Final Project

Yosuke Abe Arisa Itagaki

2024-02-02

Table of contents

0.1 Setup

```
knitr::opts_chunk$set(echo = TRUE, cache = TRUE, message = FALSE, warning = FALSE)
```

```
library(tidyverse)
library(fixest)
library(patchwork)
library(here)
```

1 About this Report

1.1 Project Type

In this report, we have chosen to replicate a previously paper.

The paper we replicate is Baker, Larcker, and Wang (2021), "How Much Should We Trust Staggered Difference-In-Differences Estimates?" We used the working paper version for conducting Monte Carlo simulations in our report.

1.2 Summary of the paper

1.2.1 What the problem is

In this paper, the authors conduct Monte Carlo simulations to evaluate policy assessments with different treatment timings and analyze the methodologies. The issue identified in the paper is that using the standard two-way fixed effects model for estimation can lead to biases when treatment effects vary across units and over time.

The paper analyzes the effectiveness of the Staggered DiD methodology used to assess policy impacts on firm outcomes, points out that studies using this methodology may contain potential biases, and suggests a more robust alternative methodology.

As simulated in Section 2.2, biases occur in such cases, particularly when the timing of the treatment differs and the treatment effects are dynamic, which can even lead to a reversal in the sign of the estimated results.

1.2.2 Why it is important

In the paper, the authors highlight that many empirical studies rely on staggered DiD designs to estimate causal effects of interventions. However, estimation in Staggered DiD design can introduce biases depending on the circumstances, making it important to ensure the validity and reliability of the analysis.

1.2.3 How you solve the problem

In this report, as seen in Section 3, we address the bias that arises when there is staggered treatment and heterogeneity in treatment effects by using the Sun and Abraham estimator for estimation. Sun and Abraham (2020) use a specification with saturated interactions, including relative time indicators D_{it}^k (where k represents the relative years from the base year) and cohort indicators $1\{G_g = g\}$ (where G_g represents the group that received the treatment in year g). They refer to this as the "Interaction-Weighted" (IW) estimator. A key feature of this estimator is that it uses only units that never received treatment as the effective comparison units. We compare the results obtained using this estimator with those from the standard TWFE model to ensure that there is no bias.

1.2.4 What we find

It has been found that in estimations with staggered treatments and heterogeneous treatment effects, bias can occur, and sometimes even the sign of the estimated value can be reversed. This result will be visualized in Section 2.2.2.

Additionally, we realized that another figure could be replicated because of the extension conducted, so I performed another replication at the end of the extension. By visualizing each relative year, it was possible to confirm its robustness.

1.3 Data and Analysis

1.3.1 Our data and analysis

The data used in this report is simulated data. The data is generated for 36 years for 1000 firms. In Simulations 1 and 2, a treatment is applied once in 1998, while in Simulation 3, treatments are applied in 1989, 1998, and 2007. In Simulation 1, the treatment group receives a uniform treatment with an average of 2 and a standard deviation of 0.2. In Simulation 2, the treatment group receives a dynamic treatment that increases by an average of 0.3 (standard deviation 0.2) each year. In Simulation 3, although the timings are different, each treatment group receives a uniform treatment with an average of 3 and a standard deviation of 0.2. In Simulations 4, 5, and 6, the basic data is the same as in Simulations 1 ~ 3, but each group receives treatments at three different timings. What differs in each simulation is the treatment effect. In Simulation 4, each treatment is uniform (with averages of 5, 3, and 1, respectively, and a standard deviation of 0.2 for all). In Simulation 5, dynamic treatments similar to Simulation 2 are applied, but their effects are equal. Simulation 6 is similar to Simulation 5, but the group treated first increases by an average of 0.5 each year, the second group by 0.3, and the last group by 0.1, with a standard deviation of 0.2 for all.

1.3.2 Estimating equations

- Standard TWEF model

$$y_{it} = \alpha_i + \lambda_t + \delta^{DD} D_{it} + \epsilon_{it}$$

- Sun and Abraham (2020) estimator

$$y_{it} = \alpha_i + \lambda_t + \sum_{e} \sum_{k \neq -1} \delta_{g,k} (1\{G_g = g\} \cdot D_{it}^k) + \epsilon_{it}$$

We use the Standard TWFE model to confirm biases in staggered situations. α_i represents individual fixed effects, λ_t represents time fixed effects, and D_{it} indicates $TREAT_i \cdot POST_t$.

In the extension part of this report, we use the estimator proposed by Sun and Abraham (2020). Here, Gg denotes the group that receives treatment in year g. The expression $1Gg = g$ takes the value 1 when the group's treatment year is equal to the relative year k.

2 Replication

2.1 Plot of Simulation 1,2, and 3

2.1.1 Data preparation (Yosuke Abe)

First, for simulations 1 to 3, create data for 36 years for 1000 firms. In Simulations 1 and 2, a treatment is applied once in 1998, while in Simulation 3, treatments are applied in 1989, 1998, and 2007. In Simulation 1, the treatment group receives a uniform treatment with an average of 2 and a standard deviation of 0.2. In Simulation 2, the treatment group receives a dynamic treatment that increases by an average of 0.3 (standard deviation 0.2) each year. In Simulation 3, although the timings are different, each treatment group receives a uniform treatment with an average of 3 and a standard deviation of 0.2. Based on the created data, estimate a standard TWEF model taking into account fixed effects for firms and years. Repeat this process 500 times, store the coefficients obtained from the estimation in a dataframe, and use them for plotting.

```
set.seed(57)

mcSim1to3_500runs <- tibble(
  "sim1" = numeric(500),
  "sim2" = numeric(500),
  "sim3" = numeric(500)
)

for (i in 1:500) {
  n_firms <- 1000
  n_states <- 50
  T <- 36

  firm_id <- 1:n_firms
  state_id <- sample(n_states, size = n_firms, replace = TRUE)
  year <- 1980:2015

  fe_firm <- rnorm(n_firms, mean = 0, sd = .5)
  fe_year <- rnorm(T, mean = 0, sd = .5)
  error <- rnorm(n_firms * T, mean = 0, sd = .5)

  # Not Staggered
  treated_1998 <- sample(c(1, 0), size = n_firms,
                         replace = TRUE, prob = c(1/2, 1/2))

  # Staggered
  treated_year <- sample(c(1989, 1998, 2007), size = n_firms,
                         replace = TRUE, prob = c(17/50, 17/50, 16/50))

  df_sim1 <- tibble(
    firm_id = rep(firm_id, each = T),
    state_id = rep(state_id, each = T),
    year = rep(year, times = n_firms),
    fe_firm = rep(fe_firm, each = T),
    fe_year = rep(fe_year, times = n_firms),
    error = error,
    treated_1998 = rep(treated_1998, each = T),
    is_treated = if_else(treated_1998 == 1 & year >= 1998, 1, 0),
```

```r
  y = case_when(
    is_treated == 1 ~
      rnorm(n_firms * T, mean = 2, sd = .2) + fe_firm + fe_year + error,
    .default = fe_firm + fe_year + error
  )
)

df_sim2 <- tibble(
  firm_id = rep(firm_id, each = T),
  state_id = rep(state_id, each = T),
  year = rep(year, times = n_firms),
  fe_firm = rep(fe_firm, each = T),
  fe_year = rep(fe_year, times = n_firms),
  error = error,
  treated_1998 = rep(treated_1998, each = T),
  is_treated = if_else(treated_1998 == 1 & year >= 1998, 1, 0),
  y = case_when(
    is_treated == 1 ~
      rnorm(n_firms * T, mean = .3, sd = .2) * (year - 1997) + fe_firm + fe_year + error,
    .default = fe_firm + fe_year + error
  )
)

df_sim3 <- tibble(
  firm_id = rep(firm_id, each = T),
  state_id = rep(state_id, each = T),
  year = rep(year, times = n_firms),
  fe_firm = rep(fe_firm, each = T),
  fe_year = rep(fe_year, times = n_firms),
  error = error,
  treated_year = rep(treated_year, each = T),
  is_treated = case_when(
    treated_year == 1989 & year >= 1989 ~ 1,
    treated_year == 1998 & year >= 1998 ~ 1,
    treated_year == 2007 & year >= 2007 ~ 1,
    TRUE ~ 0
  ),
  y = case_when(
    is_treated == 1 ~
      rnorm(n_firms * T, mean = 3, sd = .2) + fe_firm + fe_year + error,
    .default = fe_firm + fe_year + error
  )
)

est_sim1 <- df_sim1 |>
  feols(y ~ is_treated | firm_id + year, cluster = "state_id")
est_sim2 <- df_sim2 |>
  feols(y ~ is_treated | firm_id + year, cluster = "state_id")
est_sim3 <- df_sim3 |>
  feols(y ~ is_treated | firm_id + year, cluster = "state_id")

mcSim1to3_500runs$sim1[i] <- est_sim1$coefficients["is_treated"]
mcSim1to3_500runs$sim2[i] <- est_sim2$coefficients["is_treated"]
```

```r
  mcSim1to3_500runs$sim3[i] <- est_sim3$coefficients["is_treated"]
}

# True treatment effects
delta1 <- 2
delta2 <- mean(seq(from = 0.3, by = 0.3, length.out = length(1998:2015)))
delta3 <- 3
```

