

Staggered DiD

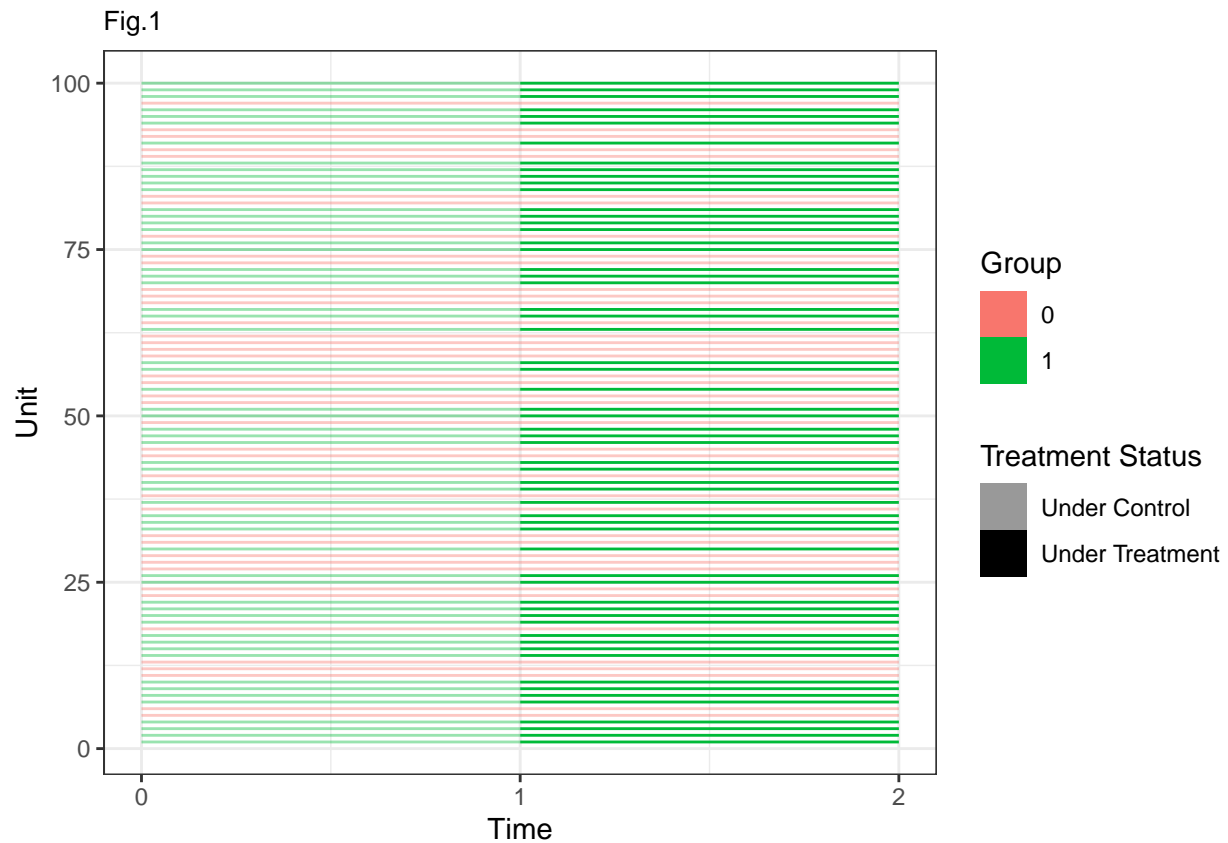
Till Tietz

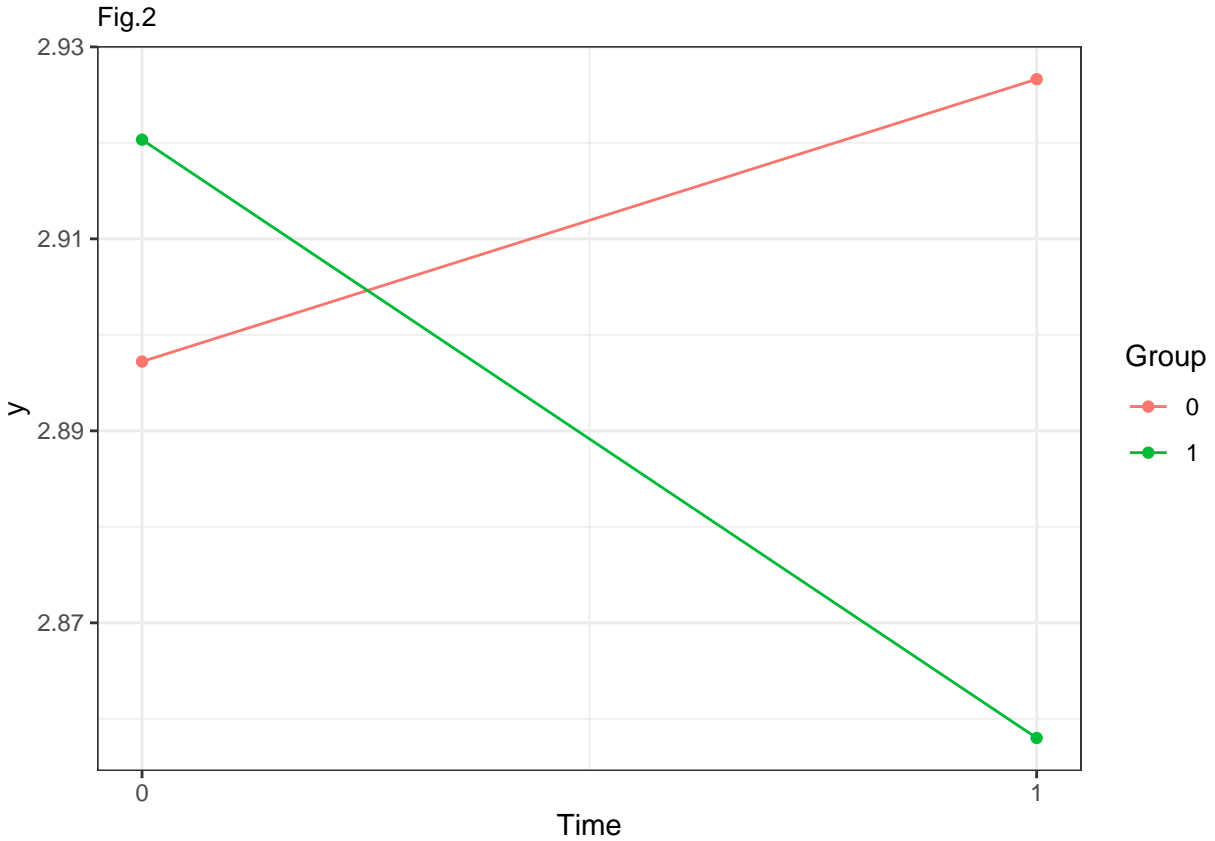
1 2 2022

Canonical DiD 2x2

We'll start with the canonical two period two group i.e. 2x2 DiD design for the purpose of illustration and to get a feel for the behavior of various *ATT* estimators.

```
data <- data.frame(id = rep(1:100, each = 2),  
                  g = rep(rbinom(100, 1, 0.5), each = 2),  
                  t = rep(c(0,1), 100),  
                  y = runif(200,1,5))%>%  
  dplyr::mutate(treat = if_else(t == 1 & g == 1,1,0))
```





Plugin Estimator:

Let \bar{Y}_{gt} be the average outcome for treatment group g in time period t . Given this, we can write a simple plugin estimator for the ATT in a 2x2 DiD as follows:

$$ATT = (\bar{Y}_{11} - \bar{Y}_{01}) - (\bar{Y}_{10} - \bar{Y}_{00})$$

$$ATT = (\bar{Y}_{11} - \bar{Y}_{10}) - (\bar{Y}_{01} - \bar{Y}_{00})$$

```
(mean(data[data$g == 1 & data$t == 1,"y"]) - mean(data[data$g == 0 & data$t == 1,"y"])) -  
  (mean(data[data$g == 1 & data$t == 0,"y"]) - mean(data[data$g == 0 & data$t == 0,"y"]))
```

```
## [1] -0.09174462
```

```
(mean(data[data$g == 1 & data$t == 1,"y"]) - mean(data[data$g == 1 & data$t == 0,"y"])) -  
  (mean(data[data$g == 0 & data$t == 1,"y"]) - mean(data[data$g == 0 & data$t == 0,"y"]))
```

```
