Myanmar Election 2015 vs 2020

Nicholus Tint Zaw

Background Information

Under the military dictatorship for over 60 years, the Military set up an exit -strategy from Burma/Myanmar Politic and implemented it through the Military Sponsered constitution by 2008. As part of its game plan, the Military organized the first election in 2010, and the military backup political party (USDP - Union Solidarity and Development Party) won with a landslide victory. The opponent party, National League for Democracy - NLD (Led by Aung San Su Kyi), boycotted that election and did not run for any parliamentarian candidacy. Because the democratic force believed that the constitution does not guarantee federal democracy and the military-sponsored election was not free and fair. According to the 2008 constitution, the military still controls the hard-power ministry (defense, home-affair, border affairs) and occupies 25% of each category of parliament.

However, NLD ran for election in 2015, which was organized by the Union Election Commission appointed by the Military and won the landslide victory. Therefore, NLD had a chance to rule the semi-civic government from 2016 to 2020. As the election is organized every 5 years, in November 2020, there was a general election, and this time, it was organized by the Union Election Commission appointed by the NLD government. And, NLD did the same result as in 2015 with a landslide victory.

However, on the 1st February 2021, before the first day of the new parliament (elected by Nov 2020 election), coup d'état in Burma/Myanmar began. The main rationale was that there was a lot of election fraud as extensive inclusion errors in the eligible voters' list in the 2020 election.

Motivation

The various independent election observation organization issued the Burma 2020 election observations reports (including embassies), and all agreed that the election was not entirely free and fair. Still, those were not significant enough to change (or affect) the election results. Besides that, no articles investigated military claims on election fraud: significant inclusion error in eligible voters' list. Based on the Lab 2 exercise, I have an idea of how I can apply the difference-in-transport estimator to investigate what the military claimed as election fraud in the Burma 2020 election.

There is an opportunity to get the Burma election data for 2015 and 2020, including the individual candidacy level information on the number of valid votes and the number of eligible voters, and the voter turnout rate at each constituency level.

Challenges points

- parallel trend assumption check
- the availability of control data

Difference-in-Transports Cost Estimation

Using one election dataset as a sample demo exercise, I got the error in applying the "difrans" function in estimation optimal transport cost.

```
load("df_all_pyithu_map.rda")

df_all_pyithu_cleaned <- df_all_pyithu_map %>%
  select(ST, TS, num_eligible_voters, dataset) %>%
  group_by(ST, TS, dataset) %>%
  slice(1)


df_2015 <- df_all_pyithu_cleaned %>%
  filter(dataset == "pyithu_2015") %>%
  group_by(ST) %>%
  summarise(num_eligible_voters = sum(num_eligible_voters))


df_2020 <- df_all_pyithu_cleaned %>%
  filter(dataset == "pyithu_2020") %>%
  group_by(ST) %>%
  summarise(num_eligible_voters = sum(num_eligible_voters))
```

Visualize the 2015 vs 2020 lower house data

```
df_2015_dist <- df_2015 %>% uncount(num_eligible_voters)
df_2020_dist <- df_2020 %>% uncount(num_eligible_voters)

dist <- ggplot() +
  geom_histogram(data = df_2015_dist,
                 stat="count",
                 aes(x = ST,
                     y = ..count../sum(..count..)),
                 fill = "orange", color = "orange", alpha = 0.35) +
  geom_histogram(data = df_2020_dist,
                 stat="count",
                 aes(x = ST,
                     y = ..count../sum(..count..)),
                 fill = "steelblue", color = "steelblue", alpha = 0.35) +
  labs(x = "State and Region", y = "Density - Number of Eligable Voter") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))

dist
```

