relevel(as.factor(union$yearFromCB), ref="-1")

# Fit a regression to get the per-period tests
dyn_reg <- lm_robust(studteachratio ~ yearFromCBFactor,
                     fixed_effects = ~ as.factor(year) + as.factor(State),
                     data=union,
                     cluster=State)
dyn_reg
```

```
## 1 coefficient  not defined because the design matrix is rank deficient
```

```
##                      Estimate Std. Error t value Pr(>|t|)  CI Lower CI Upper
## yearFromCBFactor-28  -4.3042      0.705 -6.1066 2.67e-07 -5.726111 -2.88228
## yearFromCBFactor-27  -3.3981      0.685 -4.9642 1.13e-05 -4.778460 -2.01778
## yearFromCBFactor-26  -2.6574      0.635 -4.1861 1.35e-04 -3.937116 -1.37767
## yearFromCBFactor-25  -2.0666      0.672 -3.0755 6.49e-02 -4.392318  0.25903
## yearFromCBFactor-24  -2.1755      0.934 -2.3297 1.15e-01 -5.414749  1.06379
## yearFromCBFactor-23  -2.0277      1.025 -1.9778 1.57e-01 -5.601882  1.54644
## yearFromCBFactor-22  -1.1538      0.861 -1.3395 2.86e-01 -4.164428  1.85684
## yearFromCBFactor-21  -0.2020      0.822 -0.2459 8.24e-01 -3.070740  2.66666
## yearFromCBFactor-20   0.1599      0.698  0.2291 8.36e-01 -2.303409  2.62318
## yearFromCBFactor-19   0.6580      0.826  0.7963 4.71e-01 -1.649023  2.96503
## yearFromCBFactor-18   0.0508      0.639  0.0794 9.41e-01 -1.732774  1.83429
## yearFromCBFactor-17  -1.0177      0.791 -1.2865 2.31e-01 -2.812042  0.77667
## yearFromCBFactor-16  -1.1694      0.659 -1.7754 1.01e-01 -2.602712  0.26401
## yearFromCBFactor-15  -1.0892      0.557 -1.9563 6.89e-02 -2.273977  0.09554
## yearFromCBFactor-14  -0.8513      0.478 -1.7794 9.34e-02 -1.862042  0.15941
## yearFromCBFactor-13  -0.6643      0.462 -1.4365 1.68e-01 -1.635695  0.30701
## yearFromCBFactor-12  -0.5843      0.390 -1.4989 1.48e-01 -1.392637  0.22404
## yearFromCBFactor-11  -0.7280      0.409 -1.7784 8.60e-02 -1.565668  0.10969
## yearFromCBFactor-10  -0.9588      0.437 -2.1923 3.60e-02 -1.850647 -0.06693
## yearFromCBFactor-9   -0.8743      0.378 -2.3111 2.75e-02 -1.645155 -0.10335
## yearFromCBFactor-8   -0.7484      0.342 -2.1874 3.63e-02 -1.445811 -0.05096
## yearFromCBFactor-7   -0.4554      0.294 -1.5462 1.32e-01 -1.055399  0.14467
## yearFromCBFactor-6   -0.3740      0.233 -1.6082 1.18e-01 -0.847680  0.09972
```