## [1] -0.09174462
```

Regression Estimator:

We can also write the above plugin estimators in linear regression form:

$$\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 g_i + \hat{\beta}_2 t_i + \hat{\beta}_3 g_i t_i$$

Where $\hat{\beta}_3 = ATT$ since:

$$\begin{aligned} Y_{11} &= \hat{\beta}_0 + \hat{\beta}_1 + \hat{\beta}_2 + \hat{\beta}_3 \\ Y_{10} &= \hat{\beta}_0 + \hat{\beta}_1 \\ Y_{01} &= \hat{\beta}_0 + \hat{\beta}_2 \\ Y_{00} &= \hat{\beta}_0 \\ ATT &= (\bar{Y}_{11} - \bar{Y}_{01}) - (\bar{Y}_{10} - \bar{Y}_{00}) \\ ATT &= (\hat{\beta}_1 + \hat{\beta}_3) - (\hat{\beta}_1) \\ ATT &= \hat{\beta}_3 \end{aligned}$$

```
lm(y ~ g*t, data = data)[["coefficients"]][4]
```

```
##          g:t
## -0.09174462
```

TWFE Estimator:

Given the above regression form, we can write a more general and succinct estimator by subsuming group and time indicators into group and time fixed effects, and adding the interaction as a variable.

```
# TWFE estimator (feols)
fixest::feols(y ~ treat | id + t, data = data)[["coefficients"]]
```

```
##          treat
## -0.09174462
```

```
# TWFE estimator (demeaned)
data_demeaned <- data%>%
  dplyr::mutate(y = y - ave(y,g) - ave(y,t),
               treat = treat - ave(treat,g) - ave(treat,t))

lm(y ~ treat, data = data_demeaned)[["coefficients"]][2]
```

