Princeton predoc

Gergel Anastasia (Asya)

2023-05-31

Task 1: constructing panel dataset

```
setwd('/Users/asya/Dropbox/_ jobs/princeton')
library(tidyverse)

voter_dta <- read.csv('voter_dta.csv')

years <- seq(1987, 1997) # a sequence of years from 1987 to 1997


### create the base panel dataset
# create a grid of all possible combinations between unique family_id values and the years sequence
panel_data <- expand_grid(family_id = unique(voter_dta$family_id),
                          year = years) %>%
  # create a boolean vector that indicates whether a birth occurred in each year for each family
  group_by(family_id) %>%
  mutate(birth_occurred = if_else(year %in% voter_dta[voter_dta$family_id == family_id,
                                                      "birth_year"], 1, 0))



### merge the base panel dataset with voter_dta, matching the respective family and year
panel_data <- panel_data %>%
  left_join(voter_dta, by = c("family_id", "year" = "birth_year")) %>%

  ### var transmutations
  # expand the within-group constants (father_value and religion)
  group_by(family_id) %>%
  mutate(father_value = mean(father_value, na.rm = TRUE),
         religion = unique(na.omit(religion))) %>%
  # create the 'family_religion_score' variable
  # set it equal to 'voter_value' first
  ungroup() %>%
  mutate(family_religion_score = voter_value) %>%
  # for each group, fill NAs downwards
  group_by(family_id) %>%
  # i.e.repeat the previous non-missing value in the column until a new value is encountered,
  # then repeat the new value
  fill(family_religion_score, .direction = "down") %>%
  # replace (the before-first-son) NAs with father_value
```

```r
  mutate(family_religion_score = if_else(is.na(family_religion_score),
                                         father_value, family_religion_score))



### leave only: family_id, year, family's religion, and a family religion score
target_data <- panel_data %>%
  select(c(family_id, year, religion, family_religion_score))
```

(a) How many Muslim families are there?

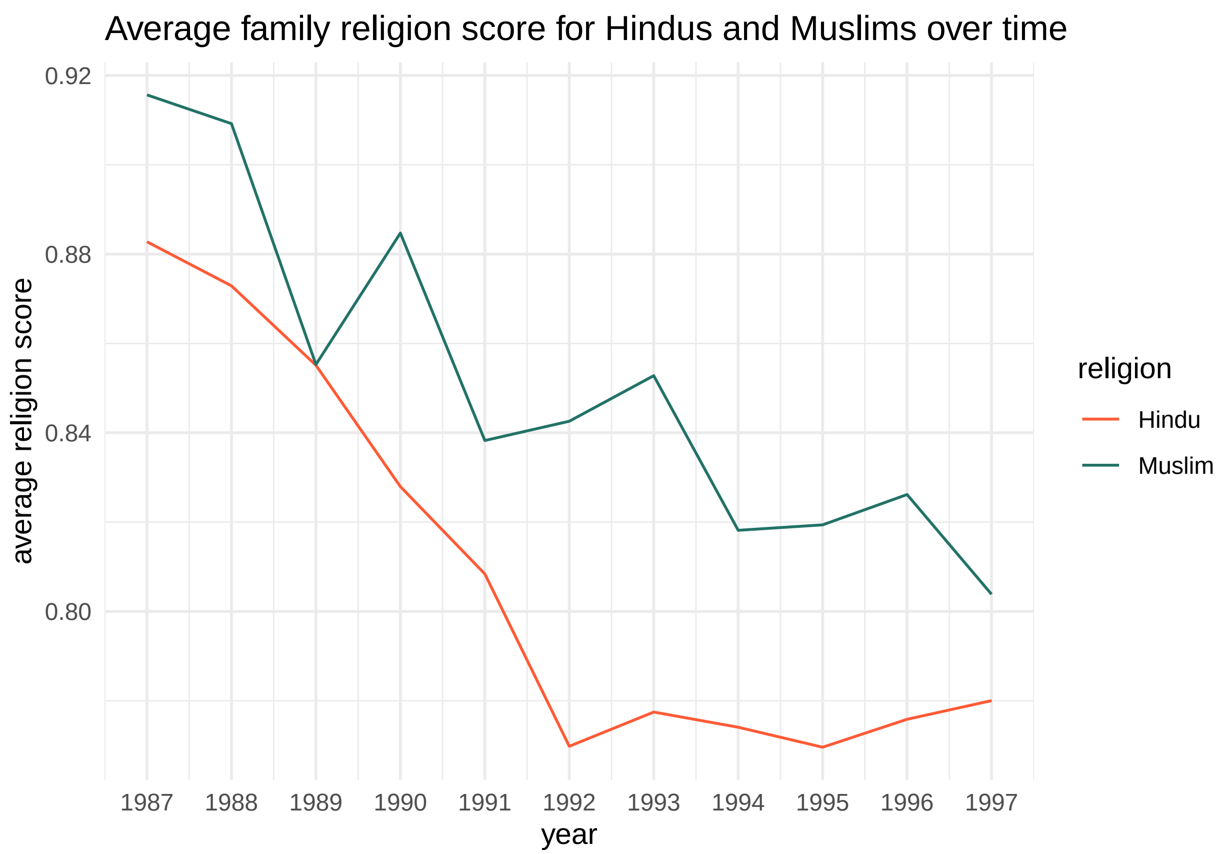
```r
target_data %>%
filter(religion == 'Muslim')  %>%
  distinct(family_id) %>% nrow()
```

```
[1] 53
```

