Homework Four

Andie Creel

2023-03-29

Diff-in-diff

10 time periods. 1000 ids. treatment turns on in period 5 (post). standard treatment and control. Be specific about what SE are used.

```
myData_og <- vroom("https://raw.githubusercontent.com/paulgp/applied-methods-phd/main/homework/dind_data
```

```
## Rows: 10000 Columns: 8
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (6): ids, time_id, y_instant, y_instant2, y_dynamic, y_dynamic2
## lgl (2): treated_group, post
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
myData_og <- myData_og %>%
  mutate(time_id = as.factor(time_id)) %>%
  mutate(ids = as.factor(ids)) %>%
  mutate(treated_group = treated_group*1) %>%
  mutate(post = post*1)
```

a) Estimate three regressions.

```
myData <- myData_og %>%
  mutate(time_id = relevel(time_id, ref = "1"))


reg_1 <- feols(y_instant ~ treated_group * post, data = myData, vcov = "hetero")
reg_2 <- feols(y_instant ~  treated_group + treated_group*post | time_id, data = myData, vcov = "hetero
```

```
## The variable ’post’ has been removed because of collinearity (see $collin.var).
```

```
reg_3 <- feols(y_instant ~  treated_group*post | time_id + ids, data = myData, vcov = "hetero")
```

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

```
etable(reg_1, reg_2, reg_3)
```

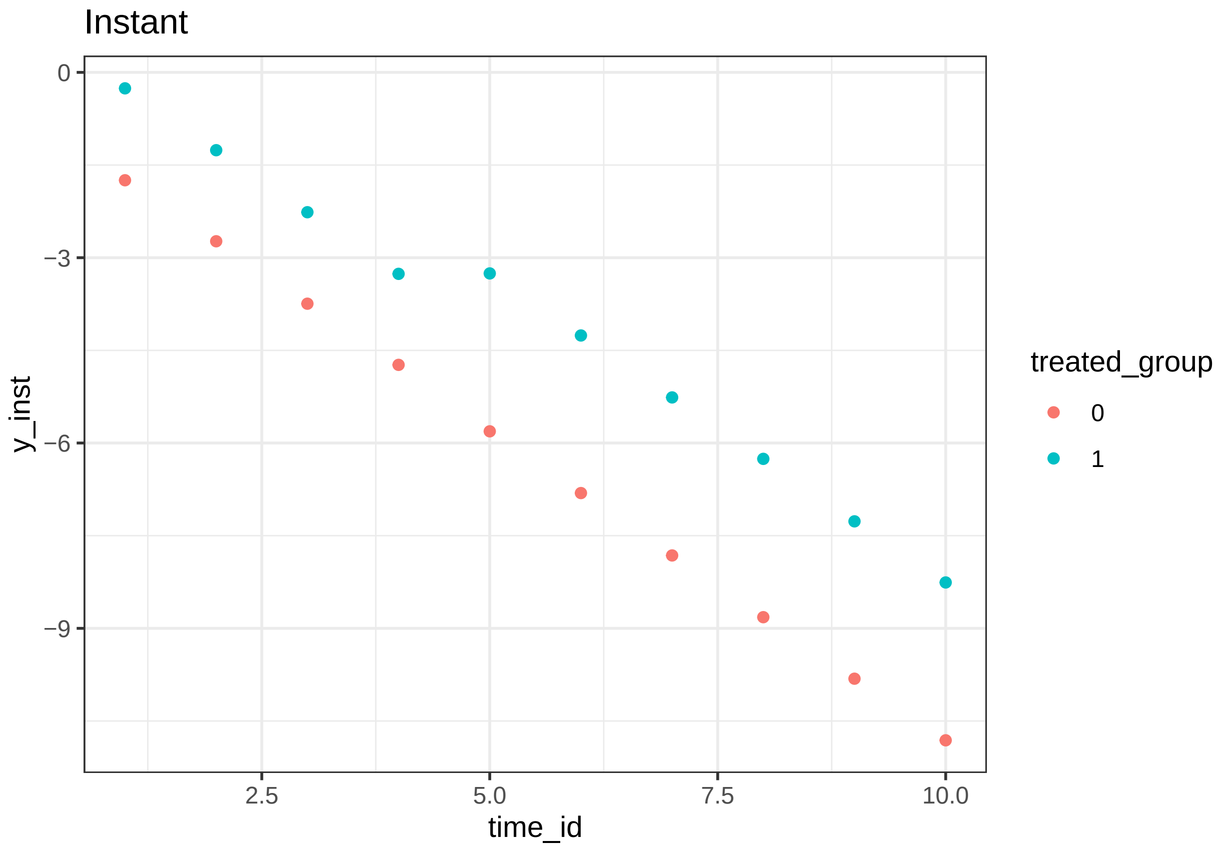
```
##                                   reg_1             reg_2             reg_3
## Dependent Var.:               y_instant         y_instant         y_instant
##
## Constant             -3.240*** (0.0355)
## treated_group         1.479*** (0.0512) 1.479*** (0.0371)
## post                 -5.075*** (0.0519)
## treated_group x post  1.076*** (0.0746) 1.076*** (0.0486) 1.076*** (0.0326)
## Fixed-Effects:       ------------------ ----------------- -----------------
## time_id                              No               Yes               Yes
## ids                                  No                No               Yes
## ____________________ __________________ _________________ _________________
## S.E. type            Heteroskedas.-rob. Heteroskeda.-rob. Heteroskeda.-rob.
## Observations                     10,000            10,000            10,000
## R2                              0.62593           0.85419           0.95022
## Within R2                            --           0.45406           0.12377
## ---
## Signif. codes: 0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

First, I’m using heteroskedastically robust standard errors. The point estimates are the same across regressions. The standard error decrease across regressions because more controls are added by using additional fixed effects.

Note that time period one is being omitted so that we have a reference level.