```
## yearFromCBFactor-5   -0.3205      0.208 -1.5378 1.34e-01 -0.745047  0.10413
## yearFromCBFactor-4   -0.4782      0.232 -2.0578 4.79e-02 -0.951763 -0.00463
## yearFromCBFactor-3   -0.2677      0.258 -1.0379 3.07e-01 -0.793309  0.25797
## yearFromCBFactor-2   -0.2423      0.149 -1.6228 1.15e-01 -0.546692  0.06210
## yearFromCBFactor0     0.0935      0.119  0.7838 4.39e-01 -0.149651  0.33658
## yearFromCBFactor1     0.2480      0.173  1.4302 1.62e-01 -0.105440  0.60152
## yearFromCBFactor2     0.1789      0.167  1.0707 2.92e-01 -0.161579  0.51931
## yearFromCBFactor3     0.2342      0.183  1.2777 2.11e-01 -0.139297  0.60773
## yearFromCBFactor4     0.4545      0.209  2.1758 3.71e-02  0.028933  0.88009
## yearFromCBFactor5     0.4448      0.255  1.7421 9.11e-02 -0.075195  0.96477
## yearFromCBFactor6     0.5651      0.288  1.9587 5.88e-02 -0.022226  1.15242
## yearFromCBFactor7     0.5444      0.313  1.7389 9.15e-02 -0.092855  1.18162
## yearFromCBFactor8     0.3848      0.338  1.1399 2.63e-01 -0.302103  1.07167
## yearFromCBFactor9     0.6930      0.382  1.8118 7.91e-02 -0.085112  1.47109
## yearFromCBFactor10    0.7022      0.373  1.8843 6.83e-02 -0.055693  1.46014
## yearFromCBFactor11    0.7261      0.388  1.8725 6.99e-02 -0.062367  1.51449
## yearFromCBFactor12    0.7796      0.406  1.9213 6.32e-02 -0.045299  1.60452
## yearFromCBFactor13    0.7431      0.415  1.7910 8.22e-02 -0.100199  1.58648
## yearFromCBFactor14    0.9799      0.448  2.1880 3.57e-02  0.069638  1.89011
## yearFromCBFactor15    0.8959      0.469  1.9122 6.43e-02 -0.056125  1.84801
## yearFromCBFactor16    1.0378      0.484  2.1465 3.90e-02  0.055621  2.02008
## yearFromCBFactor17    0.9793      0.505  1.9392 6.07e-02 -0.046308  2.00488
## yearFromCBFactor18    1.0571      0.517  2.0451 4.84e-02  0.007615  2.10667
## yearFromCBFactor19    1.0099      0.533  1.8942 6.64e-02 -0.072285  2.09216
## yearFromCBFactor20    1.1162      0.540  2.0672 4.62e-02  0.020100  2.21225
## yearFromCBFactor21    1.0420      0.547  1.9066 6.47e-02 -0.067047  2.15108
## yearFromCBFactor22    0.9426      0.580  1.6264 1.13e-01 -0.235315  2.12047
## yearFromCBFactor23    0.9791      0.606  1.6163 1.16e-01 -0.253675  2.21186
## yearFromCBFactor24    0.9956      0.601  1.6574 1.07e-01 -0.229051  2.22020
## yearFromCBFactor25    0.9666      0.610  1.5854 1.23e-01 -0.277345  2.21059
## yearFromCBFactor26    1.0629      0.640  1.6614 1.07e-01 -0.243663  2.36954
## yearFromCBFactor27    1.2215      0.651  1.8775 7.16e-02 -0.115253  2.55827
## yearFromCBFactor28    1.4598      0.686  2.1281 4.99e-02  0.000601  2.91894
## yearFromCBFactor29    1.1636      0.712  1.6350 1.42e-01 -0.490674  2.81789
## yearFromCBFactor30    1.0152      0.742  1.3682 2.39e-01 -0.998214  3.02866
## yearFromCBFactor31    1.0833      0.702  1.5431 1.93e-01 -0.817143  2.98371
## yearFromCBFactor32    1.2953      0.685  1.8916 2.66e-01 -3.999307  6.58982
## yearFromCBFactorInf       NA         NA      NA       NA        NA       NA
##                        DF
## yearFromCBFactor-28 42.51
## yearFromCBFactor-27 43.14
## yearFromCBFactor-26 43.58
## yearFromCBFactor-25  2.62
## yearFromCBFactor-24  2.61
## yearFromCBFactor-23  2.59
## yearFromCBFactor-22  2.58
## yearFromCBFactor-21  2.59
## yearFromCBFactor-20  2.54
## yearFromCBFactor-19  3.95
## yearFromCBFactor-18  3.95
## yearFromCBFactor-17  8.84
## yearFromCBFactor-16 12.13
## yearFromCBFactor-15 15.29
## yearFromCBFactor-14 16.72
```