A: 53 Muslim families.

(b) Create a line plot that shows the average family religion score for Hindus and Muslims over time (different color line for each religion).

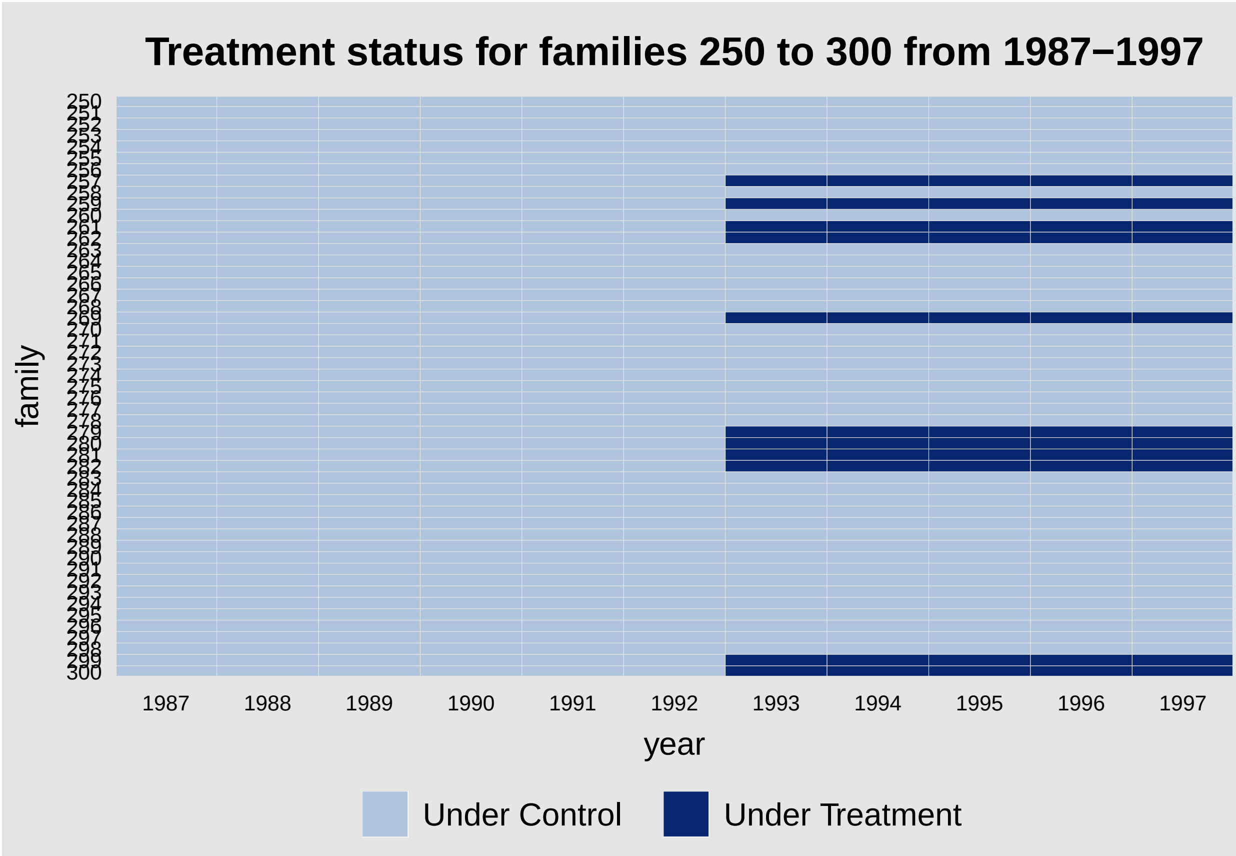
```r
library(ggplot2)

target_data %>%
  group_by(year, religion) %>%
  # calculate the average religion score for each religion for each year
  summarise(avr_score = mean(family_religion_score)) %>%
  # create the plot
  ggplot() +
  aes(x = year, y = avr_score, color = religion) +
  geom_line() +
  # customize the appearance of the plot
  scale_color_manual(values=c('#FF5A36', '#227267')) +
  scale_x_continuous(breaks = 1987:1997) +
  labs(title = 'Average family religion score for Hindus and Muslims over time',
       y = 'average religion score') +
  theme_minimal()
```