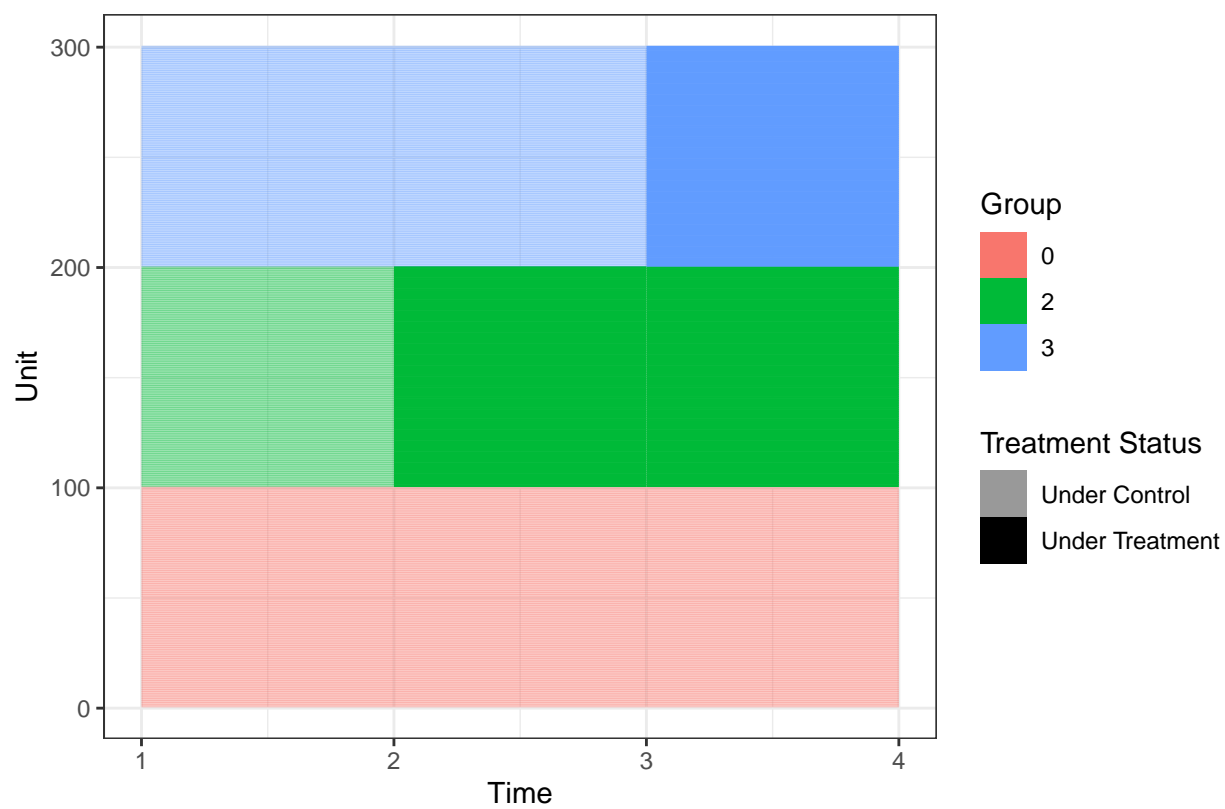
```
##          treat
## -0.09174462
```