```
myPlot <- myData %>%
  group_by(time_id, treated_group) %>%
  mutate(y_inst = mean(y_instant)) %>%
  select(time_id, treated_group, y_inst) %>%
  distinct() %>%
  mutate(time_id = as.numeric(time_id)) %>%
  mutate(treated_group = as.factor(treated_group))

ggplot(myPlot, aes(x = time_id, y = y_inst, color = treated_group)) +
  geom_point() +
  theme_bw() +
  ggtitle("Instant")
```



There is a steady decrease in Y across treatment and control. However, in the time period of treatment the treatment group didn't decrease. The decreasing trend continues after treatment but the gap is larger.

b)

```
# regression
myData <- myData_og %>%
  mutate(time_id = relevel(time_id, ref = "4"))

reg_4 <- feols(y_instant ~  treated_group*time_id | time_id + ids, data = myData, vcov = "hetero")
```

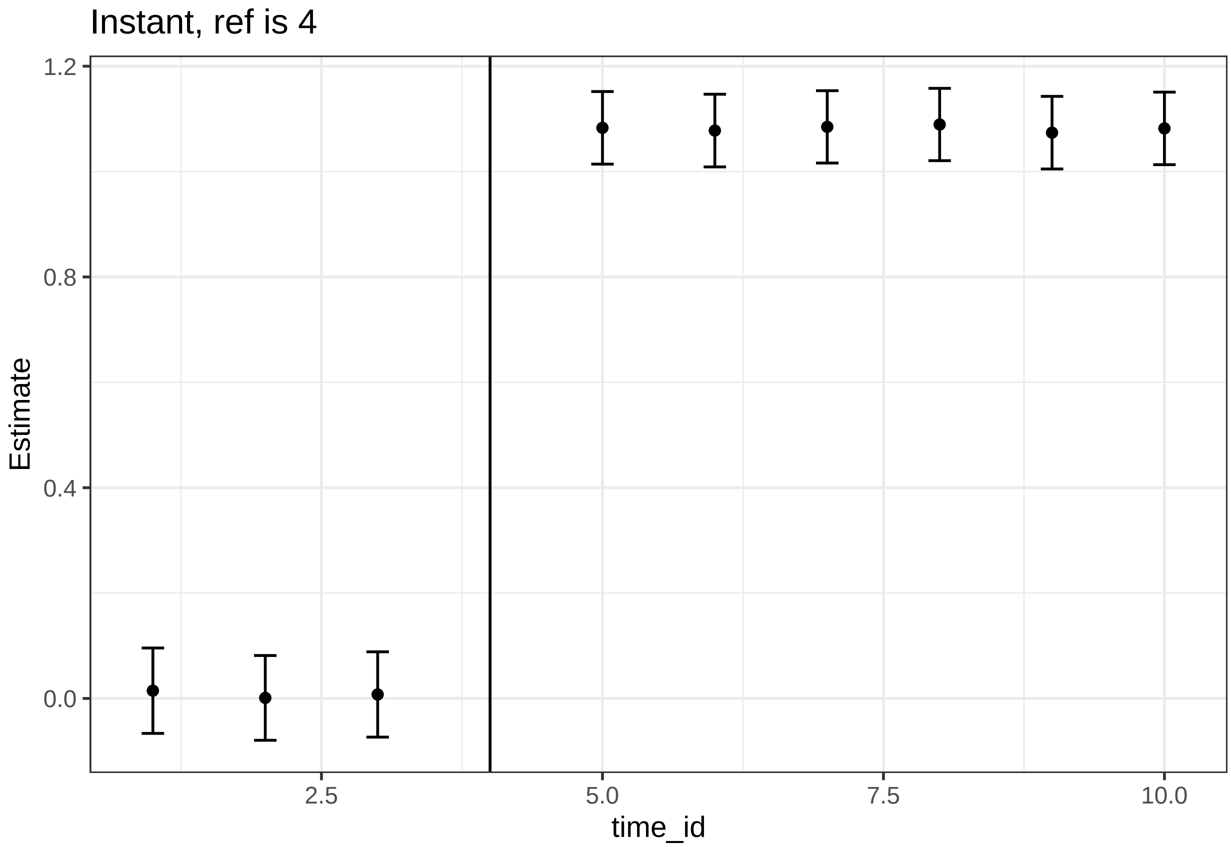
```
## The variables 'treated_group', 'time_id1' and eight others have been removed because of collinearity
```

```
etable(reg_4)
```

```
##                                        reg_4
## Dependent Var.:                    y_instant
##
## treated_group x time_id1    0.0147 (0.0810)
## treated_group x time_id2    0.0010 (0.0805)
## treated_group x time_id3    0.0075 (0.0810)
## treated_group x time_id5  1.083*** (0.0689)
## treated_group x time_id6  1.078*** (0.0690)
## treated_group x time_id7  1.085*** (0.0687)
## treated_group x time_id8  1.089*** (0.0687)
## treated_group x time_id9  1.074*** (0.0689)
## treated_group x time_id10 1.082*** (0.0688)
## Fixed-Effects:            -----------------
## time_id                                 Yes
## ids                                     Yes
## _________________________ _________________
## S.E. type                 Heteroskeda.-rob.
## Observations                         10,000
## R2                                  0.95022
## Within R2                           0.12379
## ---
## Signif. codes: 0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```r
# plot
estDF_4<- as.data.frame(reg_4$coeftable) %>%
  mutate(time_id = c(1,2, 3,  5, 6, 7, 8, 9, 10)) %>%
  rename(se = `Std. Error`)

ggplot(estDF_4, aes(x = time_id, y = Estimate)) +
  geom_point() +
  geom_errorbar(aes(ymin=Estimate-se, ymax=Estimate+se), width=.2,
                position=position_dodge(.9)) +
  geom_vline(xintercept = 4) +
  theme_bw() +
  ggtitle("Instant, ref is 4")
```



Point est in T = 6: 1.0778141 Stand. error in T = 6: 0.0689599

If we do not omit a level (such as period 4) then we do not have a comparison level and interpreting our treatment effect becomes non-sensible.

c)

$$E[y(1) - y(0)]$$