Task 2: visualizing using panelView

Create a plot using `panelView` to show the treatment status for families 250 to 300 from 1987-1997.

```
library(panelView)

# create the treatment indicator variable (Muslim family, post-1992)
target_data$treatment_indicator <- ifelse(target_data$religion == "Muslim" &
                                            target_data$year > 1992, 1, 0)

panelview(family_religion_score ~ treatment_indicator,
          # include families from 250 to 300 only
          data = target_data[target_data$family_id >= 250 & target_data$family_id <= 300, ],
          index = c("family_id","year"),
          xlab = "year",
          ylab = "family",
          main = "Treatment status for families 250 to 300 from 1987-1997")
```



Task 3: estimating equation, DiD

The two-way group-time fixed effects specification for a two-period DiD design takes the following form:

$$Y_{it} = \beta_0 + \beta_1 T_i + \beta_2 \text{Post}_t + \delta(T_i * \text{Post}_t) + \epsilon_{i,t}$$

where

- Y_{it} is the observed outcomes (family naming conventions proxied by the family religion score) for family i at time period t;
- T_i is a dummy equal to 1 if family is from the treated group (Muslim) and 0 if from the control group (Hindu);
- Post_t is a dummy equal to 1 if time period t is after the treatment (demolition of Babri Masjid, 1992) and 0 otherwise;
- β_0 is the intercept, i.e. the average family religion score for Hindu before 1992;
- β_1 represents the difference in the average treatment effect (ATE) between the treatment (Muslim) and control (Hindu) groups before the 1992 intervention;
- β_2 represents the difference in the average effect of time between the pre-intervention and post-intervention periods for the control (Hindu) group;
- δ represents the difference in the ATE between the pre-intervention and post-intervention periods for the treatment (Muslim) group, i.e. the DiD estimate;
- $\epsilon_{i,t}$ is the error term capturing the effect of factors not accounted for in the model.

Hence, the DiD estimate is

$$\mathbb{E}[Y_{i1} - Y_{i0} | T_i = 1] - \mathbb{E}[Y_{i1} - Y_{i0} | T_i = 0] = [(\beta_0 + \beta_1 + \beta_2 + \delta) - (\beta_0 + \beta_1)] - [(\beta_0 + \beta_2) - \beta_0] = \delta$$

i.e. (Treatment_post - Treatment_pre) - (Control_post - Control_pre).