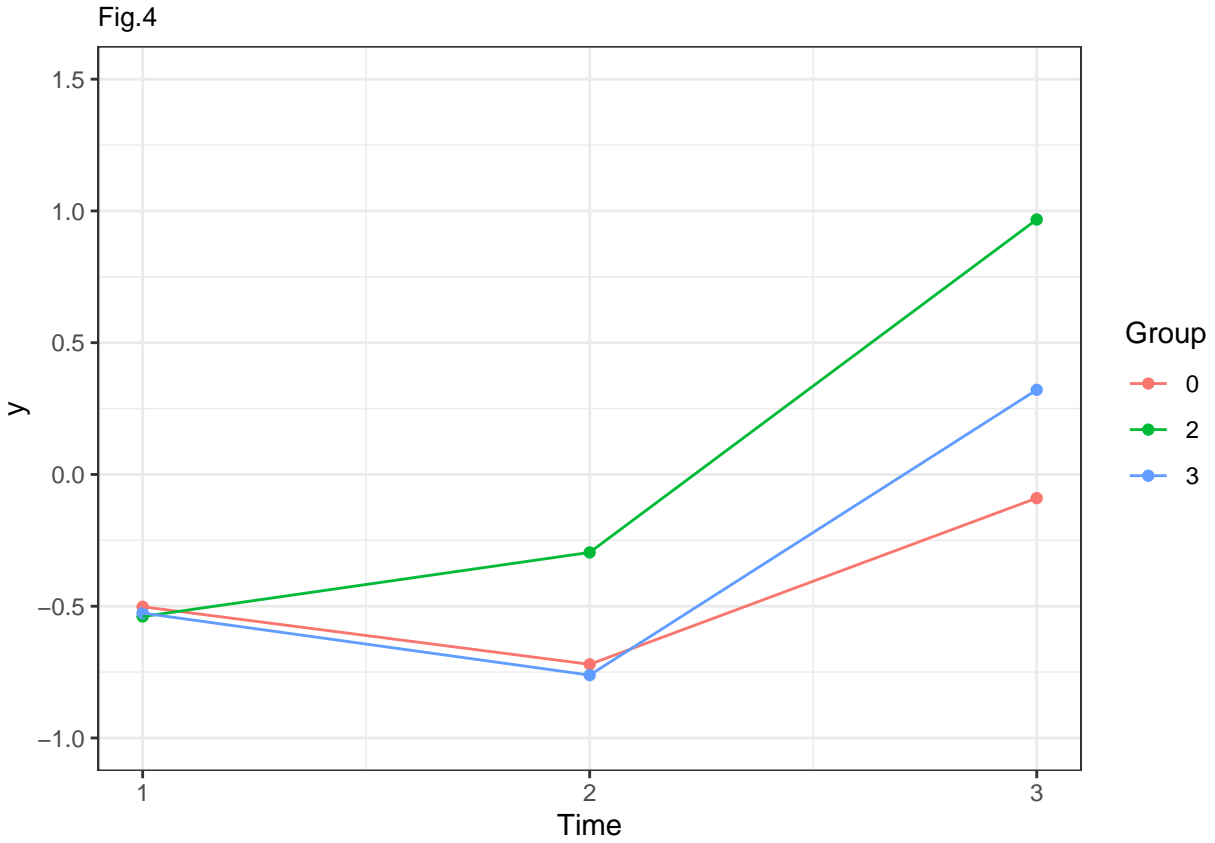
Staggered DiD

We'll now explore the behavior of the TWFE, as well as the CSA and Panel-Match estimator in a staggered DiD context. For illustration purposes we'll consider a 3x3 DiD. That is, our data will comprise three groups: never treated, treated early, treated late; over three time periods.

```
make_data <- function(){  
  
  unit <- tibble(  
    unit = 1:300,  
    unit_fe = rnorm(300, 0, 0.5),  
  
    # generate treatment groups  
    group = case_when(  
      unit %in% 1:100 ~ NA_real_,  
      unit %in% 101:200 ~ 2,  
      unit %in% 201:300 ~ 3  
    ),  
    # avg yearly treatment effects by group  
    hat_gamma = case_when(  
      is.na(group) ~ 0,  
      group == 2 ~ .5,  
      group == 3 ~ .3  
    ) %>%  
    # generate unit specific yearly treatment effects  
    rowwise() %>%  
    mutate(gamma = if_else(is.na(group) == TRUE, 0, rnorm(1, hat_gamma, .2))) %>%  
    ungroup()  
  
    # year fixed effects  
    year <- tibble(  
      year = 1:3,  
      year_fe = rnorm(3, 0, 0.5))  
  
    # full interaction of unit X year  
    crossing(unit, year) %>%  
    # make error term and get treatment indicators and treatment effects  
    mutate(error = rnorm(nrow(.), 0, 0.5),  
      treat = ifelse(year >= group & is.na(group)==F, 1, 0),  
      tau = ifelse(treat == 1 & is.na(group)==F, gamma, 0)) %>%  
    # calculate the dep variable  
    group_by(unit) %>%  
    mutate(cumtau = cumsum(tau)) %>%  
    mutate(y = unit_fe + year_fe + cumtau + error)  
  }  
  
  data <- make_data() %>%  
    dplyr::mutate(group_CSA = if_else(is.na(group), 0, group),  
      group = if_else(is.na(group), 10000, group),  
      time_to_treatment = ifelse(group != 10000, year - group, -1000)) %>%  
    dplyr::mutate(D = ifelse(is.na(time_to_treatment) | time_to_treatment != 0, 0, 1))  
}
```