2.1.2 Plot (Arisa Itagaki)

Here is a description of the plot that is common through all of them.

Panel (a) plots the average of the outcomes for each treatment group. The background area shows the maximum and minimum ranges of the outcome paths.

Panel (b) plots the distribution of treatment effect estimates $\widehat{\delta^{DD}}$ from 500 Monte Carlo simulations of our three different data generating processes. The red vertical line indicates the true treatment effect.

```r
df_s_sim1 <- df_sim1 |>
  mutate(treated_1998 = if_else(treated_1998 == 1, "T", "C")) |>
  group_by(treated_1998, year) |>
  summarise(heikin = mean(y), saidai = max(y), saisyou = min(y))

df_s_sim2 <- df_sim2 |>
  mutate(treated_1998 = if_else(treated_1998 == 1, "T", "C")) |>
  group_by(treated_1998, year) |>
  summarise(heikin = mean(y), saidai = max(y), saisyou = min(y))

df_s_sim3 <- df_sim3 |>
  group_by(treated_year, year) |>
  summarise(heikin = mean(y), saidai = max(y), saisyou = min(y))


fig1_1 <- df_s_sim1 |>
  ggplot() +
  theme_classic() +
  geom_hline(yintercept = c(-4, -2, 0, 2, 4, 6), linetype = "dotted", color = "gray70") +
  geom_ribbon(aes(x = year, ymax = saidai, ymin = saisyou, fill = factor(treated_1998)),
              alpha = 0.1) +
  geom_line(aes(x = year, y = heikin, color = factor(treated_1998))) +
  scale_y_continuous(limits = c(-4, 6), breaks = seq(-4, 6, by = 2)) +
  labs(title = "Simulation 1",
       subtitle = expression(paste("Not Staggered + Constant ", tau)),
       y = "Value",
       x = NULL) +
  geom_vline(xintercept = 1997.5, linetype = "dashed", color = "black") +
  theme(legend.position = "bottom", legend.title = element_blank(),
        plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        aspect.ratio = 1)

fig2_1 <- df_s_sim2 |>
```

```r
  ggplot() +
  theme_classic() +
  geom_hline(yintercept = c(-5, 0, 5, 10, 15), linetype = "dotted", color = "gray70") +
  geom_ribbon(aes(x = year, ymax = saidai, ymin = saisyou, fill = factor(treated_1998)),
              alpha = 0.1) +
  geom_line(aes(x = year, y = heikin, color = factor(treated_1998))) +
  labs(title = "Simulation 2",
       subtitle = expression(paste("Not Staggered + Dynamic ", tau)),
       y = NULL,
       x = "Year") +
  scale_y_continuous(breaks = seq(-5, 15, by = 5), labels = seq(-5, 15, by = 5)) +
  geom_vline(xintercept = 1997.5, linetype = "dashed", color = "black") +
  theme(legend.position = "bottom", legend.title = element_blank(),
        plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        aspect.ratio = 1)

fig3_1 <- df_s_sim3 |>
  ggplot() +
  theme_classic() +
  geom_hline(yintercept = c(-3, 0, 3, 6), linetype = "dotted", color = "gray70") +
  geom_ribbon(aes(x = year, ymax = saidai, ymin = saisyou, fill = factor(treated_year)),
              alpha = 0.1) +
  geom_line(aes(x = year, y = heikin, color = factor(treated_year))) +
  labs(title = "Simulation 3",
       subtitle = expression(paste("Staggered + Constant/Equal ", tau)),
       y = NULL,
       x = NULL) +
  scale_y_continuous(breaks = seq(-3, 6, by = 3), labels = seq(-3, 6, by = 3)) +
  geom_vline(xintercept = c(1988.5, 1997.5, 2006.5), linetype = "dashed", color = "black") +
  theme(legend.position = "bottom", legend.title = element_blank(),
        plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        aspect.ratio = 1)
```

```r
fig1_2 <- mcSim1to3_500runs |>
  ggplot(aes(x = sim1)) +
  theme_classic() +
  geom_hline(yintercept = c(0, 10, 20, 30, 40), linetype = "dotted", color = "gray70") +
  geom_density(fill = "blue",alpha = 0.2) +
  labs(title = "Simulation 1",
       subtitle = expression(paste("Not Staggered + Constant ", tau)),
       y = "Density",
       x = NULL) +
  scale_y_continuous(breaks = seq(0, 40, by = 10), labels = seq(0, 40, by = 10)) +
  geom_vline(xintercept = delta1, linetype = "dashed", color = "red") +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5))

fig2_2 <- mcSim1to3_500runs |>
  ggplot(aes(x = sim2)) +
  theme_classic() +
```

```
  geom_hline(yintercept = c(0, 5, 10, 15, 20), linetype = "dotted", color = "gray70") +
  geom_density(fill = "blue",alpha = 0.2) +
  labs(title = "Simulation 2",
       subtitle = expression(paste("Not Staggered + Dynamic ", tau)),
       y = NULL,
       x = expression(widehat(delta^{DD}))) +
  scale_y_continuous(breaks = seq(0, 20, by = 5), labels = seq(0, 20, by = 5)) +
  geom_vline(xintercept = delta2, linetype = "dashed", color = "red") +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5))


fig3_2 <- mcSim1to3_500runs |>
  ggplot(aes(x = sim3)) +
  theme_classic() +
  geom_hline(yintercept = c(0, 10, 20, 30, 40), linetype = "dotted", color = "gray70") +
  geom_density(fill = "blue",alpha = 0.2) +
  labs(title = "Simulation 3",
       subtitle = expression(paste("Staggered + Constant/Equal ", tau)),
       y = NULL,
       x = NULL) +
  geom_vline(xintercept = delta3, linetype = "dashed", color = "red") +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5))
```

```
##########
# Yosuke Abe
fig1_trends <- fig1_1 + fig2_1 + fig3_1
fig1_dens <- fig1_2 + fig2_2 + fig3_2

ggsave(here("out/fig_final/fig1_trends.png"), plot = fig1_trends,
       width = 360, height = 160, units = "mm")
ggsave(here("out/fig_final/fig1_dens.png"), plot = fig1_dens,
       width = 360, height = 160, units = "mm")

knitr::include_graphics(here("out/fig_final/fig1_trends.png"))
knitr::include_graphics(here("out/fig_final/fig1_dens.png"))
##########
```

Simulation 1, 2, 3 (TWFE DiD Estimates Under Uniform Treatment Timing or Treatment Effect Homogeneity) shows that the true treatment effect is correctly estimated.