```r
y_0 <- estDF_4 %>%
  filter(time_id <= 4)

y_1 <- estDF_4 %>%
  filter(time_id > 4)


mean(y_1$Estimate) - mean(y_0$Estimate)
```

```
## [1] 1.074033
```

They're the same.

d)

```
# regression
myData <- myData_og %>%
  mutate(time_id = relevel(time_id, ref = "3"))

reg_5 <- feols(y_instant ~  treated_group*time_id | time_id + ids, data = myData, vcov = "hetero")
```

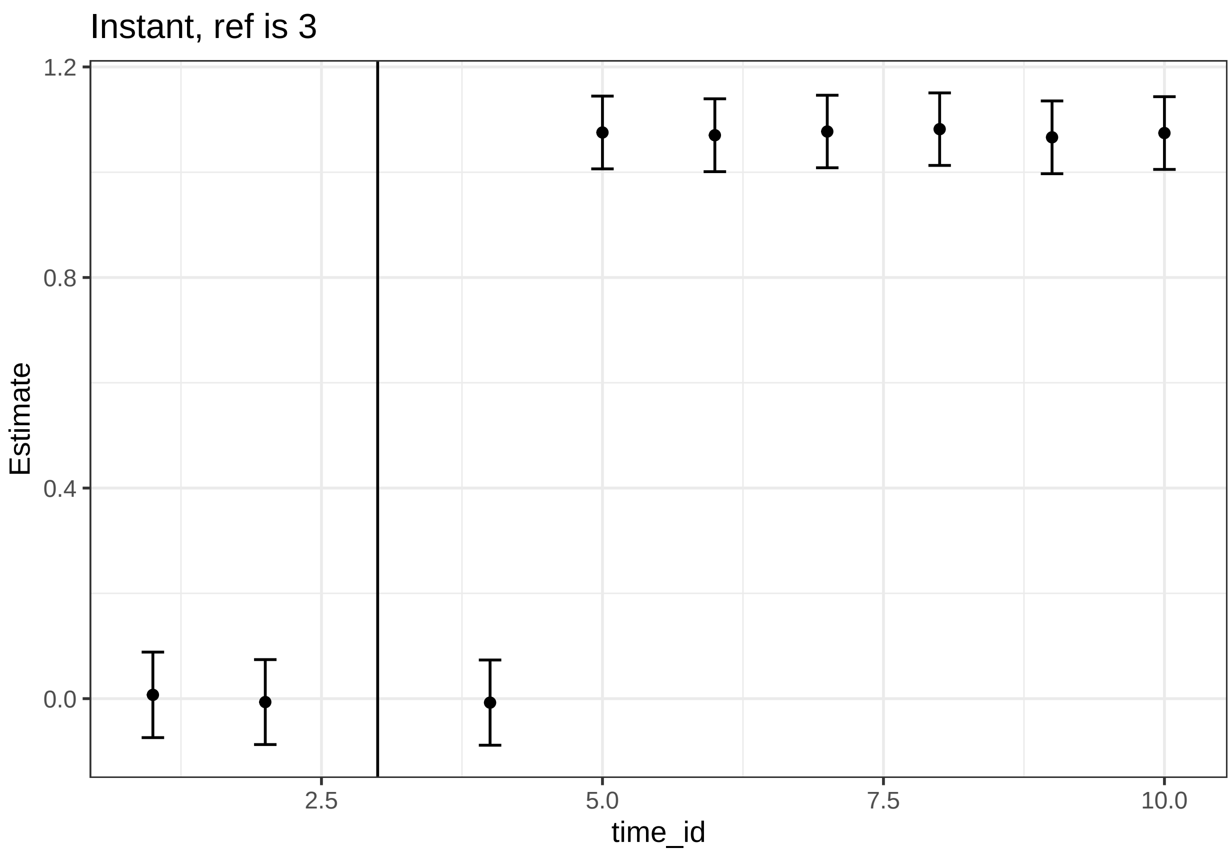
```
## The variables 'treated_group', 'time_id1' and eight others have been removed because of collinearity
```

```
etable(reg_5)
```

```
##                                          reg_5
## Dependent Var.:                      y_instant
##
## treated_group x time_id1    0.0072 (0.0812)
## treated_group x time_id2   -0.0065 (0.0807)
## treated_group x time_id4   -0.0075 (0.0810)
## treated_group x time_id5  1.075*** (0.0691)
## treated_group x time_id6  1.070*** (0.0692)
## treated_group x time_id7  1.077*** (0.0689)
## treated_group x time_id8  1.082*** (0.0689)
## treated_group x time_id9  1.066*** (0.0691)
## treated_group x time_id10 1.074*** (0.0691)
## Fixed-Effects:            -----------------
## time_id                                 Yes
## ids                                     Yes
## _________________________ _________________
## S.E. type                 Heteroskeda.-rob.
## Observations                         10,000
## R2                                  0.95022
## Within R2                           0.12379
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# plot
estDF_5<- as.data.frame(reg_5$coeftable) %>%
  mutate(time_id = c(1,2, 4,  5, 6, 7, 8, 9, 10)) %>%
  rename(se = `Std. Error`)

ggplot(estDF_5, aes(x = time_id, y = Estimate)) +
  geom_point() +
  geom_errorbar(aes(ymin=Estimate-se, ymax=Estimate+se), width=.2,
                position=position_dodge(.9)) +
  geom_vline(xintercept = 3) +
  theme_bw() +
  ggtitle("Instant, ref is 3")
```



e)

```
myData <- myData_og %>%
  mutate(time_id = relevel(time_id, ref = "1"))

reg_1.e<- feols(y_dynamic ~ treated_group * post, data = myData, vcov = "hetero")
reg_2.e<- feols(y_dynamic ~  treated_group + treated_group*post | time_id, data = myData, vcov = "heter
```

```
## The variable 'post' has been removed because of collinearity (see $collin.var).
```