```
## yearFromCBFactor-13 18.07
## yearFromCBFactor-12 22.05
## yearFromCBFactor-11 28.64
## yearFromCBFactor-10 31.10
## yearFromCBFactor-9  31.63
## yearFromCBFactor-8  31.41
## yearFromCBFactor-7  31.78
## yearFromCBFactor-6  31.95
## yearFromCBFactor-5  31.78
## yearFromCBFactor-4  31.63
## yearFromCBFactor-3  31.51
## yearFromCBFactor-2  31.32
## yearFromCBFactor0   31.31
## yearFromCBFactor1   31.50
## yearFromCBFactor2   31.61
## yearFromCBFactor3   31.73
## yearFromCBFactor4   31.86
## yearFromCBFactor5   32.13
## yearFromCBFactor6   32.47
## yearFromCBFactor7   32.58
## yearFromCBFactor8   32.84
## yearFromCBFactor9   33.07
## yearFromCBFactor10  33.36
## yearFromCBFactor11  33.53
## yearFromCBFactor12  33.68
## yearFromCBFactor13  33.88
## yearFromCBFactor14  33.88
## yearFromCBFactor15  34.09
## yearFromCBFactor16  34.37
## yearFromCBFactor17  34.62
## yearFromCBFactor18  34.87
## yearFromCBFactor19  35.14
## yearFromCBFactor20  35.08
## yearFromCBFactor21  35.43