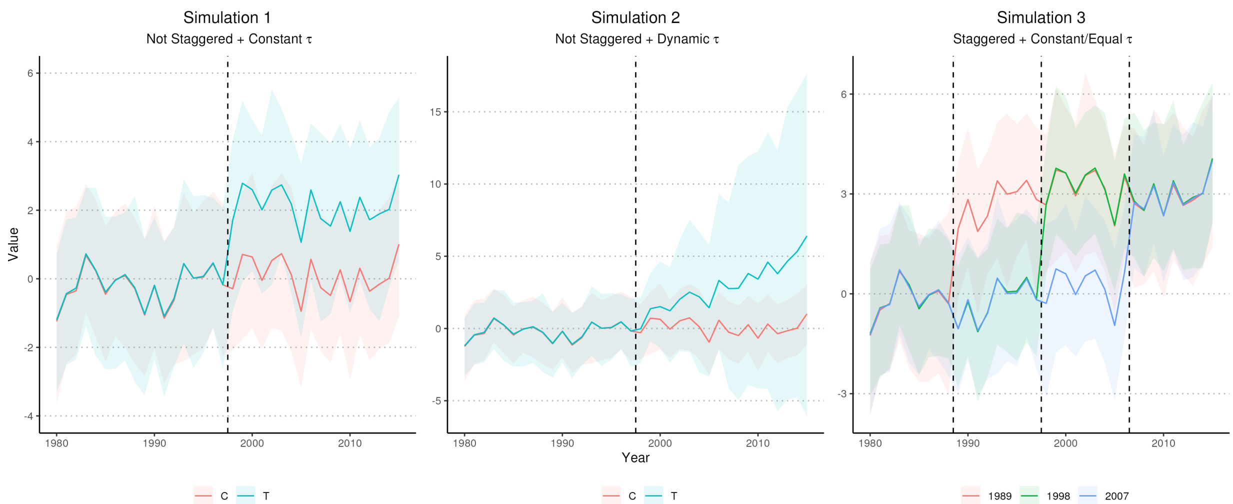
2.2 Plot of Simulation 4, 5, and 6

2.2.1 Data preperation (Yosuke Abe)

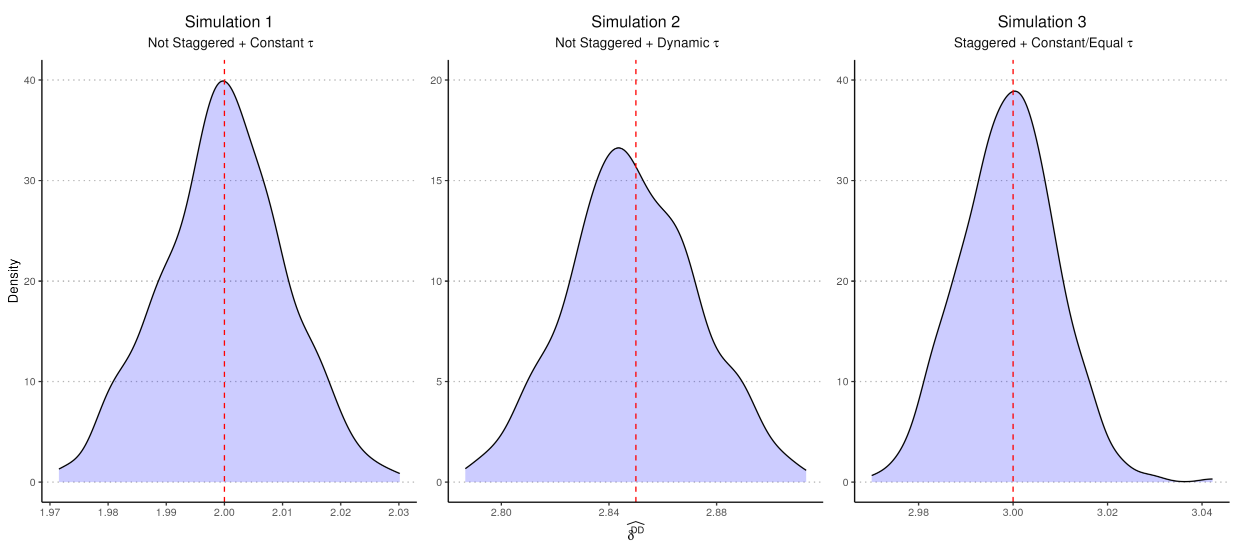
In Simulations 4, 5, and 6, the basic data is the same as in Simulations 1 ~ 3, but each group receives treatments at three different timings. What differs in each simulation is the treatment effect. In Simulation 4, each treatment is uniform (with averages of 5, 3, and 1, respectively, and a standard deviation of 0.2 for all). In Simulation 5, dynamic treatments similar to Simulation 2 are applied, but their effects are equal. Simulation 6 is similar to Simulation 5, but the group treated first increases by an average of 0.5 each year,

Figure 1: Simulation: TWFE DiD Estimates Under Uniform Treatment Timing or Treatment Efect Homogeneity