```
reg_3.e<- feols(y_dynamic ~  treated_group*post | time_id + ids, data = myData, vcov = "hetero")
```

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

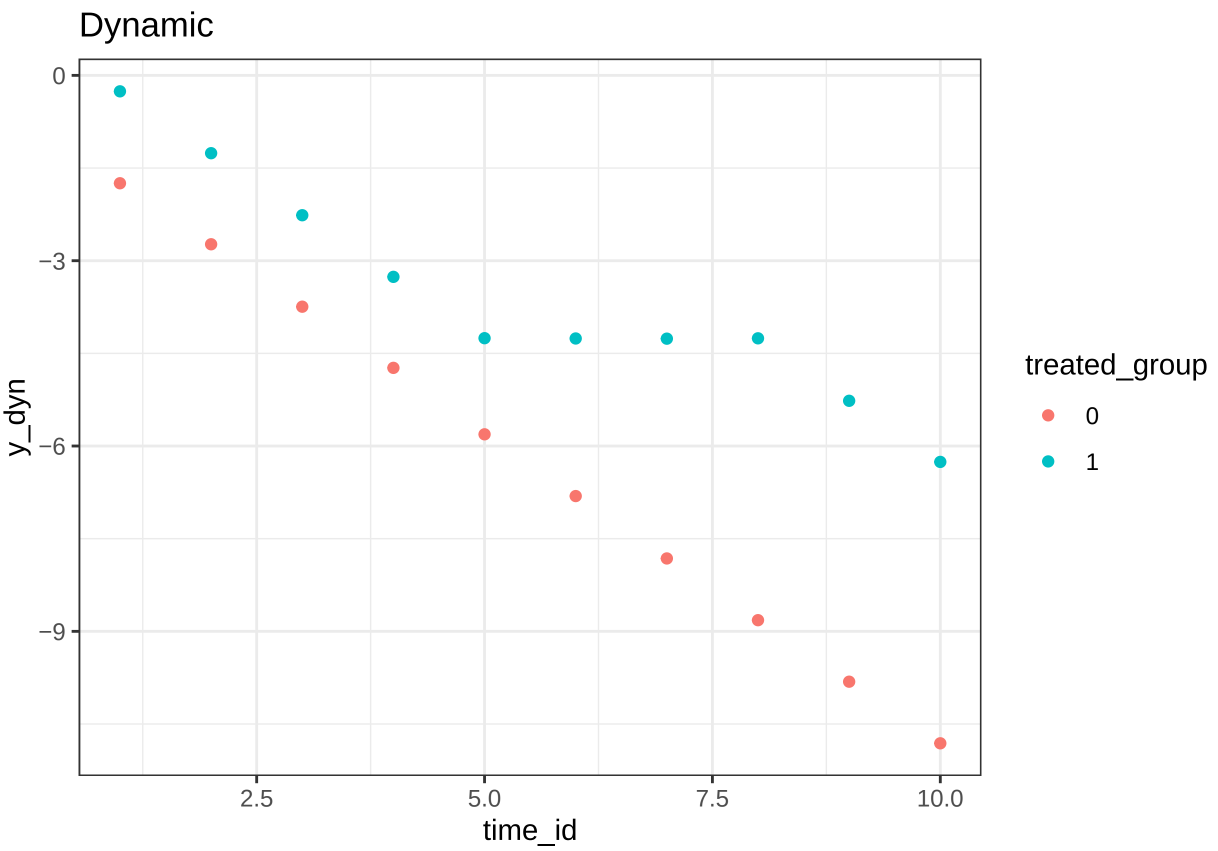
```
etable(reg_1.e, reg_2.e, reg_3.e)
```

```
##                                 reg_1.e          reg_2.e          reg_3.e
## Dependent Var.:               y_dynamic        y_dynamic        y_dynamic
##
## Constant              -3.240*** (0.0355)
## treated_group          1.479*** (0.0512) 1.479*** (0.0371)
## post                  -5.075*** (0.0519)
## treated_group x post  2.076*** (0.0690) 2.076*** (0.0509) 2.076*** (0.0362)
## Fixed-Effects:        ------------------ ----------------- -----------------
## time_id                               No               Yes               Yes
## ids                                   No                No               Yes
## _____________________ __________________ _________________ _________________
## S.E. type            Heteroskedas.-rob. Heteroskeda.-rob. Heteroskeda.-rob.
## Observations                     10,000            10,000            10,000
## R2                              0.66903           0.82004           0.92410
## Within R2                            --           0.56312           0.27208
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The treatment effect has increased.

```
myPlot <- myData %>%
  group_by(time_id, treated_group) %>%
  mutate(y_dyn = mean(y_dynamic)) %>%
  select(time_id, treated_group, y_dyn) %>%
  distinct() %>%
  mutate(time_id = as.numeric(time_id)) %>%
  mutate(treated_group = as.factor(treated_group))

ggplot(myPlot, aes(x = time_id, y = y_dyn, color = treated_group)) +
  geom_point() +
  theme_bw() +
  ggtitle("Dynamic ")
```



We see that the decreasing trend is now halted for 3 time periods for the treated group (6, 7, 8).

```
# regression
myData <- myData_og %>%
  mutate(time_id = relevel(time_id, ref = "4"))

reg_4.e <- feols(y_dynamic ~  treated_group*time_id | time_id + ids, data = myData, vcov = "hetero")
```