Task 4: estimating the causal effect of the Babri Masjid demolition on naming conventions

Model 1: static DiD with lm, no fixed effects

```
library(sandwich)
library(stargazer)

### add DiD dummies to the dataset
# treat dummy, T_i
target_data$treat <- ifelse(target_data$religion == "Muslim", 1, 0)
# time dummy, Post_t
target_data$post <- ifelse(target_data$year > 1992, 1, 0)


m1 <- lm(family_religion_score ~ treat + post + treat:post, data = target_data)

clustered_cov <- vcovCL(m1, cluster = ~ family_id, type = 'HC1')
robust_se_m1 <- sqrt(diag(clustered_cov)) # clustered robust SEs


stargazer(m1,
          type = 'text',
          dep.var.labels = "family religion score",
          covariate.labels = c("treat", "post", "treat:post", "intercept"),
          se = list(robust_se_m1),
          add.lines = list(c("Number of Clusters", length(unique(target_data$family_id))),
                           c("Observations", nrow(target_data)),
                           c("Controls", "No"),
                           c("Fixed Effects", "No"),
                           c("Clustered Robust SEs", "by Family ID")))
```

```
================================================
                         Dependent variable:
                     ---------------------------
                       family religion score
------------------------------------------------
treat                           0.038
                               (0.028)

post                          -0.061***
                               (0.009)

treat:post                      0.011
                               (0.035)

intercept                     0.836***
```

```
                               (0.007)

------------------------------------------------
Number of Clusters                 599
Observations                      6589
Controls                           No
Fixed Effects                      No
Clustered Robust SEs        by Family ID
Observations                      6,589
R2                                0.020
Adjusted R2                       0.019
Residual Std. Error       0.226 (df = 6585)
F Statistic            44.468*** (df = 3; 6585)
================================================
Note:                *p<0.1; **p<0.05; ***p<0.01
```

Model 2: static DiD, two-way fixed effects model

The classic two-way fixed effects model for panel data, controlling for family-specific and time-specific effects. Due to the presence of fixed effects in the model, I remove `treat` and `post` dummies from the equation and leave their interaction term only to avoid collinearity, since these variables represent a linear combination of family and year.

```r
library(fixest)
m2 <- feols(family_religion_score ~ treat:post | family_id + year,
            cluster = ~ family_id, # clustered robust SEs
            target_data)

# stargazer does not support `feols`
etable(m2, tex=F,
       extralines = list("^_Controls"= c("No", ""),
                         "^Number of Clusters" =
                           c(length(unique(target_data$family_id)), "")))
```

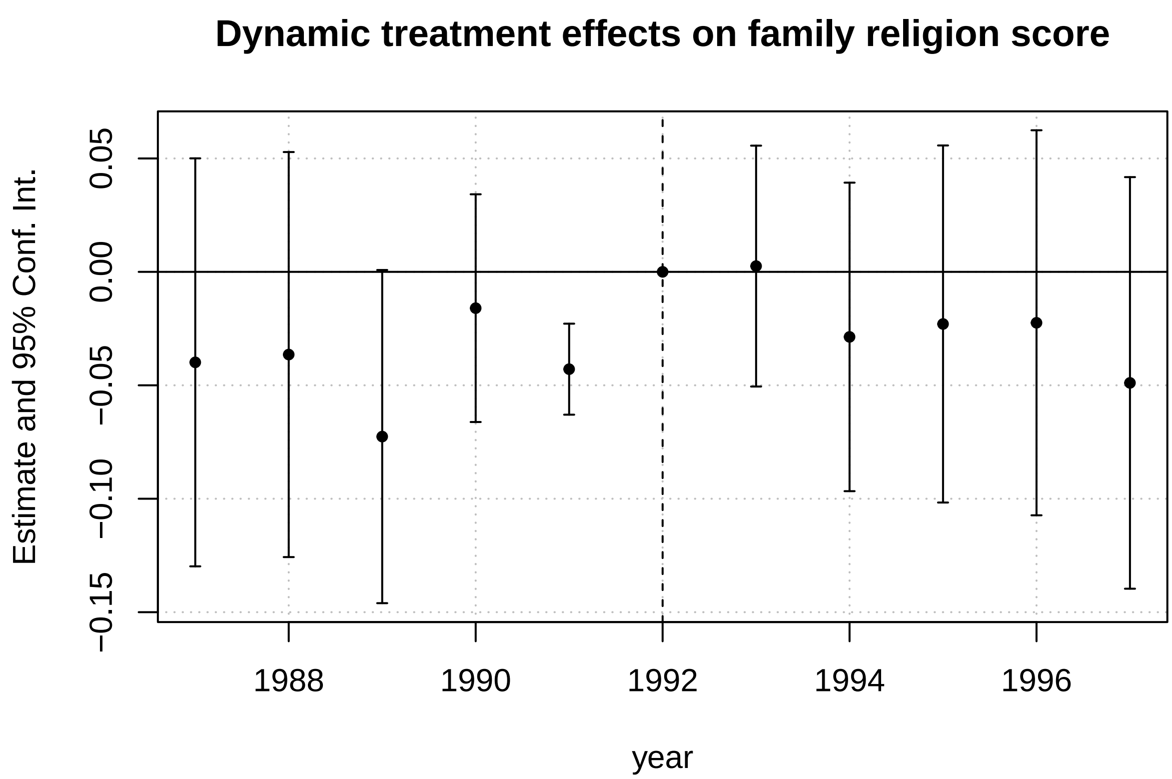
```
                                        m2
Dependent Var.:    family_religion_score

treat x post             0.0105 (0.0353)
Controls                              No
Number of Clusters                   599
Fixed-Effects:     ---------------------
family_id                            Yes
year                                 Yes
__________________ _____________________
S.E.: Clustered             by: family_id
Observations                       6,589
R2                               0.51316
Within R2                        8.73e-5
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Task 5: event study plot

Event Study 1: the classic dynamic DiD

The standard event study approach is the distributed lag two-way fixed effects model that estimates dynamic treatment effects (i.e. estimating effects of years relative to 1992) and allows to compare pre-trends.