## yearFromCBFactor22  33.91
## yearFromCBFactor23  32.77
## yearFromCBFactor24  31.32
## yearFromCBFactor25  30.70
## yearFromCBFactor26  30.00
## yearFromCBFactor27  26.24
## yearFromCBFactor28  15.35
## yearFromCBFactor29   7.65
## yearFromCBFactor30   4.25
## yearFromCBFactor31   4.28
## yearFromCBFactor32   1.28
## yearFromCBFactorInf    NA
```

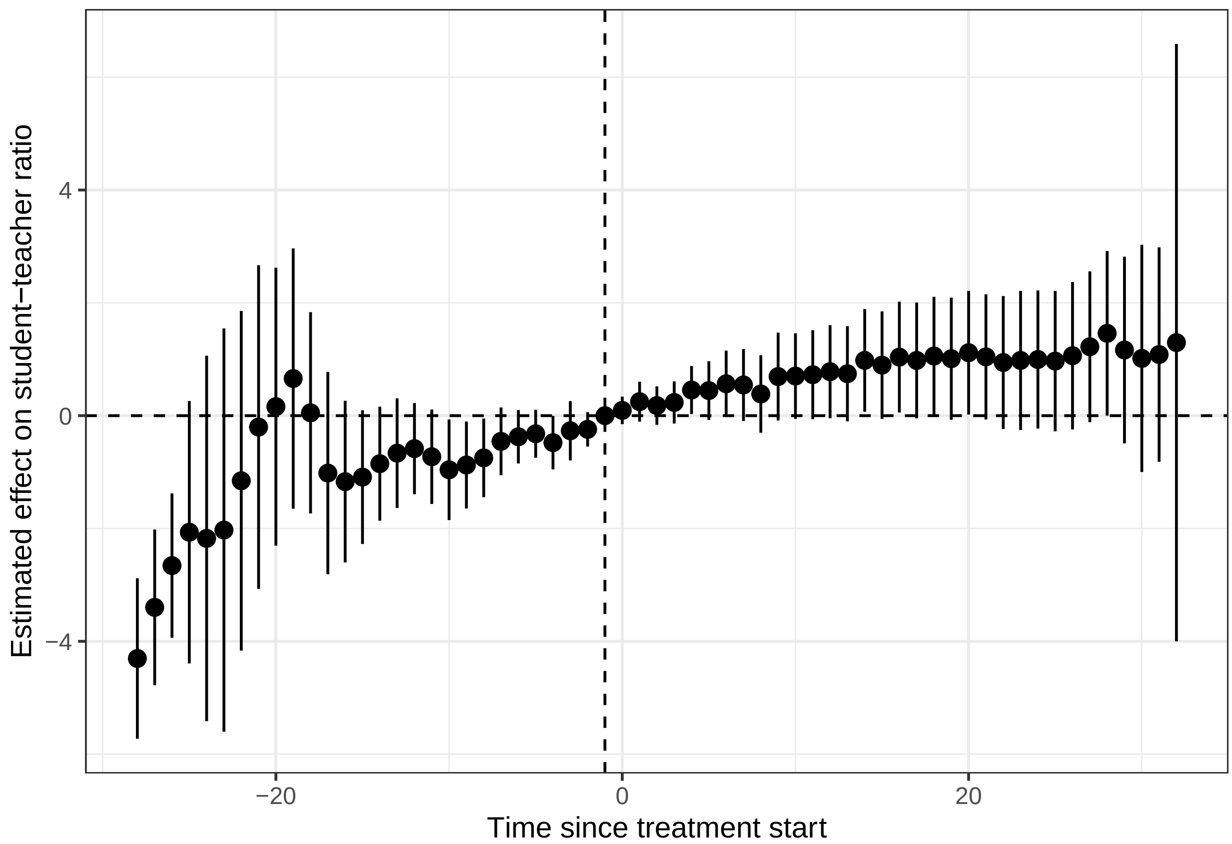
What gets dropped here? First, by default the -1 period isn't included (since R drops the baseline factors). R also is dropping the "Infinity" dummy for our never-treateds (as it should) - this is something of a coding trick. If you were to code up these dummies manually as is commonly done, you would just make a dummy indicator for each of the periods minus your left-out period - you never would make a dummy for your never-treateds to begin with. But we want to make sure R doesn't *drop* the never-treateds entirely as it would if the factor variable were `NA` for them.

Now, let's put this into a plot