(a) Trends in Outcome Path



(b) TWFE DiD Estimates on Simulated Data



the second group by 0.3, and the last group by 0.1, with a standard deviation of 0.2 for all. The coefficients estimated using the TWFE model, as in Section 2.1, are stored in a dataframe and used for plotting."

```
set.seed(57)

mcSim4to6_500runs <- tibble(
  "sim4" = numeric(500),
  "sim5" = numeric(500),
  "sim6" = numeric(500)
)

for (i in 1:500) {
  n_firms <- 1000
  n_states <- 50
  T <- 36

  firm_id <- 1:n_firms
  state_id <- sample(n_states, size = n_firms, replace = TRUE)
  year <- 1980:2015

  fe_firm <- rnorm(n_firms, mean = 0, sd = .5)
  fe_year <- rnorm(T, mean = 0, sd = .5)
  error <- rnorm(n_firms * T, mean = 0, sd = .5)

  # Staggered
  treated_year <- sample(c(1989, 1998, 2007), size = n_firms,
                         replace = TRUE, prob = c(17/50, 17/50, 16/50))

  df_sim4 <- tibble(
    firm_id = rep(firm_id, each = T),
    state_id = rep(state_id, each = T),
    year = rep(year, times = n_firms),
    fe_firm = rep(fe_firm, each = T),
    fe_year = rep(fe_year, times = n_firms),
    error = error,
    treated_year = rep(treated_year, each = T),
    is_treated = case_when(
      treated_year == 1989 & year >= 1989 ~ 1,
      treated_year == 1998 & year >= 1998 ~ 1,
      treated_year == 2007 & year >= 2007 ~ 1,
      TRUE ~ 0
    ),
    y = case_when(
      treated_year == 1989 & is_treated == 1 ~
        rnorm(n_firms * T, mean = 5, sd = .2) + fe_firm + fe_year + error,
      treated_year == 1998 & is_treated == 1 ~
        rnorm(n_firms * T, mean = 3, sd = .2) + fe_firm + fe_year + error,
      treated_year == 2007 & is_treated == 1 ~
        rnorm(n_firms * T, mean = 1, sd = .2) + fe_firm + fe_year + error,
      .default = fe_firm + fe_year + error
    )
  )
```

```r
df_sim5 <- tibble(
  firm_id = rep(firm_id, each = T),
  state_id = rep(state_id, each = T),
  year = rep(year, times = n_firms),
  fe_firm = rep(fe_firm, each = T),
  fe_year = rep(fe_year, times = n_firms),
  error = error,
  treated_year = rep(treated_year, each = T),
  is_treated = case_when(
    treated_year == 1989 & year >= 1989 ~ 1,
    treated_year == 1998 & year >= 1998 ~ 1,
    treated_year == 2007 & year >= 2007 ~ 1,
    TRUE ~ 0
  ),
  y = case_when(
    treated_year == 1989 & is_treated == 1 ~
      rnorm(n_firms * T, mean = .3, sd = .2) * (year - 1988) + fe_firm + fe_year + error,
    treated_year == 1998 & is_treated == 1 ~
      rnorm(n_firms * T, mean = .3, sd = .2) * (year - 1997) + fe_firm + fe_year + error,
    treated_year == 2007 & is_treated == 1 ~
      rnorm(n_firms * T, mean = .3, sd = .2) * (year - 2006) + fe_firm + fe_year + error,
    .default = fe_firm + fe_year + error
  )
)

df_sim6 <- tibble(
  firm_id = rep(firm_id, each = T),
  state_id = rep(state_id, each = T),
  year = rep(year, times = n_firms),
  fe_firm = rep(fe_firm, each = T),
  fe_year = rep(fe_year, times = n_firms),
  error = error,
  treated_year = rep(treated_year, each = T),
  is_treated = case_when(
    treated_year == 1989 & year >= 1989 ~ 1,
    treated_year == 1998 & year >= 1998 ~ 1,
    treated_year == 2007 & year >= 2007 ~ 1,
    TRUE ~ 0
  ),
  y = case_when(
    treated_year == 1989 & is_treated == 1 ~
      rnorm(n_firms * T, mean = .5, sd = .2) * (year - 1988) + fe_firm + fe_year + error,
    treated_year == 1998 & is_treated == 1 ~
      rnorm(n_firms * T, mean = .3, sd = .2) * (year - 1997) + fe_firm + fe_year + error,
    treated_year == 2007 & is_treated == 1 ~
      rnorm(n_firms * T, mean = .1, sd = .2) * (year - 2006) + fe_firm + fe_year + error,
    .default = fe_firm + fe_year + error
  )
)

est_sim4 <- df_sim4 |>
  feols(y ~ is_treated | firm_id + year, cluster = "state_id")
est_sim5 <- df_sim5 |>
```

```
    feols(y ~ is_treated | firm_id + year, cluster = "state_id")
  est_sim6 <- df_sim6 |>
    feols(y ~ is_treated | firm_id + year, cluster = "state_id")

  mcSim4to6_500runs$sim4[i] <- est_sim4$coefficients["is_treated"]
  mcSim4to6_500runs$sim5[i] <- est_sim5$coefficients["is_treated"]
  mcSim4to6_500runs$sim6[i] <- est_sim6$coefficients["is_treated"]
}

# True treatment effects
delta4 <- 5 * 17/50 + 3 * 17/50 + 1 * 16/50
delta5 <- (17/50) * mean(seq(from = 0.3, by = 0.3, length.out = length(1989:2015))) +
          (17/50) * mean(seq(from = 0.3, by = 0.3, length.out = length(1998:2015))) +
          (16/50) * mean(seq(from = 0.3, by = 0.3, length.out = length(2007:2015)))
delta6 <- (17/50) * mean(seq(from = 0.5, by = 0.5, length.out = length(1989:2015))) +
          (17/50) * mean(seq(from = 0.3, by = 0.3, length.out = length(1998:2015))) +
          (16/50) * mean(seq(from = 0.1, by = 0.1, length.out = length(2007:2015)))
```

2.2.2 Plot (Arisa Itagaki)

```
df_s_sim4 <- df_sim4 |>
  group_by(treated_year, year) |>
  summarise(heikin = mean(y), saidai = max(y), saisyou = min(y))

df_s_sim5 <- df_sim5 |>
  group_by(treated_year, year) |>
  summarise(heikin = mean(y), saidai = max(y), saisyou = min(y))

df_s_sim6 <- df_sim6 |>
  group_by(treated_year, year) |>
  summarise(heikin = mean(y), saidai = max(y), saisyou = min(y))


fig4_1 <- df_s_sim4 |>
  ggplot() +
  theme_classic() +
  geom_hline(yintercept = c(-3, 0, 3, 6, 9), linetype = "dotted", color = "gray70") +
  geom_ribbon(aes(x = year, ymax = saidai, ymin = saisyou, fill = factor(treated_year)),
              alpha = 0.1) +
  geom_line(aes(x = year, y = heikin, color = factor(treated_year))) +
  labs(title = "Simulation 4",
       subtitle = expression(paste("Staggered + Constant/Unequal ", tau)),
       y = "Value",
       x = NULL) +
  geom_vline(xintercept = c(1988.5, 1997.5, 2006.5), linetype = "dashed", color = "black") +
  scale_y_continuous(breaks = seq(-3, 9, by = 3), labels = seq(-3, 9, by = 3)) +
  theme(legend.position = "bottom", legend.title = element_blank(),
        plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        aspect.ratio = 1)
```

```r
fig5_1 <- df_s_sim5 |>
  ggplot() +
  theme_classic() +
  geom_hline(yintercept = seq(-10, 25, by = 5), linetype = "dotted", color = "gray70") +
  geom_ribbon(aes(x = year, ymax = saidai, ymin = saisyou, fill = factor(treated_year)),
              alpha = 0.1) +
  geom_line(aes(x = year, y = heikin, color = factor(treated_year))) +
  labs(title = "Simulation 5",
       subtitle = expression(paste("Staggered + Dynamic/Equal ", tau)),
       y = NULL,
       x = "Year") +
  geom_vline(xintercept = c(1988.5, 1997.5, 2006.5), linetype = "dashed", color = "black") +
  scale_y_continuous(breaks = seq(-10, 25, by = 5), labels = seq(-10, 25, by = 5)) +
  theme(legend.position = "bottom", legend.title = element_blank(),
        plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        aspect.ratio = 1)

fig6_1 <- df_s_sim6 |>
  ggplot() +
  theme_classic() +
  geom_hline(yintercept = seq(-5, 30, by=5), linetype = "dotted", color = "gray70") +
  geom_ribbon(aes(x = year, ymax = saidai, ymin = saisyou, fill = factor(treated_year)),
              alpha = 0.1) +
  geom_line(aes(x = year, y = heikin, color = factor(treated_year))) +
  labs(title = "Simulation 6",
       subtitle = expression(paste("Staggered + Dynamic/Unequal ", tau)),
       y = NULL,
       x = NULL) +
  geom_vline(xintercept = c(1988.5, 1997.5, 2006.5), linetype = "dashed", color = "black") +
  scale_y_continuous(breaks = seq(-5, 30, by = 5), labels = seq(-5, 30, by = 5)) +
  theme(legend.position = "bottom", legend.title = element_blank(),
        plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        aspect.ratio = 1)
```

```r
fig4_2 <- mcSim4to6_500runs |>
  ggplot(aes(x = sim4)) +
  theme_classic() +
  geom_hline(yintercept = seq(0, 25, by = 5), linetype = "dotted", color = "gray70") +
  geom_density(fill = "blue", alpha = 0.2) +
  labs(title = "Simulation 4",
       subtitle = expression(paste("Staggered + Constant/Unequal ", tau)),
       y = "Density",
       x = NULL) +
  geom_vline(xintercept = delta4, linetype = "dashed", color = "red") +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        aspect.ratio = 1)

fig5_2 <- mcSim4to6_500runs |>
  ggplot(aes(x = sim5)) +
```