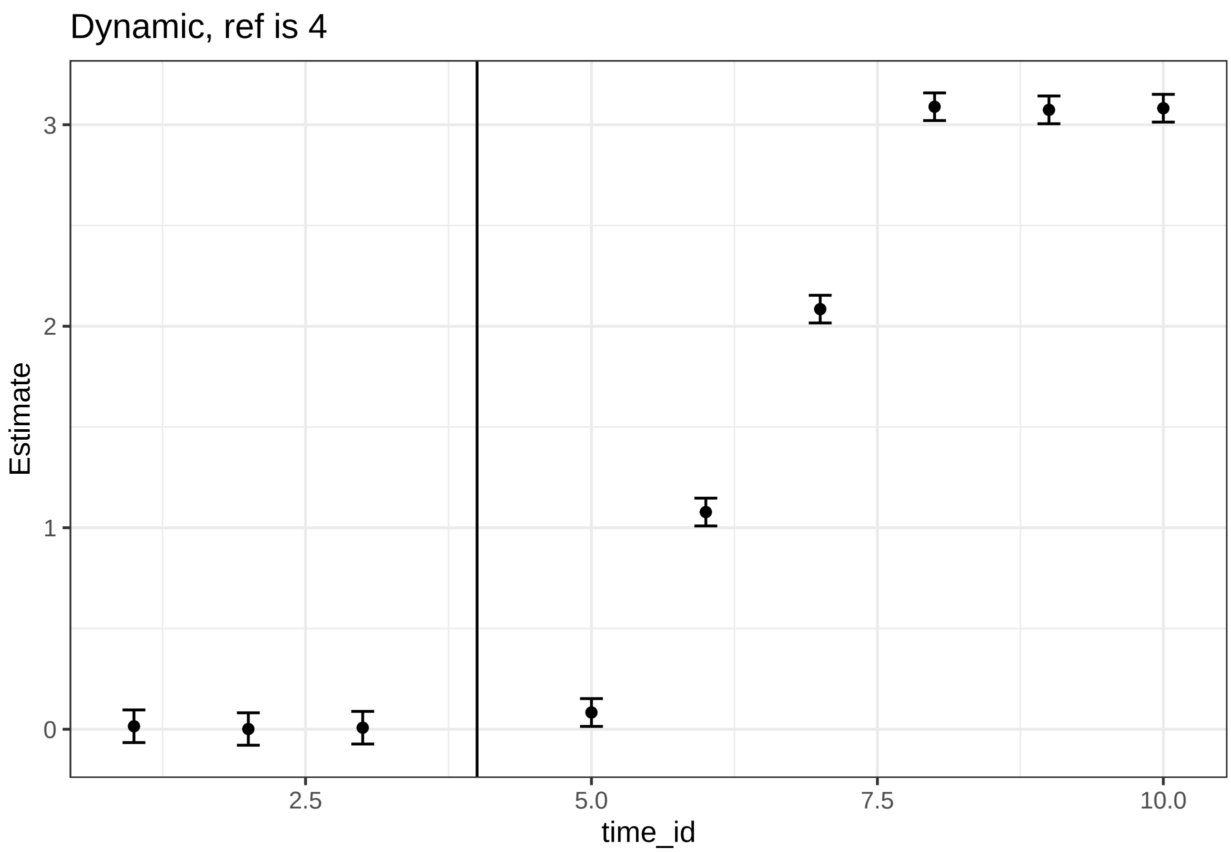
```
## The variables 'treated_group', 'time_id1' and eight others have been removed because of collinearity
```

```
etable(reg_4.e)
```

```
##                                        reg_4.e
## Dependent Var.:                      y_dynamic
##
## treated_group x time_id1    0.0147 (0.0810)
## treated_group x time_id2    0.0010 (0.0805)
## treated_group x time_id3    0.0075 (0.0810)
## treated_group x time_id5    0.0830 (0.0689)
## treated_group x time_id6  1.078*** (0.0690)
## treated_group x time_id7  2.085*** (0.0687)
## treated_group x time_id8  3.089*** (0.0687)
## treated_group x time_id9  3.074*** (0.0689)
## treated_group x time_id10 3.082*** (0.0688)
## Fixed-Effects:            -----------------
## time_id                                 Yes
## ids                                     Yes
## _________________________ _________________
## S.E. type                 Heteroskeda.-rob.
## Observations                         10,000
## R2                                  0.94605
## Within R2                           0.48259
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# plot
estDF_4.e<- as.data.frame(reg_4.e$coeftable) %>%
  mutate(time_id = c(1,2, 3, 5, 6, 7, 8, 9, 10)) %>%
  rename(se = `Std. Error`)

ggplot(estDF_4.e, aes(x = time_id, y = Estimate)) +
  geom_point() +
  geom_errorbar(aes(ymin=Estimate-se, ymax=Estimate+se), width=.2,
                position=position_dodge(.9)) +
  geom_vline(xintercept = 4) +
  theme_bw() +
  ggtitle("Dynamic, ref is 4 ")
```



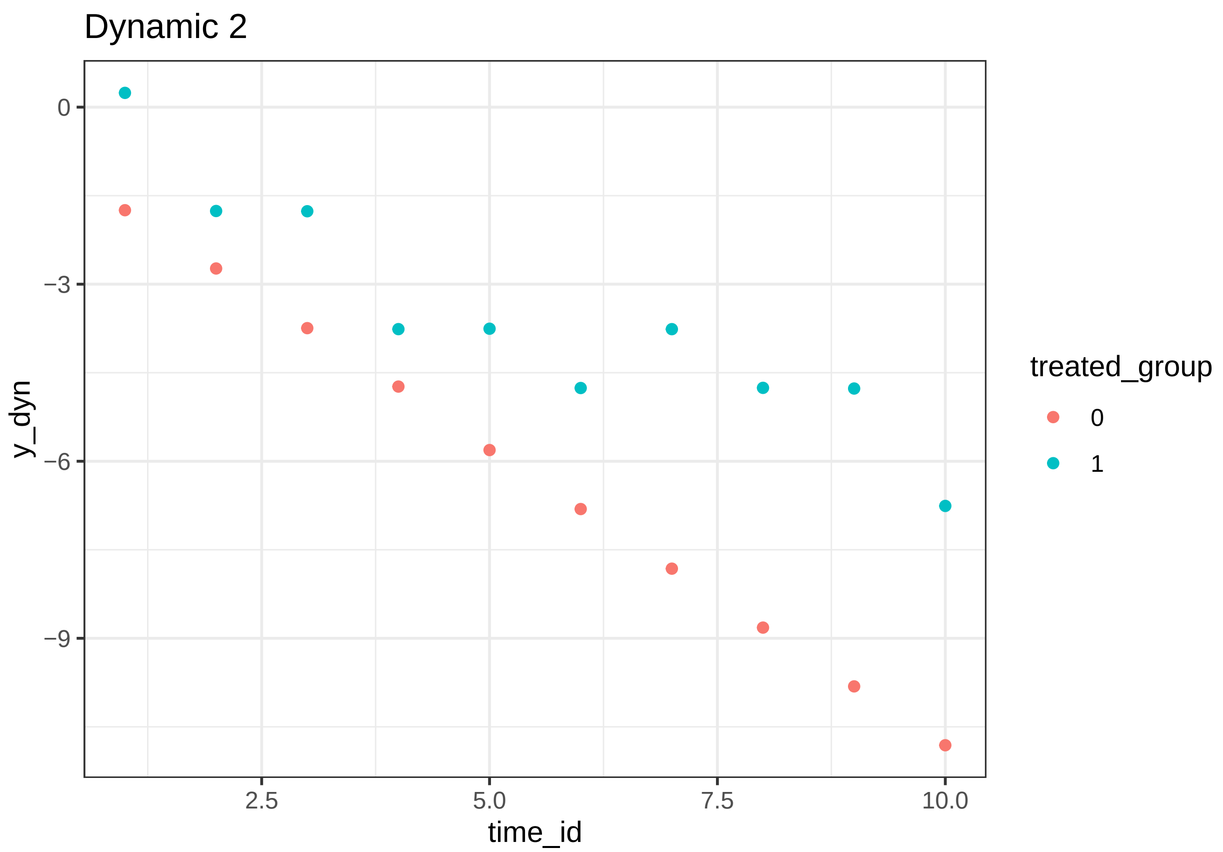
Rather than there being one time period where the trend is forgone, there are multiple. Therefore the treatment effect for each time period increase from time period 5 through 8, and then the treatment affect stabilizes for the remaining time periods.