```r
dyn_plot <- tidy(dyn_reg) %>%
  filter(!is.na(estimate)) %>%
  mutate(period = as.numeric(str_remove(term, fixed("yearFromCBFactor")))) %>%
    select(period, estimate, conf.low, conf.high) %>%
  bind_rows(data.frame(period = -1, estimate = 0, conf.low=0, conf.high=0))
```

```
## 1 coefficient  not defined because the design matrix is rank deficient
```

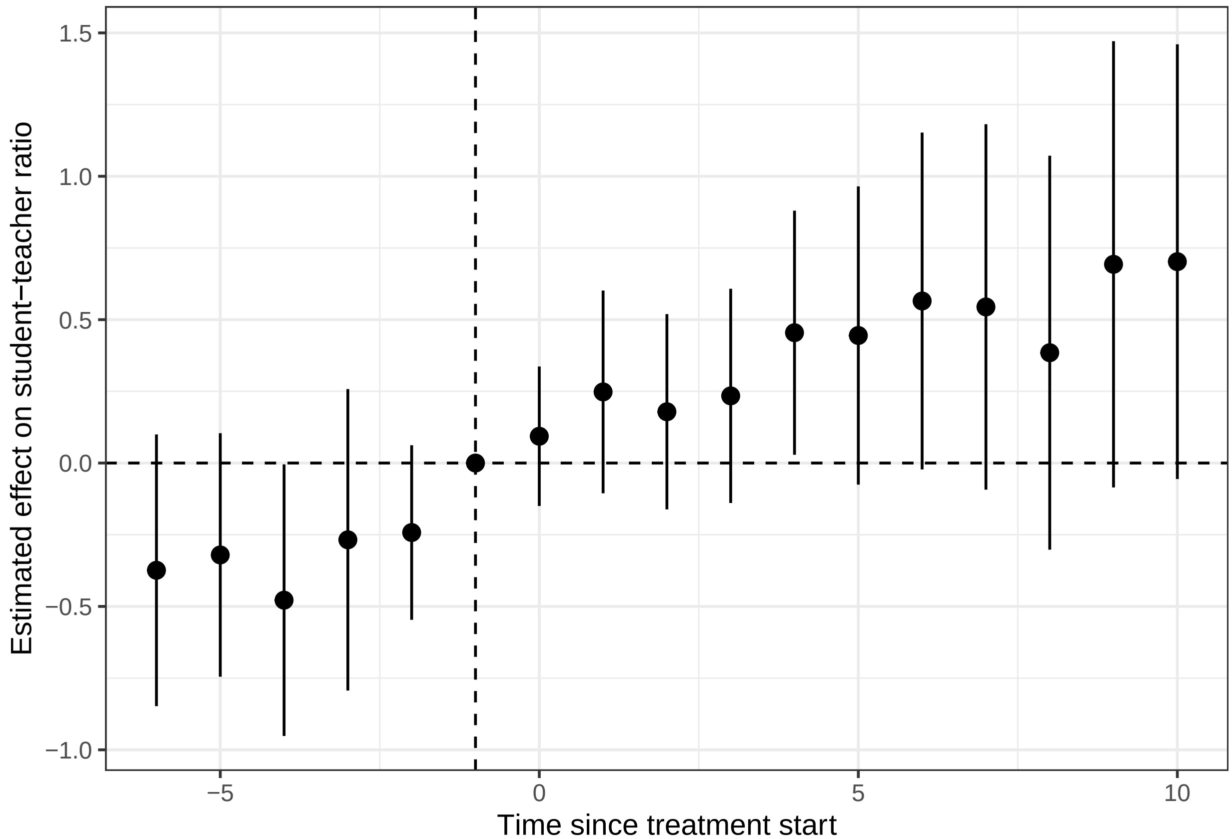
```r
dyn_plot %>%
  ggplot(aes(x=as.numeric(period), y = estimate, ymin=conf.low, ymax=conf.high)) +
  geom_point() + geom_pointrange() +
  xlab("Time since treatment start") +
  ylab("Estimated effect on student-teacher ratio") +
  geom_vline(xintercept = -1, lty=2) +
  geom_hline(yintercept = 0, lty=2) +
  theme_bw()
```



If we don't have staggered timing, this plot is equivalent to just doing a 2x2 DiD for each period relative to the left-out period (you can verify this!). For visualization purposes, we might just focus on a few periods before and after treatment initiation.

```r
dyn_plot %>%
  filter(period >= -6 & period <= 10) %>%
  ggplot(aes(x=as.numeric(period), y = estimate, ymin=conf.low, ymax=conf.high)) +
  geom_point() + geom_pointrange() +
```