```
m3 <- feols(family_religion_score ~ i(year, treat, 1992) | family_id + year,
            cluster = ~ family_id,
            target_data)

iplot(m3, main = "Dynamic treatment effects on family religion score")
```



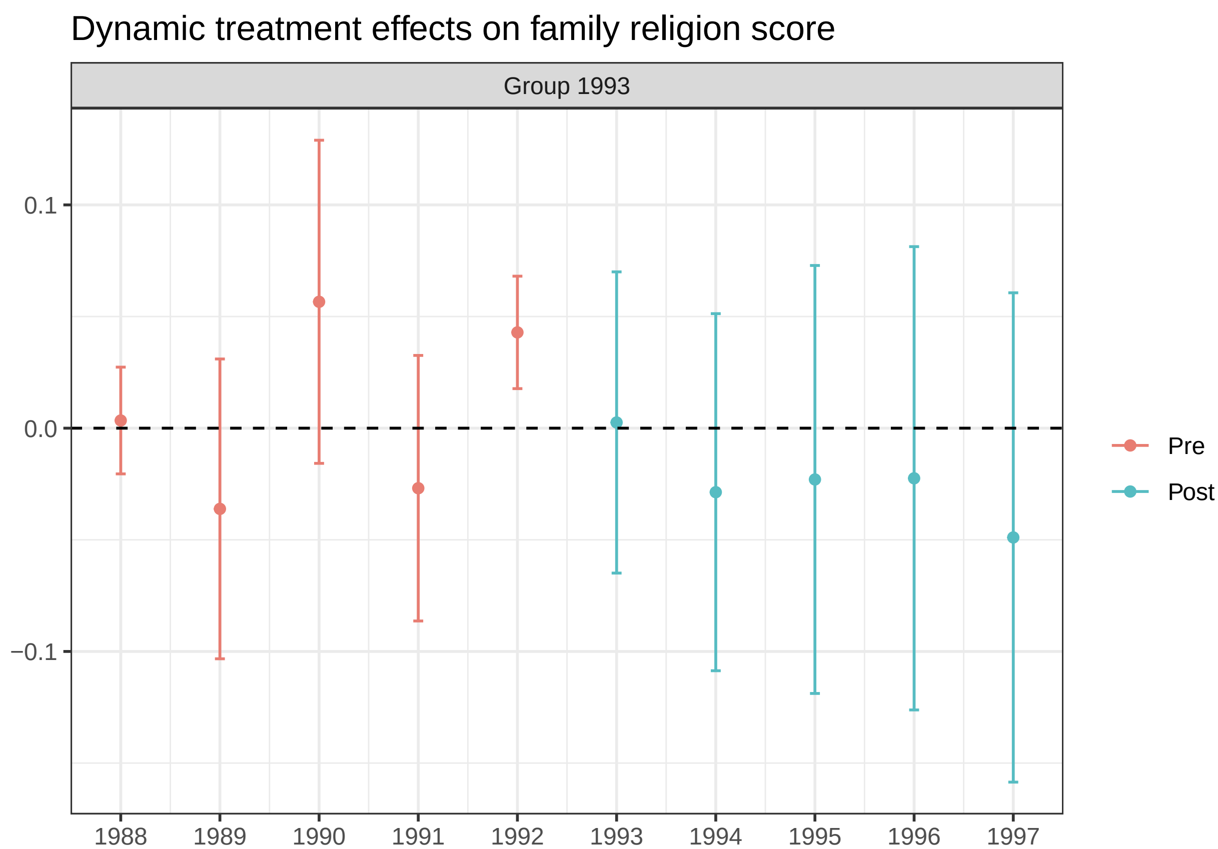
The Callaway and Sant'Anna (2020) approach in estimating group-time average treatment effects is another estimation strategy that shows dynamic treatment effects, i.e. event-study parameters.