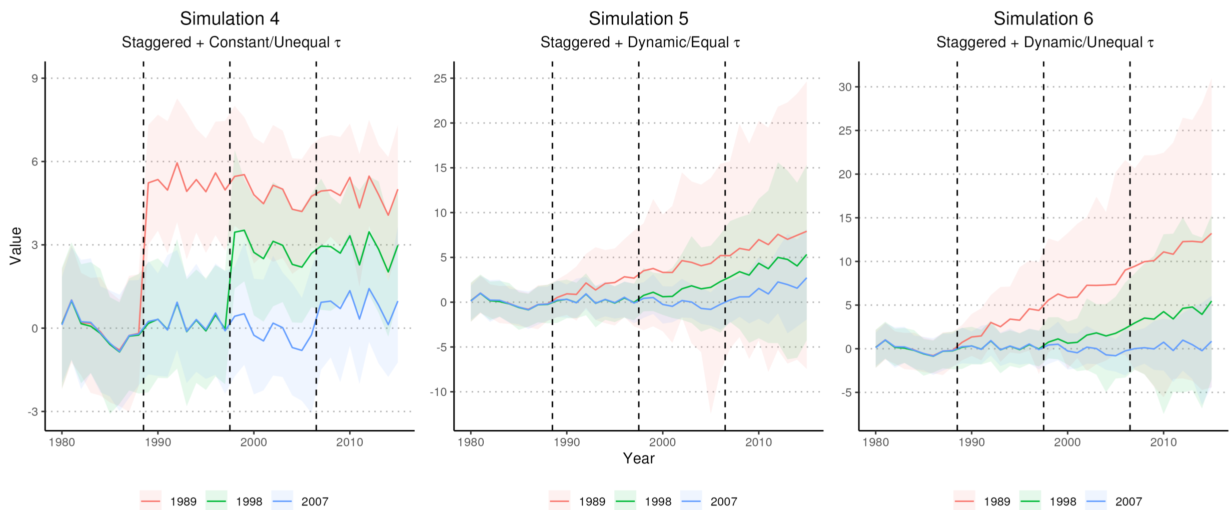
```r
  theme_classic() +
  geom_hline(yintercept = c(0, 3, 6, 9, 12),
             linetype = "dotted", color = "gray70") +
  geom_density(fill = "blue", alpha = 0.2) +
  scale_y_continuous(breaks = seq(0, 12, by = 3), labels = seq(0, 12, by = 3)) +
  labs(title = "Simulation 5",
       subtitle = expression(paste("Staggered + Dynamic/Equal ", tau)),
       y = NULL,
       x = expression(widehat(delta^{DD}))) +
  geom_vline(xintercept = delta5, linetype = "dashed", color = "red") +
  geom_vline(xintercept = 0, linetype = "dashed", color = "black")+
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        aspect.ratio = 1)

fig6_2 <- mcSim4to6_500runs |>
  ggplot(aes(x = sim6)) +
  theme_classic() +
  geom_hline(yintercept = c(0, 3, 6, 9),
             linetype = "dotted", color = "gray70") +
  geom_density(fill = "blue", alpha = 0.2) +
  labs(title = "Simulation 6",
       subtitle = expression(paste("Staggered + Dynamic/Unequal ", tau)),
       y = NULL,
       x = NULL) +
  geom_vline(xintercept = delta6, linetype = "dashed", color = "red") +
  geom_vline(xintercept = 0, linetype = "dashed", color = "black")+
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        aspect.ratio = 1)
```

```r
##########
# Yosuke Abe
fig2_trends <- fig4_1 + fig5_1 + fig6_1
fig2_dens <- fig4_2 + fig5_2 + fig6_2

ggsave(here("out/fig_final/fig2_trends.png"), plot = fig2_trends,
       width = 320, height = 160, units = "mm")
ggsave(here("out/fig_final/fig2_dens.png"), plot = fig2_dens,
       width = 320, height = 160, units = "mm")

knitr::include_graphics(here("out/fig_final/fig2_trends.png"))
knitr::include_graphics(here("out/fig_final/fig2_dens.png"))
##########
```

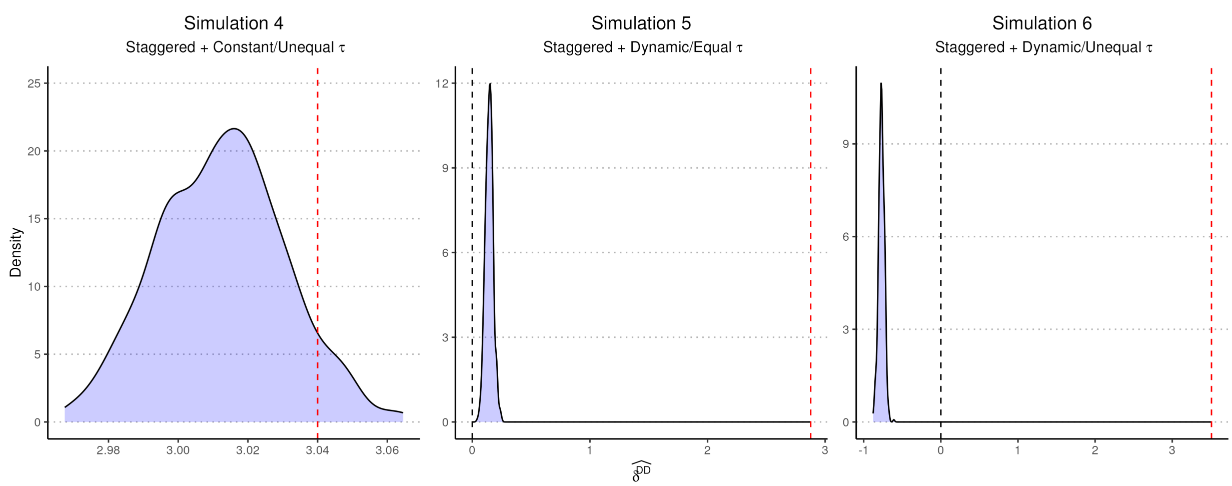
Simulation 4, 5, 6 (TWFE DiD Estimates Under Staggered Timing and Treatment Effect Heterogeneity) showed that the treatment effect estimates were biased, underestimated and incorrectly estimated. In particular, the sign is reversed in 6.

Figure 2: Simulation: TWFE DiD Estimates Under Staggered Timing and Treatment Effect Heterogeneity

(a) Trends in Outcome Path



(b) TWFE DiD Estimates on Simulated Data



3 Extension

As an extension, we will conduct the simulations performed so far using the Sun and Abraham estimator and compare them with the standard TWFE model. Additionally, since the Sun and Abraham estimator uses only units that never received treatment as the effective comparison units, we will exclude the group treated in 2007 and conduct the estimation using data up to 2006 (27 years). In this section, we use `fixest::sunab()` function.