```
  xlab("Time since treatment start") +
  ylab("Estimated effect on student-teacher ratio") +
  geom_vline(xintercept = -1, lty=2) +
  geom_hline(yintercept = 0, lty=2) +
  theme_bw()
```



The question in these sorts of plots is always which dummy indicators are "left out" - we need to make sure to leave out at *least* one in order to identify the parameters. Typically this is the first period before treatment initiation (here -1, in lecture 0). But including *every* single dummy is going to lead to some high-variance estimates so its common for researchers to omit other dummy conditions. The original paper omits dummies before 6 but after 10 (though interestingly *includes* the first period before treatment initiation)

An issue arises though from which periods get omitted since they form the baseline comparison group against which our placebos are computed - if we leave out "post-treatment" periods (leads), then our placebo estimates are going to be invalid (since they're not valid placebo periods). See Abraham and Sun (2021) for the full technical breakdown. I'd also recommend

> Baker, Andrew C., David F. Larcker, and Charles CY Wang. "How much should we trust staggered difference-in-differences estimates?." Journal of Financial Economics 144.2 (2022): 370-395.

for a clearer breakdown of the issues along with some simulations that show when things really break down.

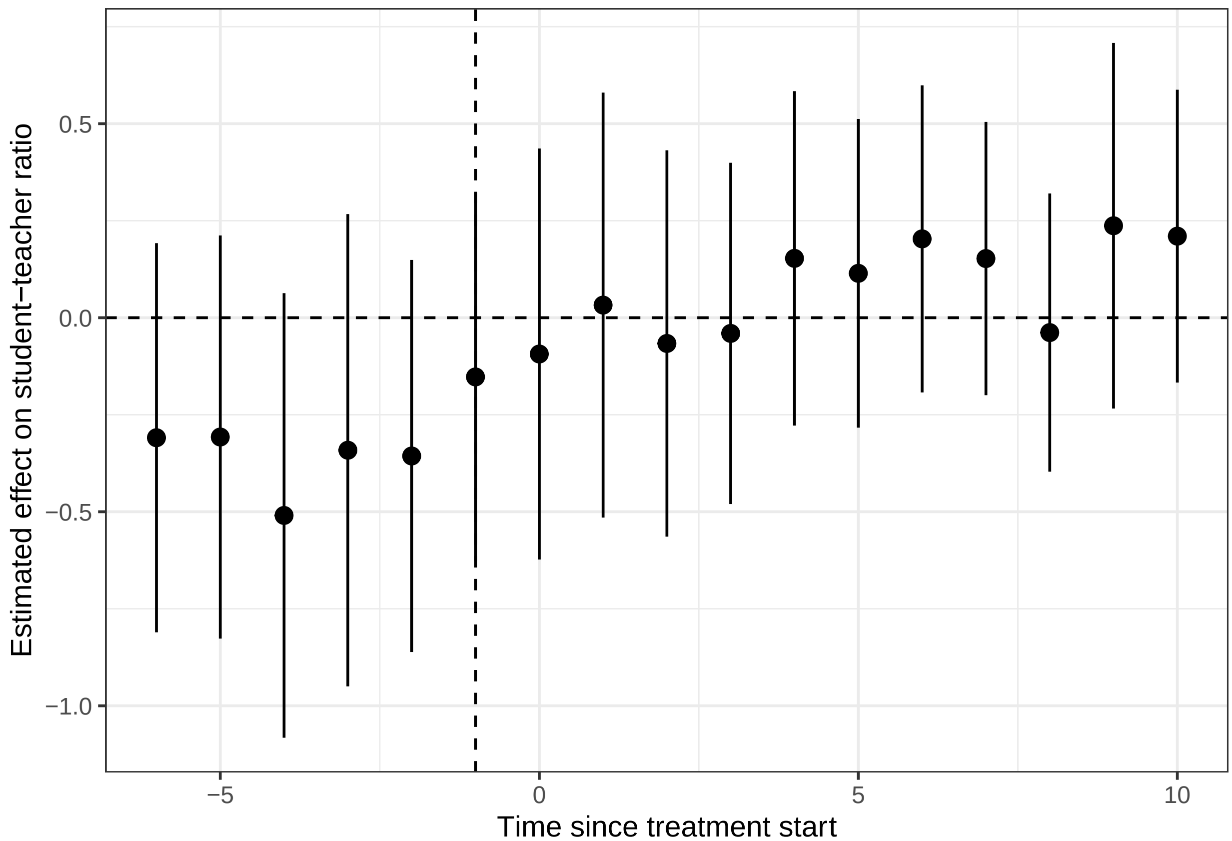
Let's actually see how the plots before would differ from the plots if we did them the way the original paper did them:

```r
# Make the dummy variables manually!
union$`treat-6` <- as.numeric(union$yearFromCBFactor == -6)
union$`treat-5` <- as.numeric(union$yearFromCBFactor == -5)
union$`treat-4` <- as.numeric(union$yearFromCBFactor == -4)
union$`treat-3` <- as.numeric(union$yearFromCBFactor == -3)
union$`treat-2` <- as.numeric(union$yearFromCBFactor == -2)
union$`treat-1` <- as.numeric(union$yearFromCBFactor == -1)
union$`treat0` <- as.numeric(union$yearFromCBFactor == 0)
union$`treat1` <- as.numeric(union$yearFromCBFactor == 1)
union$`treat2` <- as.numeric(union$yearFromCBFactor == 2)
union$`treat3` <- as.numeric(union$yearFromCBFactor == 3)
union$`treat4` <- as.numeric(union$yearFromCBFactor == 4)
union$`treat5` <- as.numeric(union$yearFromCBFactor == 5)
union$`treat6` <- as.numeric(union$yearFromCBFactor == 6)
union$`treat7` <- as.numeric(union$yearFromCBFactor == 7)
union$`treat8` <- as.numeric(union$yearFromCBFactor == 8)
union$`treat9` <- as.numeric(union$yearFromCBFactor == 9)
union$`treat10` <- as.numeric(union$yearFromCBFactor == 10)

# Fit a regression to get the per-period tests
dyn_reg_2 <- lm_robust(studteachratio ~ `treat-6` + `treat-5` + `treat-4` + `treat-3` +
                            `treat-2` + `treat-1` + `treat0` + `treat1` + `treat2` + `treat3` + `treat4
                            `treat5` + `treat6` + `treat7` + `treat8` + `treat9` + `treat10`, fixed_eff
dyn_reg_2
```

```
##             Estimate Std. Error t value Pr(>|t|) CI Lower CI Upper   DF
## ‘treat-6‘  -0.3093      0.246  -1.256   0.2181   -0.811   0.1921 32.3
## ‘treat-5‘  -0.3074      0.255  -1.205   0.2370   -0.827   0.2120 32.4
## ‘treat-4‘  -0.5095      0.281  -1.811   0.0794   -1.082   0.0633 32.5
## ‘treat-3‘  -0.3414      0.299  -1.142   0.2617   -0.950   0.2670 32.6
## ‘treat-2‘  -0.3563      0.248  -1.436   0.1606   -0.862   0.1489 32.7
## ‘treat-1‘  -0.1526      0.233  -0.655   0.5168   -0.626   0.3212 32.8
## treat0     -0.0935      0.260  -0.359   0.7217   -0.623   0.4361 32.8
## treat1      0.0325      0.269   0.121   0.9045   -0.515   0.5801 32.8
## treat2     -0.0663      0.245  -0.271   0.7880   -0.564   0.4315 32.9
## treat3     -0.0404      0.216  -0.187   0.8528   -0.480   0.3993 32.8
## treat4      0.1529      0.212   0.722   0.4753   -0.278   0.5838 32.8
## treat5      0.1144      0.195   0.585   0.5624   -0.283   0.5121 32.8
## treat6      0.2032      0.194   1.045   0.3037   -0.193   0.5989 32.7
## treat7      0.1525      0.173   0.881   0.3845   -0.200   0.5045 32.6
## treat8     -0.0383      0.176  -0.217   0.8292   -0.397   0.3202 32.5
## treat9      0.2370      0.231   1.024   0.3132   -0.234   0.7081 32.4
## treat10     0.2102      0.185   1.134   0.2650   -0.167   0.5874 32.3
```

```r
# Messy data cleaning to make this into a coefficient plot
dyn_plot_2 <- tidy(dyn_reg_2) %>%
  filter(!is.na(estimate)) %>%
  mutate(term = str_remove_all(term, "`")) %>%
  mutate(period = as.numeric(str_remove(term, fixed("treat")))) %>%
  select(period, estimate, conf.low, conf.high)

dyn_plot_2 %>%
  ggplot(aes(x=as.numeric(period), y = estimate, ymin=conf.low, ymax=conf.high)) +
```