```
library(did)

target_data$gname <- ifelse(target_data$religion == "Muslim", 1993, 0)

set.seed(1)
out <- att_gt(yname = "family_religion_score",
              gname = "gname",
              idname = "family_id",
              tname = "year",
```

```r
              xformla = ~1, # covariates (if any)
              data = target_data,
              est_method = "reg",
              allow_unbalanced_panel = T # allow unbalanced panel
              )
#tidy(out) %>% head()

# dynamic ATTs
aggte(out, type = "dynamic", na.rm=T) %>% tidy() %>%
  filter(event.time%in%seq(-5, 5, 1)) %>% select(term, event.time, estimate,
                                                std.error, conf.low, conf.high)
```

```
      term event.time     estimate   std.error     conf.low  conf.high
1  ATT(-5)         -5  0.003431403 0.009556771 -0.019888224 0.02675103
2  ATT(-4)         -4 -0.036149358 0.027819976 -0.104033321 0.03173460
3  ATT(-3)         -3  0.056598894 0.025706364 -0.006127611 0.11932540
4  ATT(-2)         -2 -0.026888567 0.025525052 -0.089172649 0.03539551
5  ATT(-1)         -1  0.042888625 0.011055227  0.015912592 0.06986466
6   ATT(0)          0  0.002547314 0.028769671 -0.067654014 0.07274864
7   ATT(1)          1 -0.028670709 0.035138812 -0.114413478 0.05707206
8   ATT(2)          2 -0.022972847 0.040340968 -0.121409479 0.07546379
9   ATT(3)          3 -0.022445278 0.044970871 -0.132179409 0.08728885
10  ATT(4)          4 -0.048944116 0.044601799 -0.157777669 0.05988944
```

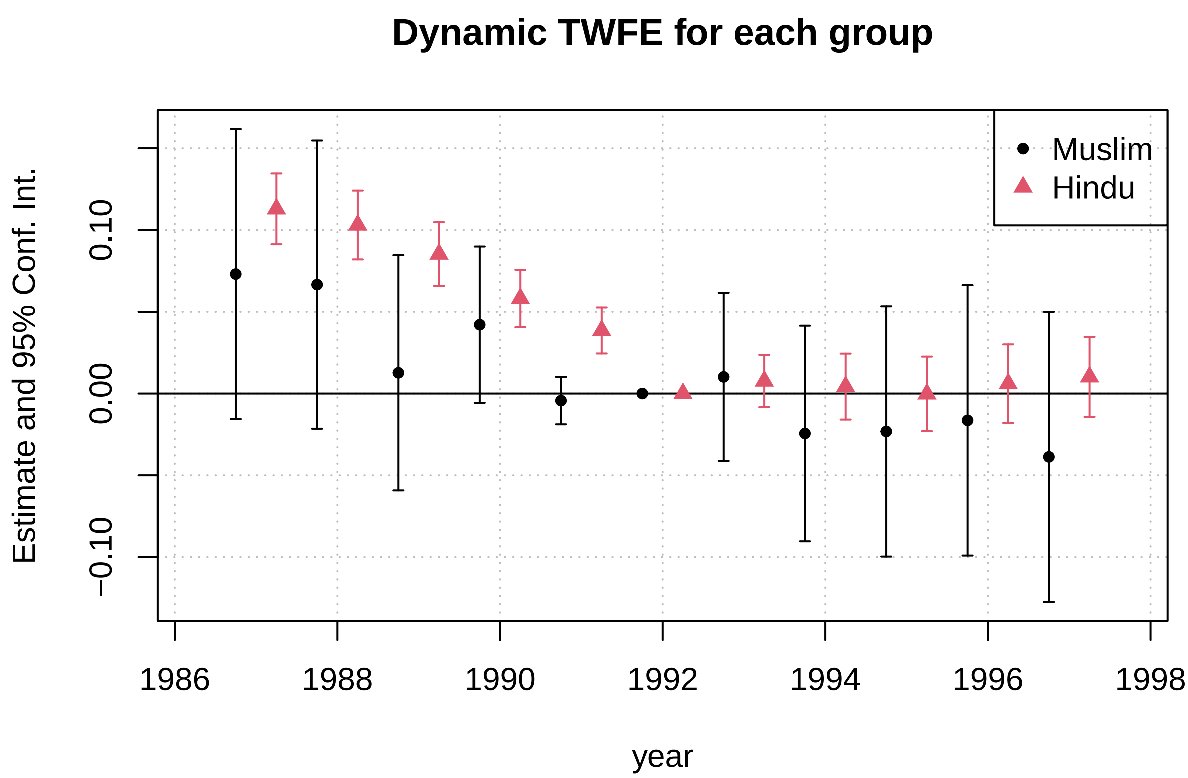
```r
ggdid(out, xgap=1, title="Dynamic treatment effects on family religion score",
      theming=F)+ theme_bw()
```



Event Study 2: analyze C and T groups separately

Analyze control and treatment groups separately to understand if we observe any changes there. De facto, I am estimating two time fixed effects models (`m3_muslim` and `m3_hindu`) separately for each group relative to the reference year, 1992. Alternatively, we can estimate an interaction term with the treatment variable (model `a`) and extract the coefficients and errors to plot the event study.