Fig.3

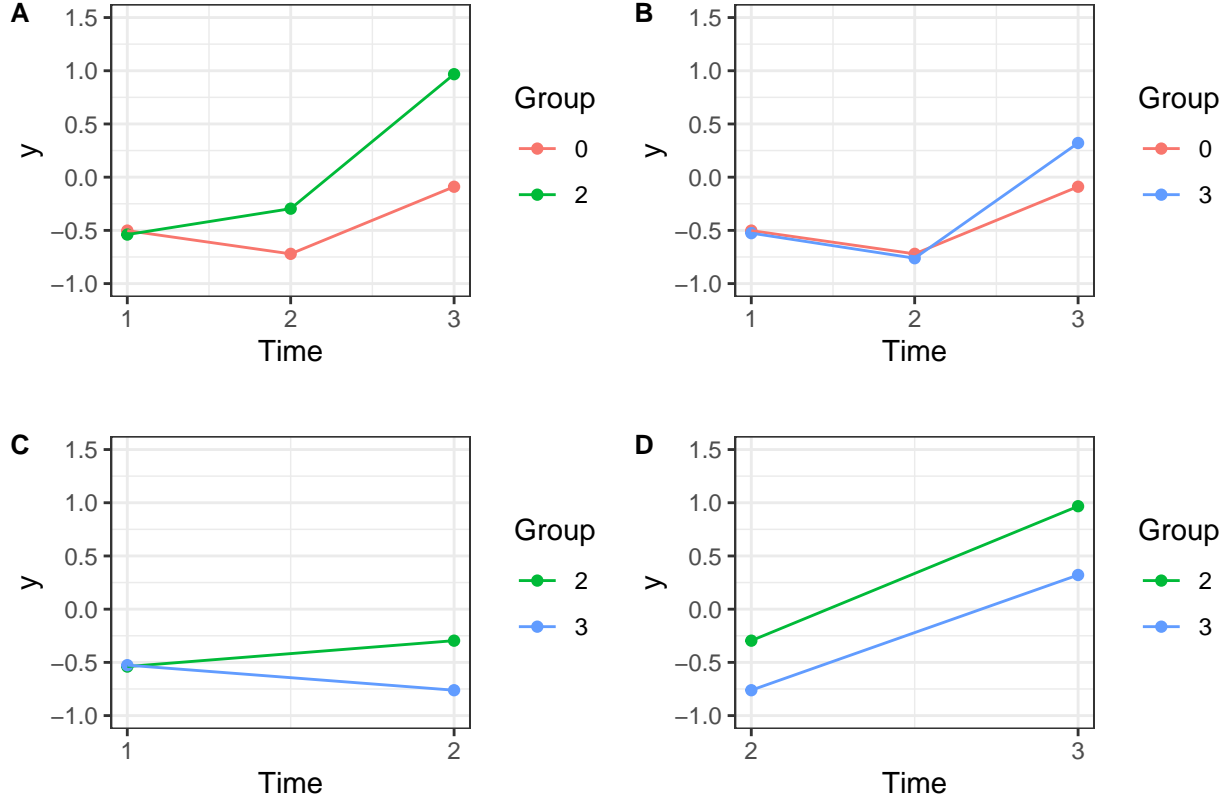




TWFE:

The key intuition behind DiD is to compare groups with constant treatment status over some time period with groups whose treatment status varies over this time period. With this intuition in hand, we can decompose this 3x3 DiD in to four sets of DiDs comparing two groups as follows:

Fig.5



More generally given n treated groups the TWFE estimator decomposes into $n^2 - n$ comparisons. The addition of a never treated group adds a further n decompositions for a total of n^2 .

Notice (with reference to Fig.3) that within each decomposed DiD one group exhibits constant treatment status, while the other changes treatment status. This fact raises three readily apparent issues. Time frame length varies across decompositions (compare A&B vs C&D), groups spend varied lengths of time in the treatment condition and already treated units can serve as a comparison group (see D). Moreover, each decomposition only utilizes a subset of the data for estimation.

Given the above, the TWFE estimator is simply a weighted average of ATTs from all possible decompositions, where the weights are determined by sample proportions (subsample relative to full sample & treatment conditions within subsamples) and variance of treatment within each decomposition.

Given the above, we can derive a plugin version of the TWFE estimator as follows:

Let \bar{Y}_t^g be the average outcome for treatment group g in time period t .

We estimate 4 ATTs:

$$\begin{aligned}
 ATT_1 &= (\bar{Y}_{year>1}^2 - \bar{Y}_{year\leq 1}^2) - (\bar{Y}_{year>1}^0 - \bar{Y}_{year\leq 1}^0) \\
 ATT_2 &= (\bar{Y}_{year>2}^3 - \bar{Y}_{year\leq 2}^3) - (\bar{Y}_{year>2}^0 - \bar{Y}_{year\leq 2}^0) \\
 ATT_3 &= (\bar{Y}_{1<year<2}^2 - \bar{Y}_{year\leq 1}^2) - (\bar{Y}_{1<year<2}^3 - \bar{Y}_{year\leq 1}^3) \\
 ATT_4 &= (\bar{Y}_{year>2}^3 - \bar{Y}_{1<year\leq 2}^3) - (\bar{Y}_{year>1}^2 - \bar{Y}_{1<year\leq 2}^2)
 \end{aligned}$$

Let $n_{g_i g_j}$ be the relative size of treatment group g_i to comparison group g_j $n_{g_i g_j} = \frac{n_{g_i}}{(n_{g_i} + n_{g_j})}$ and D_g be the amount of time group g is treated.

Given the above, we can derive the following weights for the estimated ATTs:

$$\begin{aligned}
W_1 &= (n_2 + n_0)^2 n_{20} (1 - n_{20}) D_2 (1 - D_2) \\
W_2 &= (n_3 + n_0)^2 n_{30} (1 - n_{30}) D_3 (1 - D_3) \\
W_3 &= ((n_2 + n_3)(1 - D_3))^2 n_{23} (1 - n_{23}) \left(\frac{D_2 - D_3}{1 - D_3}\right) \left(\frac{1 - D_2}{1 - D_3}\right) \\
W_4 &= ((n_2 + n_3)(D_2))^2 n_{32} (1 - n_{32}) \left(\frac{D_3}{D_2}\right) \left(\frac{D_2 - D_3}{D_2}\right)
\end{aligned}$$

Where the first portion of each weight corresponds to the relative size of treatment and control subsets, the second portion corresponds to the relative size of the subsample and the third portion corresponds to the variance of treatment status scaled by the size of the respective time window.

It follows that:

$$TWFE = \frac{\sum_{i=1}^n W_i (ATT_i)}{\sum_{i=1}^n W_i}$$

where n is the number of decompositions.

```
att <- function(data, year, group, year_pre, year_post, group_t, group_c, y){
  return(
    (mean(data[data[[group]] == group_t & data[[year]] %in% year_post,y[[1]]) -
     mean(data[data[[group]] == group_t & data[[year]] %in% year_pre,y[[1]])) -
    (mean(data[data[[group]] == group_c & data[[year]] %in% year_post,y[[1]]) -
     mean(data[data[[group]] == group_c & data[[year]] %in% year_pre,y[[1]]))
  )
}

params <- list(
  year_pre = list(1,c(1,2),1,2),
  year_post = list(c(2,3),3,2,3),
  group_t = c(2,3,2,3),
  group_c = c(0,0,3,2)
)

atts <- rep(NA,4)

for(i in 1:length(params)){
  atts[i] <- att(data = data, year = "year", group = "group_CSA",
    year_pre = params[["year_pre"]][[i]],
    year_post = params[["year_post"]][[i]],
    group_t = params[["group_t"]][i],
    group_c = params[["group_c"]][i],
    y = "y")
}

weight <- function(data, year, group, group_t, group_c, treat){
  n <- table(data[[group]])/max(data[[year]])
  n_i <- n[as.character(group_t)]
  n_j <- n[as.character(group_c)]
  n_ij <- n_i / (n_i + n_j)
```



```

if(group_c == 0){

  D_i <- mean(data[data[[group]] == group_t, treat][[1]])
  v_ij <- n_ij * (1 - n_ij) * D_i * (1 - D_i)
  weight <- (n_i + n_j)^2 * v_ij

} else if(group_t < group_c){

  D_i <- mean(data[data[[group]] == group_t, treat][[1]])
  D_j <- mean(data[data[[group]] == group_c, treat][[1]])
  v_ij <- n_ij * (1 - n_ij) * (D_i - D_j) / (1 - D_j) * (1 - D_i) / (1 - D_j)
  weight <- ((n_i + n_j) * (1 - D_j))^2 * v_ij

} else if(group_t > group_c){

  D_i <- mean(data[data[[group]] == group_t, treat][[1]])
  D_j <- mean(data[data[[group]] == group_c, treat][[1]])
  v_ij <- n_ij * (1 - n_ij) * (D_i / D_j) * (D_j - D_i) / (D_j)
  weight <- ((n_j + n_i) * D_j)^2 * v_ij

}
return(weight)
}

weights <- mapply(weight, group_t = c(2,3,2,3), group_c = c(0,0,3,2),
  MoreArgs = list(data = data, year = "year", treat = "treat",
    group = "group_CSA"))

weights <- unname(weights)/ sum(weights)
weights

## [1] 0.3333333 0.3333333 0.1666667 0.1666667

twfe_plug <- sum(weights * atts)/sum(weights)
twfe_plug

## [1] 0.4569091

twfe_reg <- unname(fixest::feols(y ~ treat | unit + year, data = data)[["coefficients"]])
twfe_reg

## [1] 0.4569091

round(twfe_plug,7) == round(twfe_reg,7)

## [1] TRUE

```

Evidently the plugin and regression estimators are equivalent.

From our derivation of the plugin estimator, we have seen that the weights on the decomposed DiD estimates

depend crucially on the size and variance in treatment condition of the subsample used for estimation. ATTs estimated from large subsamples with units treated closer towards the middle of the panel, therefore factor into the TWFE estimate more heavily. Additionally, the fact that already treated units can serve as a comparison group (see Fig.5) may potentially bias ATT estimates in decompositions, particularly when ATTs in periods prior to the estimation period were relatively large. These properties are theoretically undesirable and potentially severely bias the TWFE estimator. At the very least, it is evident that the TWFE does not map cleanly onto the ATT.

We will now take a look at two methods that attempt to address some of the aforementioned issues with the TWFE estimator.

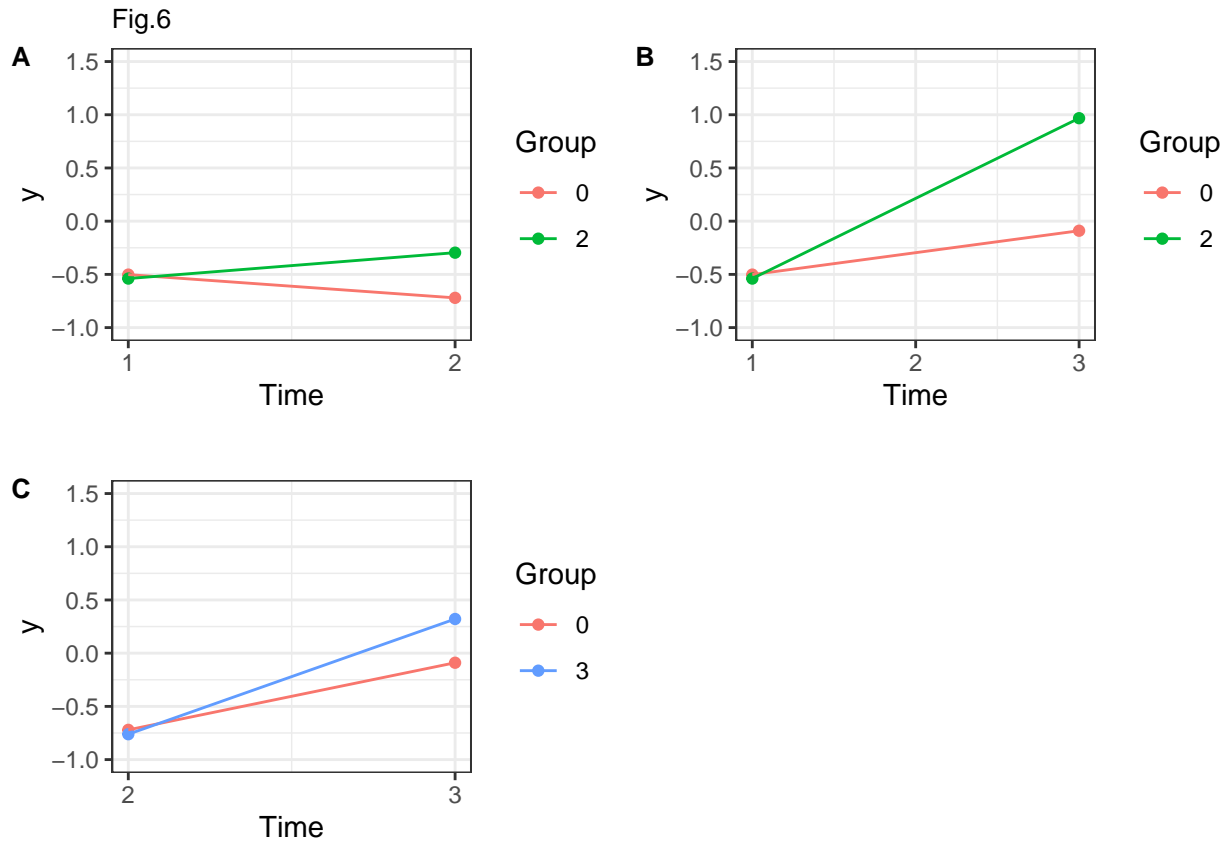
CSA:

Callaway and Sant’Anna (2018) address the shortcomings of the TWFE estimator by decomposing the multi-period DiD ATT into “group-time average treatment effects”. That is a set of average treatment effects for groups g at times t where groups comprise units that were first treated at the same time (e.g. unit i treated at t is in group t). Note that once a unit is treated it remains treated. In the most fundamental case of the CSA methodology these “group-time ATEs” are simply 2x2 DiD estimates with the time period prior to treatment of group g serving as a baseline and either never treated or not-yet-treated units serving as a comparison.

For illustration:

Let $Y_t(0)$ and $Y_t(1)$ be potential outcomes under treatment and control respectively and D_t be a binary treatment. Observed outcomes are realized as $Y_t = D_t Y_t(1) + (1 - D_t) Y_t(0)$. The unobserved parameter of interest is thus $ATT(g, t) = E[Y_t(1) - Y_t(0) | G = g]$. $Y_t(1)$ is observed for group g at $t \geq g$. $Y_t(0)$ can be imputed for g under parallel trends.

CSA thus decomposes our 3x3 DiD into 3 post treatment 2x2 DiDs or “group-time ATEs”.



We'll consider the most elementary CSA set up (never treated comparison group, no additional covariates) for illustration.

```
att_1 <- att(data = data, year = "year", year_pre = 1, year_post = 2,
             group_t = 2, group_c = 0, group = "group_CSA", y = "y")

att_2 <- att(data = data, year = "year", year_pre = 1, year_post = 3,
             group_t = 2, group_c = 0, group = "group_CSA", y = "y")

att_3 <- att(data = data, year = "year", year_pre = 2, year_post = 3,
             group_t = 3, group_c = 0, group = "group_CSA", y = "y")

atts <- c(att_1, att_2, att_3)
atts
```

```
## [1] 0.4618412 1.0946382 0.4519770
```

```
csa_est <- att_gt(ynname = 'y',
                  tname = 'year',
                  idname = 'unit',
                  gname = 'group_CSA',
                  est_method = 'reg',
                  control_group = 'nevertreated',
                  base_period = "universal",
                  data = data)
```

```
csa_est%>%
  broom::tidy()%>%
  .[c(2,3,6),"estimate"]
```

```
## [1] 0.4618412 1.0946382 0.4519770
```

The plugin estimator and the `att_gt` function evidently yield the same results.

The estimated “group-time ATEs” can be aggregated to summary measures as desired. They can be averaged within groups across time, within time across groups or by length of treatment exposure and weighted to account for variations in group size and treatment length (see referenced paper for details).

Finally CSA further proposes estimating “group-time ATEs” using propensity score weighted observations.

$$ATT(g, t) = E \left[\left(\frac{G_g}{E[G_g]} - \frac{\frac{p_g(X)C}{1-p_g(X)}}{E\left[\frac{p_g(X)C}{1-p_g(X)}\right]} \right) (Y_t - Y_{g-1} - m_{g,t}^{nev}(X)) \right]$$