```
  geom_point() +
  geom_pointrange() +
  xlab("Time since treatment start") +
  ylab("Estimated effect on student-teacher ratio") +
  geom_vline(xintercept = -1, lty=2) +
  geom_hline(yintercept = 0, lty=2) +
  theme_bw()
```



This version of the plot probably understates the extent to which the parallel trends are violated relative to the plot above since it incorporates "invalid" placebo-comparison periods.

In general would recommend that if you are going to not include dummies for periods that are very far from treatment initiation, you should also just *drop* them from the estimation so that they don't contaminate your placebo tests (or include them in the full regression but just omit them from the plot).

Note that even this plot differs slightly from what is presented in the original paper - again, this is because the original paper drops the never-treated units. Having no "never-treated" units causes particularly troublesome issues with the "dynamic" regression - in order to identify the parameters, researchers need to drop not only the baseline period (typically period just before treatment) but another period to allow for identification of the parameters (see Borusyak, Jaravel and Spiess (2021) for more). In general, such plots get very strange.

Lastly, you can make these plots with the att_gt estimates from `did`

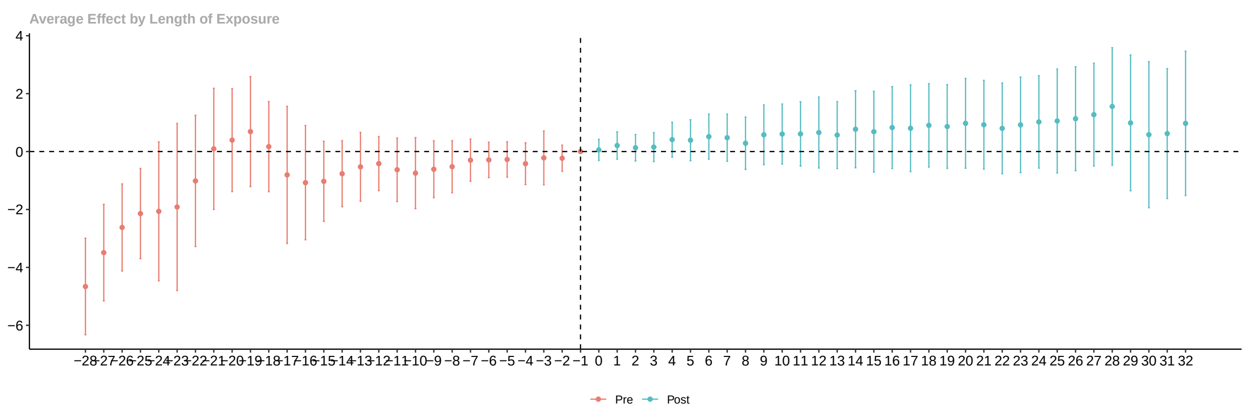
```
# Re-estimate the ATT_GTs but use a universal placebo period - by default it uses a moving period so yo
full_attgt_uni <- att_gt(yname = "studteachratio",
                         tname = "year",
                         idname = "Stateid",
```

```r
                              gname = "YearTreated",
                              data= union, control_group = "notyettreated",
                              base_period = "universal")
```

```
## Warning in pre_process_did(yname = yname, tname = tname, idname = idname, : Be aware that there are s
##   Check groups: 1965,1966,1968,1969,1971,1972,1973,1974,1975,1976,1978,1984,1987.
```

```
## Warning in att_gt(yname = "studteachratio", tname = "year", idname =
## "Stateid", : Not returning pre-test Wald statistic due to singular covariance
## matrix
```

```r
# Get the effects for each level of exposure (time since treatment) from our original att_gt estimates
full_dynamic <- aggte(full_attgt_uni, type = "dynamic")
ggdid(full_dynamic) + geom_vline(xintercept = -1, lty=2)
```



Ultimately, we think there might be evidence that parallel trends are violated to begin with if we include the never-treated units. This could suggest that the original paper is correct that we don't want to use the never-treated units since we don't believe that parallel trends would hold between them and the units that receive treatment at some point. However, this does *substantially* reduce our power - we're stuck with about 32 states!