```
m3_muslim <- feols(family_religion_score ~ i(year,  ref = 1992) | family_id,
                   cluster = ~ family_id,
                   subset(target_data, treat == 1))

m3_hindu <- feols(family_religion_score ~ i(year,  ref = 1992) | family_id,
                  cluster = ~ family_id,
                  subset(target_data, treat == 0))

iplot(list(m3_muslim, m3_hindu), sep = .5,
      main = "Dynamic TWFE for each group")
legend("topright", col = c(1, 2),
       pch = c(20, 17),
       legend = c("Muslim", "Hindu"))
```




```r
### estimate a joint model for ggplot visualization
library(stringr)
a <- feols(family_religion_score ~ i(year,  ref = 1992) * treat | family_id,
                   cluster = ~ family_id,
                   target_data)

# create a df for ggplot with estimates, SEs, treat var, years
df_a <- data_frame(coef = a$coefficients,
                   se = a$se,
                   group = c(rep("Hindu", 10), rep("Muslim", 10)),
                   year = as.numeric(str_extract(names(a$se), pattern = '\\d+')))

# add the reference year observations
df_a <- df_a %>%
  rbind(data_frame(coef = rep(0, 2),
                   se = rep(NA, 2),
                   group = c("Hindu", "Muslim"),
                   year = rep(1992, 2))
)

df_a %>%
  ggplot(aes(x = year, y = coef, color = group)) +
  geom_point() +
  geom_line() +
  geom_errorbar(aes(ymin = coef - qnorm(0.975)*se,
                    ymax = coef + qnorm(0.975)*se),
```

```
               width = .3) +
  geom_vline(xintercept = 1992, linetype = 'dotted') +
  geom_hline(yintercept = 0) +
  scale_color_manual(values=c('#FF5A36', '#227267')) +
  labs(title = "Dynamic TWFE for each group", y = "estimates") +
  theme_minimal()
```