3.1 Data preparation (Yosuke Abe)

```
set.seed(57)

mcSim4to6_sunab <- tibble(
  "twfe_sim4" = numeric(500),
  "sunab_sim4" = numeric(500),
  "twfe_sim5" = numeric(500),
  "sunab_sim5" = numeric(500),
  "twfe_sim6" = numeric(500),
  "sunab_sim6" = numeric(500)
)

for (i in 1:500) {
  n_firms <- 1000
  n_states <- 50
  T <- 27

  firm_id <- 1:n_firms
  state_id <- sample(n_states, size = n_firms, replace = TRUE)
  year <- 1980:2006

  fe_firm <- rnorm(n_firms, mean = 0, sd = .5)
  fe_year <- rnorm(T, mean = 0, sd = .5)
  error <- rnorm(n_firms * T, mean = 0, sd = .5)

  # Staggered
  treated_year <- sample(c(1989, 1998, 2007), size = n_firms,
                         replace = TRUE, prob = c(17/50, 17/50, 16/50))

  df_sim4_2trt <- tibble(
    firm_id = rep(firm_id, each = T),
    state_id = rep(state_id, each = T),
    year = rep(year, times = n_firms),
    fe_firm = rep(fe_firm, each = T),
    fe_year = rep(fe_year, times = n_firms),
    error = error,
    treated_year = rep(treated_year, each = T),
    is_treated = case_when(
      treated_year == 1989 & year >= 1989 ~ 1,
      treated_year == 1998 & year >= 1998 ~ 1,
      treated_year == 2007 & year >= 2007 ~ 1,
      TRUE ~ 0
    ),
```

```r
  y = case_when(
    treated_year == 1989 & is_treated == 1 ~
      rnorm(n_firms * T, mean = 5, sd = .2) + fe_firm + fe_year + error,
    treated_year == 1998 & is_treated == 1 ~
      rnorm(n_firms * T, mean = 3, sd = .2) + fe_firm + fe_year + error,
    treated_year == 2007 & is_treated == 1 ~
      rnorm(n_firms * T, mean = 1, sd = .2) + fe_firm + fe_year + error,
    .default = fe_firm + fe_year + error
  )
)

df_sim5_2trt <- tibble(
  firm_id = rep(firm_id, each = T),
  state_id = rep(state_id, each = T),
  year = rep(year, times = n_firms),
  fe_firm = rep(fe_firm, each = T),
  fe_year = rep(fe_year, times = n_firms),
  error = error,
  treated_year = rep(treated_year, each = T),
  is_treated = case_when(
    treated_year == 1989 & year >= 1989 ~ 1,
    treated_year == 1998 & year >= 1998 ~ 1,
    treated_year == 2007 & year >= 2007 ~ 1,
    TRUE ~ 0
  ),
  y = case_when(
    treated_year == 1989 & is_treated == 1 ~
      rnorm(n_firms * T, mean = .3, sd = .2) * (year - 1988) + fe_firm + fe_year + error,
    treated_year == 1998 & is_treated == 1 ~
      rnorm(n_firms * T, mean = .3, sd = .2) * (year - 1997) + fe_firm + fe_year + error,
    treated_year == 2007 & is_treated == 1 ~
      rnorm(n_firms * T, mean = .3, sd = .2) * (year - 2006) + fe_firm + fe_year + error,
    .default = fe_firm + fe_year + error
  )
)

df_sim6_2trt <- tibble(
  firm_id = rep(firm_id, each = T),
  state_id = rep(state_id, each = T),
  year = rep(year, times = n_firms),
  fe_firm = rep(fe_firm, each = T),
  fe_year = rep(fe_year, times = n_firms),
  error = error,
  treated_year = rep(treated_year, each = T),
  is_treated = case_when(
    treated_year == 1989 & year >= 1989 ~ 1,
    treated_year == 1998 & year >= 1998 ~ 1,
    treated_year == 2007 & year >= 2007 ~ 1,
    TRUE ~ 0
  ),
  y = case_when(
    treated_year == 1989 & is_treated == 1 ~
      rnorm(n_firms * T, mean = .5, sd = .2) * (year - 1988) + fe_firm + fe_year + error,
```

```r
      treated_year == 1998 & is_treated == 1 ~
        rnorm(n_firms * T, mean = .3, sd = .2) * (year - 1997) + fe_firm + fe_year + error,
      treated_year == 2007 & is_treated == 1 ~
        rnorm(n_firms * T, mean = .1, sd = .2) * (year - 2006) + fe_firm + fe_year + error,
      .default = fe_firm + fe_year + error
    )
  )

  # without sunab()
  est_sim4_twfe <- df_sim4_2trt |>
    feols(y ~ is_treated | firm_id + year, cluster = "state_id")

  est_sim5_twfe <- df_sim5_2trt |>
    feols(y ~ is_treated | firm_id + year, cluster = "state_id")

  est_sim6_twfe <- df_sim6_2trt |>
    feols(y ~ is_treated | firm_id + year, cluster = "state_id")

  # with sunab()
  est_sim4_sunab <- df_sim4_2trt |>
    feols(y ~ sunab(treated_year, year) | firm_id + year, cluster = "state_id")

  est_sim5_sunab <- df_sim5_2trt |>
    feols(y ~ sunab(treated_year, year) | firm_id + year, cluster = "state_id")

  est_sim6_sunab <- df_sim6_2trt |>
    feols(y ~ sunab(treated_year, year) | firm_id + year, cluster = "state_id")

  mcSim4to6_sunab$twfe_sim4[i] <- est_sim4_twfe$coefficients["is_treated"]
  mcSim4to6_sunab$twfe_sim5[i] <- est_sim5_twfe$coefficients["is_treated"]
  mcSim4to6_sunab$twfe_sim6[i] <- est_sim6_twfe$coefficients["is_treated"]
  mcSim4to6_sunab$sunab_sim4[i] <- summary(est_sim4_sunab, agg = "ATT")$coeftable[[1]]
  mcSim4to6_sunab$sunab_sim5[i] <- summary(est_sim5_sunab, agg = "ATT")$coeftable[[1]]
  mcSim4to6_sunab$sunab_sim6[i] <- summary(est_sim6_sunab, agg = "ATT")$coeftable[[1]]
}

# True treatment effects
delta4_sunab <-
  5 * length(1989:2006) / length(1980:2006) +
  3 * length(1998:2006) / length(1980:2006)
delta5_sunab <-
  mean(c(0.3, 0.3 * length(1989:2006))) * length(1989:2006) / length(1980:2006) +
  mean(c(0.3, 0.3 * length(1998:2006))) * length(1998:2006) / length(1980:2006)
delta6_sunab <-
  mean(c(0.5, 0.5 * length(1989:2006))) * length(1989:2006) / length(1980:2006) +
  mean(c(0.3, 0.3 * length(1998:2006))) * length(1998:2006) / length(1980:2006)
```