f)

```
myData <- myData_og %>%
  mutate(time_id = relevel(time_id, ref = "1"))

myPlot <- myData %>%
  group_by(time_id, treated_group) %>%
  mutate(y_dyn = mean(y_dynamic2)) %>%
  select(time_id, treated_group, y_dyn) %>%
  distinct() %>%
  mutate(time_id = as.numeric(time_id)) %>%
  mutate(treated_group = as.factor(treated_group))

ggplot(myPlot, aes(x = time_id, y = y_dyn, color = treated_group)) +
  geom_point() +
  theme_bw() +
  ggtitle("Dynamic 2")
```



The pre-trend is not perfectly parallel. However, it this were empirical data I'd argue that it fits well enough to do a diff-in-diff.

```
# regression
myData <- myData_og %>%
  mutate(time_id = relevel(time_id, ref = "4"))

reg_4.f <- feols(y_dynamic2 ~  treated_group*time_id | time_id + ids, data = myData, vcov = "hetero")
```

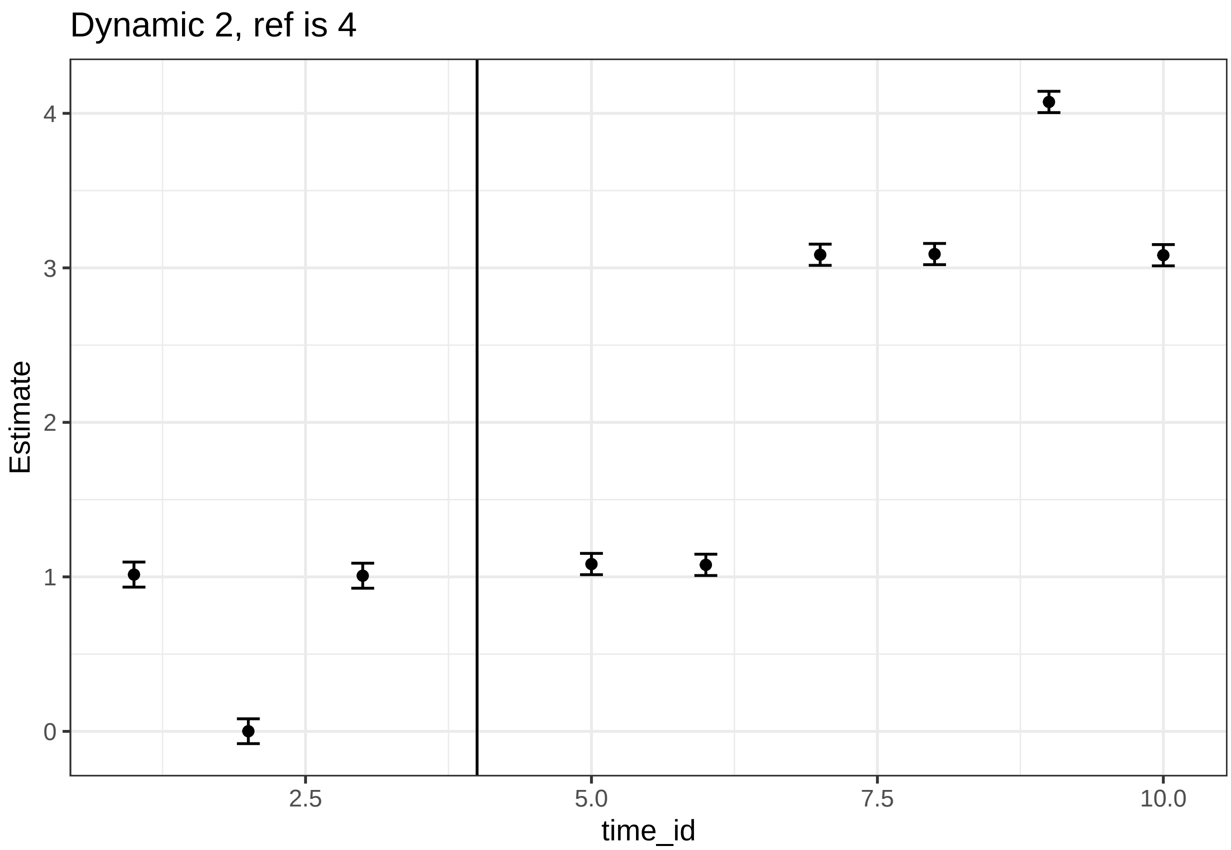
```
## The variables 'treated_group', 'time_id1' and eight others have been removed because of collinearity
```

```
etable(reg_4.f)
```

```
##                                      reg_4.f
## Dependent Var.:                   y_dynamic2
##
## treated_group x time_id1  1.015*** (0.0810)
## treated_group x time_id2    0.0010 (0.0805)
## treated_group x time_id3  1.008*** (0.0810)
## treated_group x time_id5  1.083*** (0.0689)
## treated_group x time_id6  1.078*** (0.0690)
## treated_group x time_id7  3.085*** (0.0687)
## treated_group x time_id8  3.089*** (0.0687)
## treated_group x time_id9  4.074*** (0.0689)
## treated_group x time_id10 3.082*** (0.0688)
## Fixed-Effects:            -----------------
## time_id                                   Yes
## ids                                       Yes
## _________________________ _________________
## S.E. type                 Heteroskeda.-rob.
## Observations                         10,000
## R2                                  0.94758
## Within R2                           0.48949
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# plot
estDF_4.f<- as.data.frame(reg_4.f$coeftable) %>%
  mutate(time_id = c(1,2, 3,  5, 6, 7, 8, 9, 10)) %>%
  rename(se = `Std. Error`)

ggplot(estDF_4.f, aes(x = time_id, y = Estimate)) +
  geom_point() +
  geom_errorbar(aes(ymin=Estimate-se, ymax=Estimate+se), width=.2,
                position=position_dodge(.9)) +
  geom_vline(xintercept = 4) +
  theme_bw() +
  ggtitle("Dynamic 2, ref is 4")
```



```
# regression
myData <- myData_og %>%
  mutate(time_id = relevel(time_id, ref = "3"))

reg_5.f <- feols(y_dynamic2 ~  treated_group*time_id | time_id + ids, data = myData, vcov = "hetero")
```