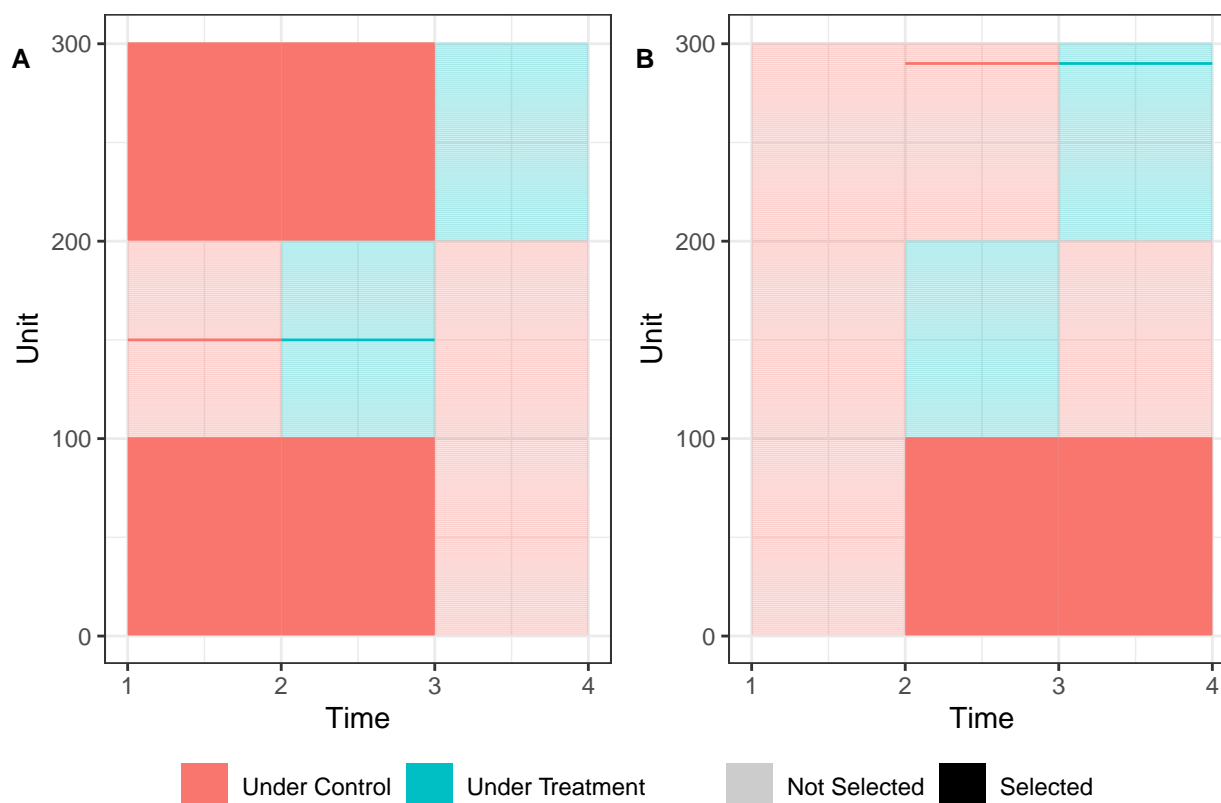
Where C is an indicator for control units, $m_{g,t}^{nev}(X) = E[Y_t - Y_{g-1} | X, C = 1]$ and $p_g(X)$ is the propensity score, i.e. essentially the probability of a unit belonging to group g . The intuition is to give more weight to comparison group units that are similar to units in the treatment group on some set of covariates.

Panel-Match:

Imai and Kim (2019) address the shortcomings of the TWFE estimator by decomposing the multi-period DiD ATT into sets of 2x2 DiD estimations on matched observations. Fundamentally each observation i that changes treatment status at period t is matched with a set of observations with constant treatment status over $t - 1$ and t . For each matched set the ATT is calculated via simple 2x2 DiD.

In our 3x3 DiD case units in groups 2 and 3 are matched as follows:

Fig.7



This plugin estimator will yield approximately equivalent results to the PanelMatch estimator for simple cases. PanelMatch utilizes a weighted TWFE estimator where units are weighted according to the size of the group time sub-set they fall into within the matched set of observation i . Reproducing this method in plugin estimator form would defeat the purpose of generating a simple, illustrative estimator.

```
att <- function(data, year, group, year_pre, year_post, group_t, group_c, y){
  return(
    (data[data[[group]] == group_t & data[[year]] %in% year_post,y][[1]] -
     data[data[[group]] == group_t & data[[year]] %in% year_pre,y][[1]]) -
    (mean(data[data[[group]] == group_c & data[[year]] %in% year_post,y][[1]] -
     data[data[[group]] == group_c & data[[year]] %in% year_pre,y][[1]]))
  )
}

att_1 <- att(data = data, year = "year", group = "group_CSA",
  year_pre = 1, year_post = 2, group_t = 2, group_c = c(0,3),
  y = "y")

att_2 <- att(data = data, year = "year", group = "group_CSA",
  year_pre = 2, year_post = 3, group_t = 3, group_c = 0,
  y = "y")

mean(c(att_1,att_2))

## [1] 0.4600179
```

```
pm <- PanelMatch(lag = 1, time.id = "year", unit.id = "unit",
                 treatment = "D", refinement.method = "none",
                 data = as.data.frame(data), match.missing = TRUE,
                 qoi = "att", outcome.var = "y",
                 lead = 0, forbid.treatment.reversal = FALSE)

unnname(PanelEstimate(sets = pm, data = as.data.frame(data))["estimates"])
```

```
## [1] 0.4611564
```

I can add the weighted TWFE estimator that can be derived from this if necessary.

References, Reading & Resources:

TWFE:

https://cdn.vanderbilt.edu/vu-my/wp-content/uploads/sites/2318/2019/07/29170757/ddtiming_7_29_2019.pdf

CSA:

<https://arxiv.org/pdf/1803.09015.pdf>
<https://bcallaway11.github.io/did/articles/did-basics.html>
<https://github.com/bcallaway11/did/>
https://www.stata.com/meeting/us21/slides/US21__SantAnna.pdf

Panel Match:

<http://web.mit.edu/insong/www/pdf/FEmatch-twoway.pdf>
<https://github.com/insongkim/PanelMatch>
https://cran.r-project.org/web/packages/PanelMatch/vignettes/using_panelmatch.html