3.2 Plot (Arisa Itagaki)

```r
df_sunab_sim4 <- df_sim4_2trt |>
  group_by(treated_year, year) |>
  summarise(heikin = mean(y), saidai = max(y), saisyou = min(y))

df_sunab_sim5 <- df_sim5_2trt |>
  group_by(treated_year, year) |>
  summarise(heikin = mean(y), saidai = max(y), saisyou = min(y))

df_sunab_sim6 <- df_sim6_2trt |>
  group_by(treated_year, year) |>
  summarise(heikin = mean(y), saidai = max(y), saisyou = min(y))


fig4_2trt_trend <- df_sunab_sim4 |>
  ggplot() +
  theme_classic() +
  geom_hline(yintercept = c(-3, 0, 3, 6, 9), linetype = "dotted", color = "gray70") +
  geom_ribbon(aes(x = year, ymax = saidai, ymin = saisyou, fill = factor(treated_year)),
              alpha = 0.1) +
  geom_line(aes(x = year, y = heikin, color = factor(treated_year))) +
  labs(title = "Simulation 4 (2 Treatment Timings)",
       subtitle = expression(paste("Staggered + Constant/Unequal ", tau)),
       y = "Value",
       x = NULL) +
  geom_vline(xintercept = c(1988.5, 1997.5), linetype = "dashed", color = "black") +
  scale_y_continuous(breaks = seq(-3, 9, by = 3), labels = seq(-3, 9, by = 3)) +
  theme(legend.position = "bottom", legend.title = element_blank(),
        plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        aspect.ratio = 1)

fig5_2trt_trend <- df_sunab_sim5 |>
  ggplot() +
  theme_classic() +
  geom_hline(yintercept = seq(-5, 20, by = 5), linetype = "dotted", color = "gray70") +
  geom_ribbon(aes(x = year, ymax = saidai, ymin = saisyou, fill = factor(treated_year)),
              alpha = 0.1) +
  geom_line(aes(x = year, y = heikin, color = factor(treated_year))) +
  labs(title = "Simulation 5 (2 Treatment Timings)",
       subtitle = expression(paste("Staggered + Dynamic/Equal ", tau)),
       y = NULL,
       x = "Year") +
  geom_vline(xintercept = c(1988.5, 1997.5), linetype = "dashed", color = "black") +
  scale_y_continuous(breaks = seq(-5, 20, by = 5), labels = seq(-5, 20, by = 5)) +
  theme(legend.position = "bottom", legend.title = element_blank(),
        plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        aspect.ratio = 1)

fig6_2trt_trend <- df_sunab_sim6 |>
```

```r
  ggplot() +
  theme_classic() +
  geom_hline(yintercept = seq(-5, 20, by=5), linetype = "dotted", color = "gray70") +
  geom_ribbon(aes(x = year, ymax = saidai, ymin = saisyou, fill = factor(treated_year)),
              alpha = 0.1) +
  geom_line(aes(x = year, y = heikin, color = factor(treated_year))) +
  labs(title = "Simulation 6 (2 Treatment Timings)",
       subtitle = expression(paste("Staggered + Dynamic/Unequal ", tau)),
       y = NULL,
       x = NULL) +
  geom_vline(xintercept = c(1988.5, 1997.5), linetype = "dashed", color = "black") +
  scale_y_continuous(breaks = seq(-5, 20, by = 5), labels = seq(-5, 20, by = 5)) +
  theme(legend.position = "bottom", legend.title = element_blank(),
        plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        aspect.ratio = 1)
```

```r
fig4_sunab <- mcSim4to6_sunab |>
  ggplot() +
  theme_classic() +
  geom_hline(yintercept = c(0, 5, 10, 15), linetype = "dotted", color = "gray70") +
  geom_density(aes(x = twfe_sim4, fill = "Standard TWFE"), alpha = 0.2) +
  geom_density(aes(x = sunab_sim4, fill = "Sun and Abraham"), alpha = 0.2) +
  labs(title = "Simulation 4 (2 Treatment Timings)",
       subtitle = expression(paste("Staggered + Constant/Unequal ", tau)),
       y = "Value",
       x = NULL) +
  geom_vline(xintercept = delta4_sunab, linetype = "dashed", color = "red") +
  scale_x_continuous(limits = c(3.8, 4.6)) +
  theme(legend.position = "bottom", legend.title = element_blank(),
        plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        aspect.ratio = 1)


fig5_sunab <- mcSim4to6_sunab |>
  ggplot() +
  theme_classic() +
  geom_hline(yintercept = c(0, 3, 6, 9, 12), linetype = "dotted", color = "gray70") +
  geom_density(aes(x = twfe_sim5, fill = "Standard TWFE"), alpha = 0.2) +
  geom_density(aes(x = sunab_sim5, fill = "Sun and Abraham"), alpha = 0.2) +
  labs(title = "Simulation 5 (2 Treatment Timings)",
       subtitle = expression(paste("Staggered + Dynamic/Equal ", tau)),
       y = NULL,
       x = expression(widehat(delta^{DD}))) +
  geom_vline(xintercept = delta5_sunab, linetype = "dashed", color = "red") +
  scale_x_continuous(limits = c(1.2, 2.7)) +
  scale_y_continuous(breaks = seq(0, 12, by = 3), labels = seq(0, 12, by = 3)) +
  theme(legend.position = "bottom", legend.title = element_blank(),
        plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        aspect.ratio = 1)
```

```r
fig6_sunab <- mcSim4to6_sunab |>
  ggplot() +
  theme_classic() +
  geom_hline(yintercept = c(0, 2, 4, 6), linetype = "dotted", color = "gray70") +
  geom_density(aes(x = twfe_sim6, fill = "Standard TWFE"), alpha = 0.2) +
  geom_density(aes(x = sunab_sim6, fill = "Sun and Abraham"), alpha = 0.2) +
  labs(title = "Simulation 6 (2 Treatment Timings)",
       subtitle = expression(paste("Staggered + Dynamic/Unequal ", tau)),
       y = NULL,
       x = NULL) +
  geom_vline(xintercept = delta6_sunab, linetype = "dashed", color = "red") +
  scale_x_continuous(limits = c(1.5, 4.3)) +
  theme(legend.position = "bottom", legend.title = element_blank(),
        plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        aspect.ratio = 1)
```

```r
##########
# Yosuke Abe
fig_ex_trends <- fig4_2trt_trend + fig5_2trt_trend + fig6_2trt_trend
fig_ex_dens <- fig4_sunab + fig5_sunab + fig6_sunab

ggsave(here("out/fig_final/fig_ex_trends.png"), plot = fig_ex_trends,
       width = 320, height = 160, units = "mm")
ggsave(here("out/fig_final/fig_ex_dens.png"), plot = fig_ex_dens,
       width = 320, height = 160, units = "mm")

knitr::include_graphics(here("out/fig_final/fig_ex_trends.png"))
knitr::include_graphics(here("out/fig_final/fig_ex_dens.png"))
##########
```

We find that the Sun and Abraham (2020) Estimator can be used to estimate without bias in situations like simulation 4, 5, 6 (TWFE DiD Estimates Under Staggered Timing and Treatment Effect Heterogeneity).

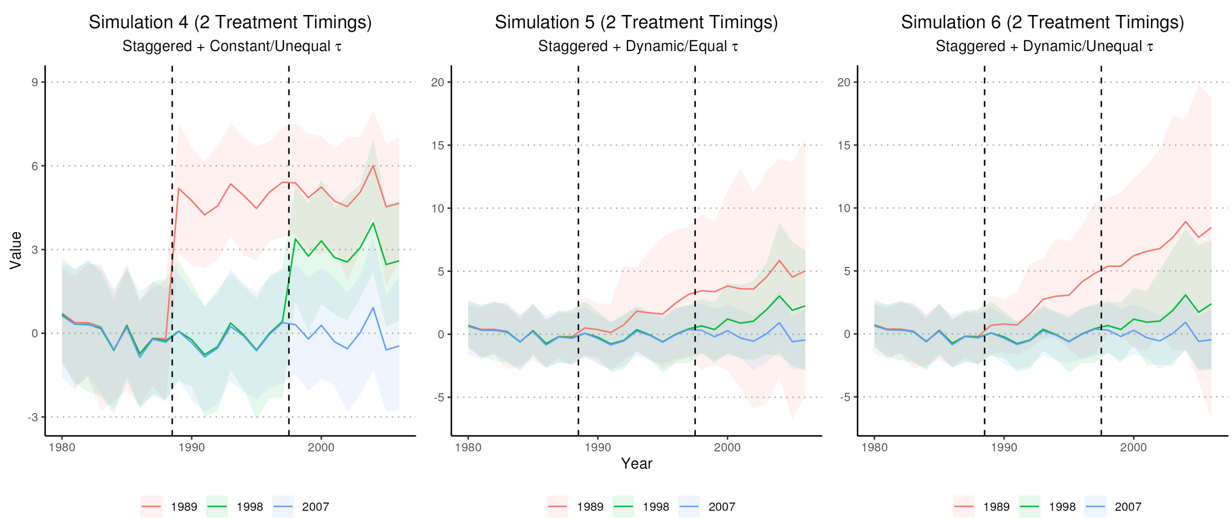
3.3 True Treatment Path (Replication) (Yosuke Abe)

This section is not an extension, but since we calculated the estimates using the Sun and Abraham estimator in the previous section, we will use it to replicate Figure 6 in the paper.