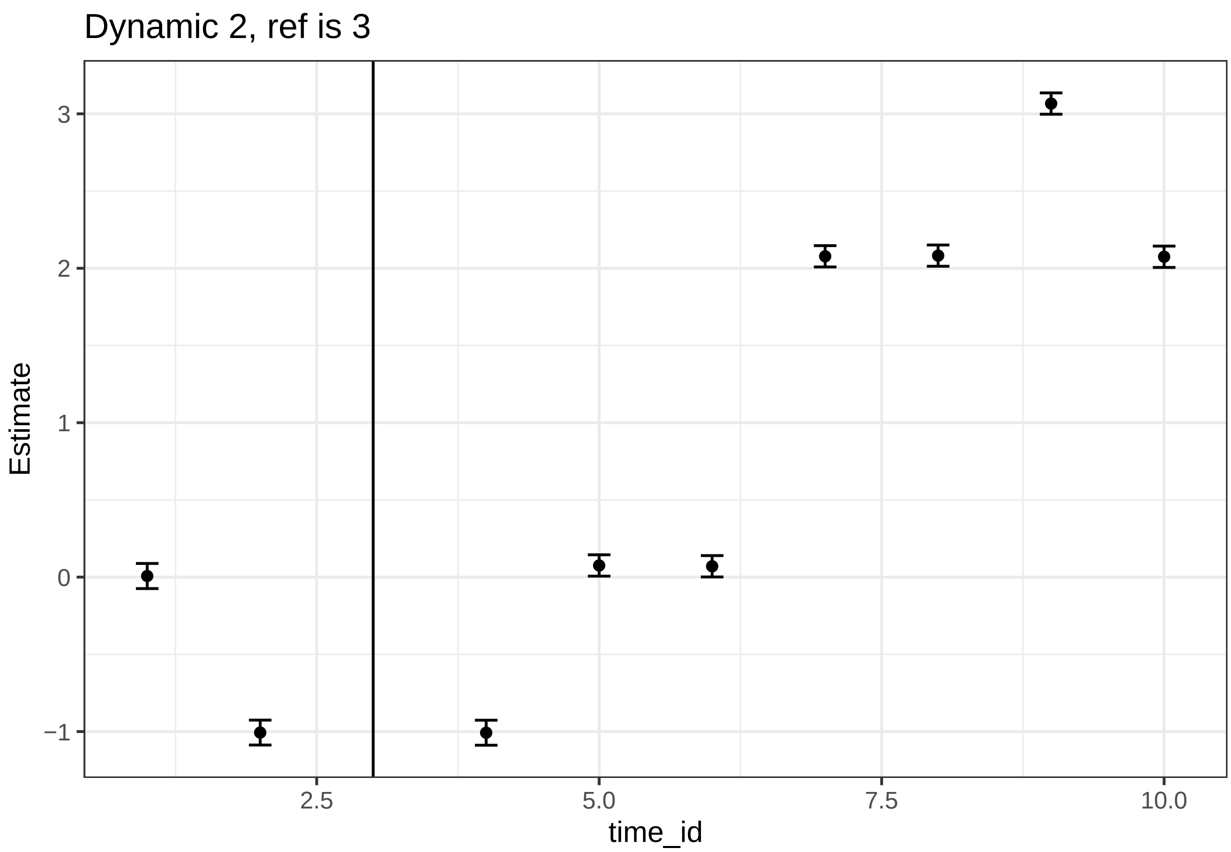
```
## The variables 'treated_group', 'time_id1' and eight others have been removed because of collinearity
```

```
etable(reg_5.f)
```

```
##                                        reg_5.f
## Dependent Var.:                     y_dynamic2
##
## treated_group x time_id1     0.0072 (0.0812)
## treated_group x time_id2  -1.007*** (0.0807)
## treated_group x time_id4  -1.008*** (0.0810)
## treated_group x time_id5     0.0755 (0.0691)
## treated_group x time_id6     0.0703 (0.0692)
## treated_group x time_id7   2.077*** (0.0689)
## treated_group x time_id8   2.082*** (0.0689)
## treated_group x time_id9   3.066*** (0.0691)
## treated_group x time_id10  2.074*** (0.0691)
## Fixed-Effects:            ------------------
## time_id                                  Yes
## ids                                      Yes
## _________________________ __________________
## S.E. type                 Heteroskedas.-rob.
## Observations                          10,000
## R2                                   0.94758
## Within R2                            0.48949
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# plot
estDF_5.f<- as.data.frame(reg_5.f$coeftable) %>%
  mutate(time_id = c(1,2, 4,  5, 6, 7, 8, 9, 10)) %>%
  rename(se = `Std. Error`)

ggplot(estDF_5.f, aes(x = time_id, y = Estimate)) +
  geom_point() +
  geom_errorbar(aes(ymin=Estimate-se, ymax=Estimate+se), width=.2,
                position=position_dodge(.9)) +
  geom_vline(xintercept = 3) +
  theme_bw() +
  ggtitle("Dynamic 2, ref is 3")
```



g)