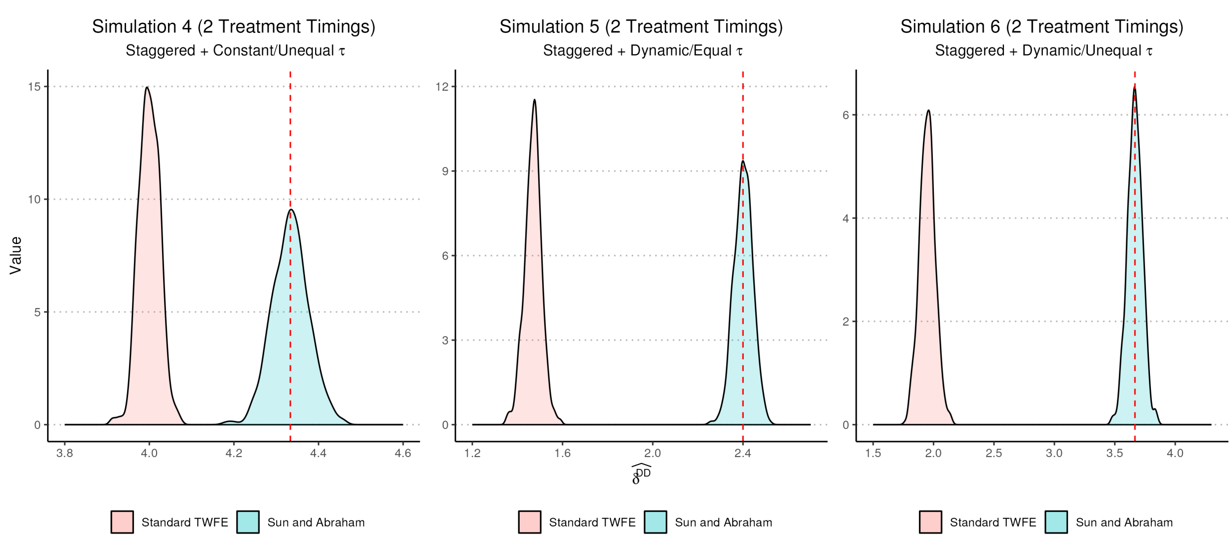
```r
df <- df_sim6_2trt |>
  select(year, treated_year) |>
  mutate(relative_year = year - treated_year) |>
  filter(treated_year != 2007 & relative_year != -1) |>
  summarise(.by = relative_year) |>
  arrange(relative_year)

trt_path <- df |>
  bind_cols(
    tibble(estimate = est_sim6_sunab$coeftable[, "Estimate"]),
    tibble(se = est_sim6_sunab$coeftable[, "Std. Error"])
  ) |>
```

Figure 3: Staggered Timing Analysis: TWFE vs. Sun and Abraham's DiD Approaches in Heterogeneous Treatment Settings

(a) Trends in Outcome Path



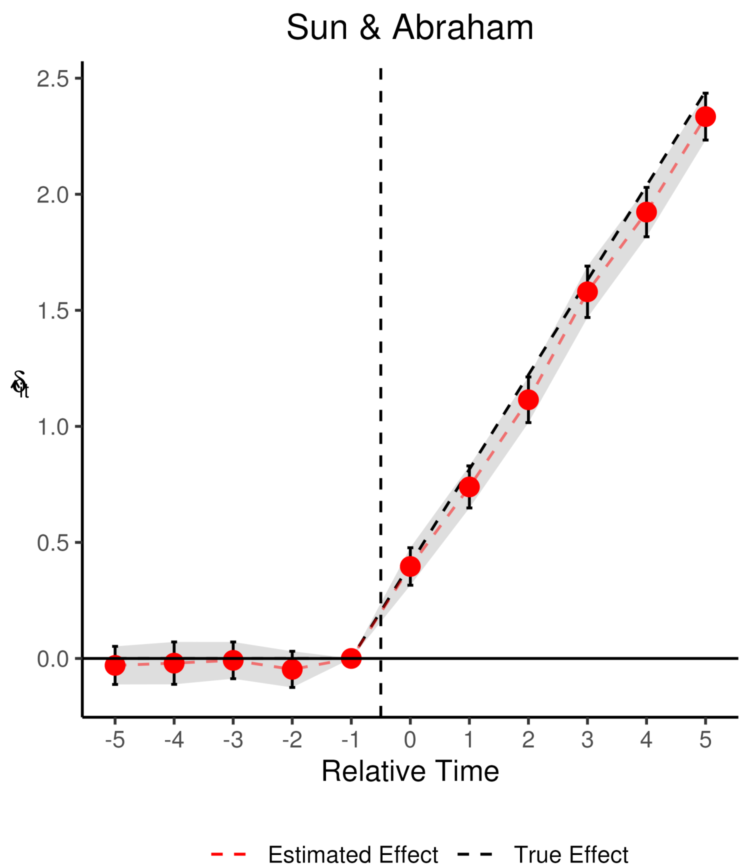
(b) TWFE vs. Sun and Abraham's DiD Estimates on Simulated Data



```r
  bind_rows(tibble(relative_year = -1, estimate = 0, se = 0,
                   est_twfe = 0, se_twfe = 0)) |>
  mutate(
    delta_hat = case_when(
      relative_year <= -1 ~ 0,
      relative_year >= 0 & relative_year <= 5 ~
        delta6_sunab/9 + relative_year * delta6_sunab/9,
      TRUE ~ NA_real_
    ),
    conf_upper = estimate + 1.96 * se,
    conf_lower = estimate - 1.96 * se
  ) |>
  arrange(relative_year)

fig_trt_path <- trt_path |>
  filter(relative_year <= 5 & relative_year >= -5) |>
  ggplot(aes(x = relative_year)) +
  geom_ribbon(aes(ymax = conf_upper, ymin = conf_lower),
              fill = "gray", alpha = 0.5) +
  geom_line(aes(y = delta_hat, color = "True Effect"), linetype = "dashed") +
  geom_errorbar(aes(ymax = conf_upper, ymin = conf_lower), width = 0.1) +
  geom_line(aes(y = estimate, color = "Estimated Effect"),
            linetype = "dashed", alpha = 0.5) +
  geom_point(aes(y = estimate), color = "red", size = 3) +
  geom_vline(xintercept = -0.5, linetype = "dashed") +
  geom_hline(yintercept = 0) +
  scale_x_continuous(breaks = seq(from = -5, to = 5, by = 1)) +
  scale_color_manual(values = c("True Effect" = "black", "Estimated Effect" = "red")) +
  ggtitle("Sun & Abraham") +
  labs(x = "Relative Time", y = expression(widehat(delta)["it"])) +
  theme_classic() +
  theme(plot.title = element_text(hjust = 0.5),
        legend.position = "bottom",
        legend.title = element_blank(),
        axis.title.y = element_text(angle = 360, hjust = 0.5, vjust = 0.5),
        aspect.ratio = 1)

ggsave(here("out/fig_final/fig-ex-path.png"), plot = fig_trt_path,
       dpi = 500, width = 160, height = 120, units = "mm")
```

```r
knitr::include_graphics(here("out/fig_final/fig-ex-path.png"))
```

Figure 4: Robust DiD Method with Staggered Treatment Assignment and Dynamic Treatment Effects



From Figure 4, it is apparent that the Sun and Abraham estimator is able to estimate without bias in each relative year. However, we also see that the True Effect is just barely within the confidence interval. The True Effect is manually calculated considering different treatment effects and the number of treatment years, so there may be some errors, but as the same trend is observed in the paper, it can be said that there is no significant difference.

Additionally, this is a technical issue, but the delta symbol in the y-axis label appears compressed and difficult to read. I thought I had looked into ways to display the formulas correctly, but could not effectively remedy the problem. However, when running the code before outputting to a .png file, it displays correctly without any problems.

References

Baker, Andrew C., David F. Larcker, and Charles C.Y. Wang (Feb. 2021). "How Much Should We Trust Staggered Difference-In-Differences Estimates?" Working paper. URL: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3794018#.