In part A I already used heteroskedastically robust standard errors. I now repeat part A with robust SE that are clustered by id.

```
myData <- myData_og %>%
  mutate(time_id = relevel(time_id, ref = "1"))

reg_3.g.hom <- feols(y_instant ~  treated_group*post | time_id + ids, data = myData, vcov = "iid")
```

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

```
reg_3.g.rob <- feols(y_instant ~  treated_group*post | time_id + ids, data = myData, vcov = "hetero")
```

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

```
reg_3.g.clust <- feols(y_instant ~  treated_group*post | time_id + ids, data = myData, cluster = myData$
```

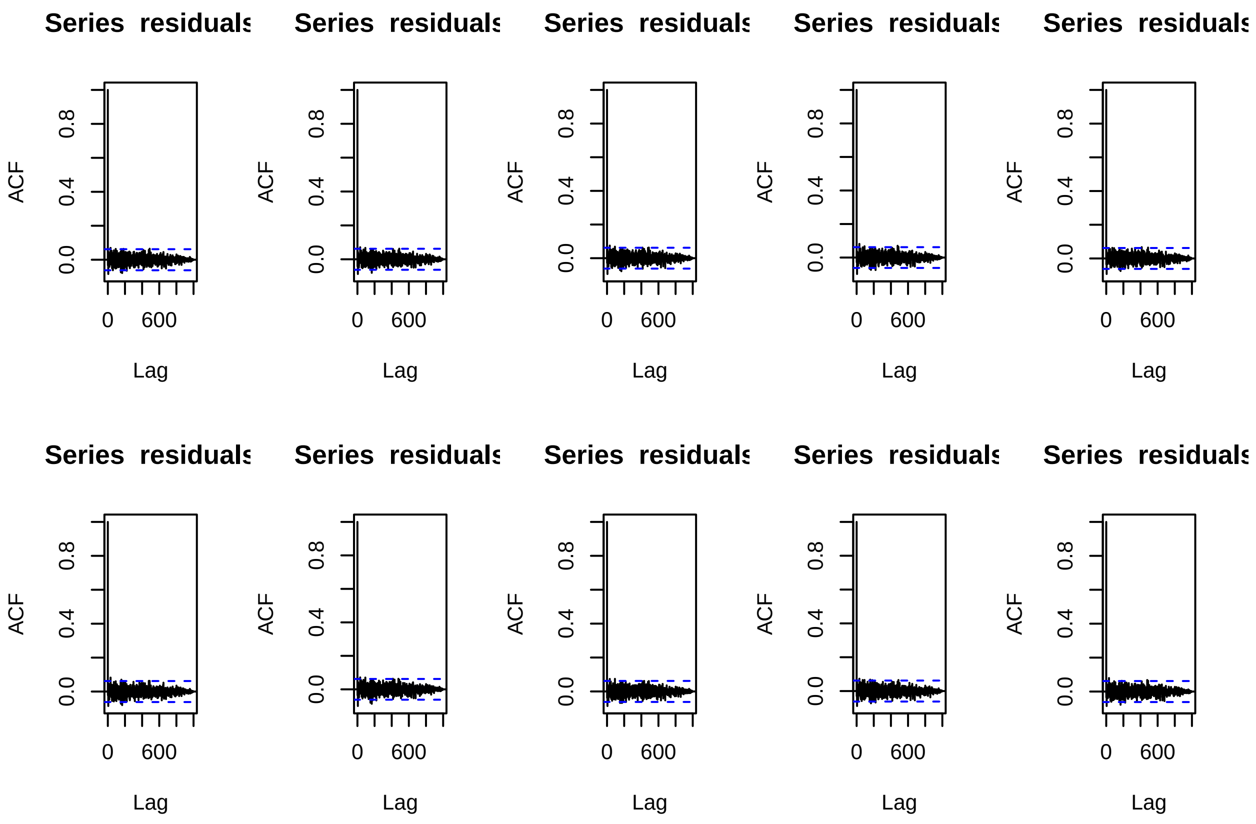
```
## The variables 'treated_group' and 'post' have been removed because of collinearity (see $collin.var)
```

```
etable(reg_3.g.hom, reg_3.g.rob, reg_3.g.clust)
```

```
##                            reg_3.g.hom       reg_3.g.rob     reg_3.g.clust
## Dependent Var.:              y_instant         y_instant         y_instant
##
## treated_group x post 1.076*** (0.0302) 1.076*** (0.0326) 1.076*** (0.0899)
## Fixed-Effects:       ----------------- ----------------- -----------------
## time_id                            Yes               Yes               Yes
## ids                                Yes               Yes               Yes
## ____________________ _________________ _________________ _________________
## S.E. type                          IID Heteroskeda.-rob.       by: cluster
## Observations                    10,000            10,000            10,000
## R2                             0.95022           0.95022           0.95022
## Within R2                      0.12377           0.12377           0.12377
## ---
## Signif. codes: 0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

The robust standard error is smaller than the clustered standard error, but larger than than the SE when assuming errors are IID.

```
par(mfrow = c(2, 5))
test <- myData_og %>%
  mutate(residuals = resid(reg_3.g.rob)) %>%
  group_by(time_id) %>%
  summarise(cor=list(acf(residuals, lag.max = 1000)))
```



There doesn’t appear to be any auto correlation to be concerned about.