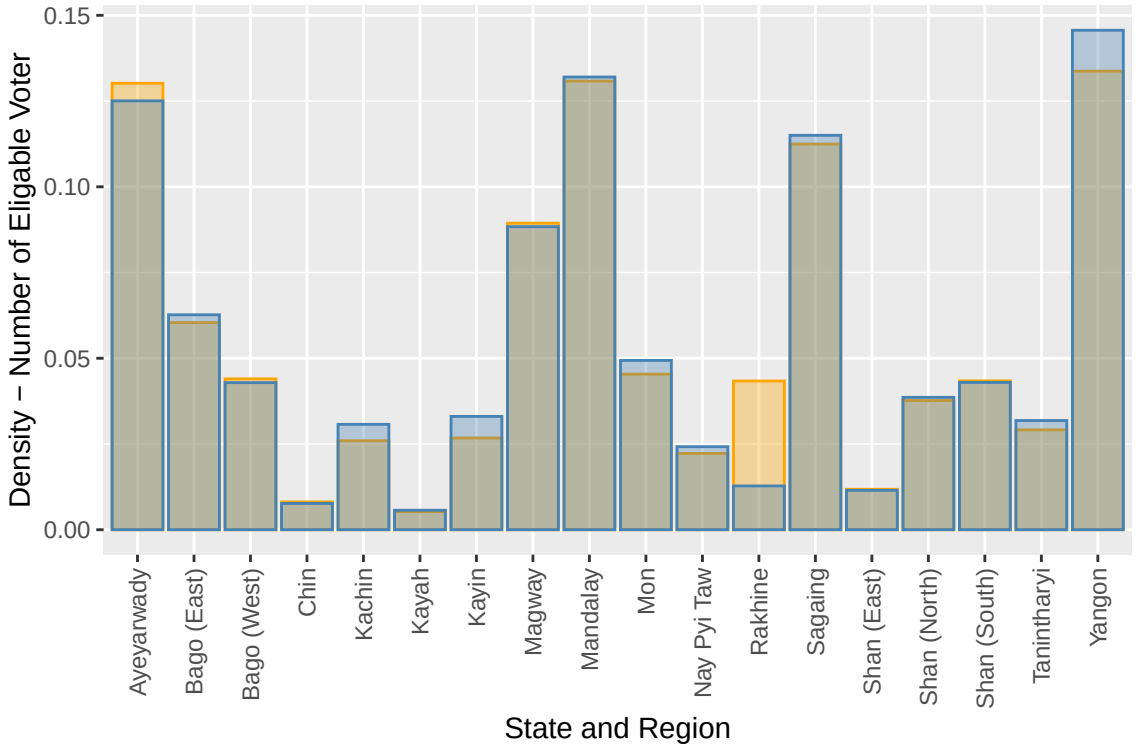

Figure 1: Eligable voter distribution 2015 vs 2020 for lower house

Placebo Distribution

(a) placebo_1

```
set.seed(4487989)

placebo_1 <- data.frame(ST = df_2015$ST,
                        count = rmultinom(n = 1,
                                          size = sum(df_2015$num_eligible_voters),
                                          prob = df_2015$num_eligible_voters))

head(placebo_1)
##            ST   count
## 1  Ayeyarwady 4404908
## 2 Bago (East) 2045241
## 3 Bago (West) 1488455
## 4        Chin  273470
## 5      Kachin  876547
## 6       Kayah  178099
```

(b) placebo_2

```
set.seed(384620)

placebo_2 <- data.frame(ST = df_2015$ST,
                        count = rmultinom(n = 1,
                                          size = sum(df_2020$num_eligible_voters),
                                          prob = df_2015$num_eligible_voters))
```

```
head(placebo_2)
##            ST   count
## 1  Ayeyarwady 4875910
## 2 Bago (East) 2260734
## 3 Bago (West) 1645733
## 4        Chin  303999
## 5      Kachin  971341
## 6       Kayah  197122
```

(c) Compare placebo_1 and placebo_2

```
placebo_1_dist <- placebo_1 %>% uncount(count)
placebo_2_dist <- placebo_2 %>% uncount(count)


ggplot() +
  geom_histogram(data = placebo_1_dist,
                 stat="count",
                 aes(x = ST,
                     y = ..count../sum(..count..)),
                 fill = "orange", color = "orange", alpha = 0.35) +
  geom_histogram(data = placebo_2_dist,
                 stat="count",
                 aes(x = ST,
                     y = ..count../sum(..count..)),
                 fill = "steelblue", color = "steelblue", alpha = 0.35) +
  labs(x = "State and Region", y = "Density - Number of Eligable Voter") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```

Optimal transport cost calculation

```
bandwidths <- c(0)

placebo_at_0 <- diftrans(pre_main = placebo_1,
                         post_main = placebo_2,
                         var = ST,
                         bandwidth_seq = bandwidths)
## ======================================================================
## Error in support[i] - support: non-numeric argument to binary operator


placebo_at_0
## Error in eval(expr, envir, enclos): object 'placebo_at_0' not found
```

(a) Compute the transport cost between the two placebo distributions with different bandwidths

```
bandwidths <- c(0, 500, 10000, 30000, 35000, 40000, 45000, 45150, 45198, 45200, 45500, 46000,
```

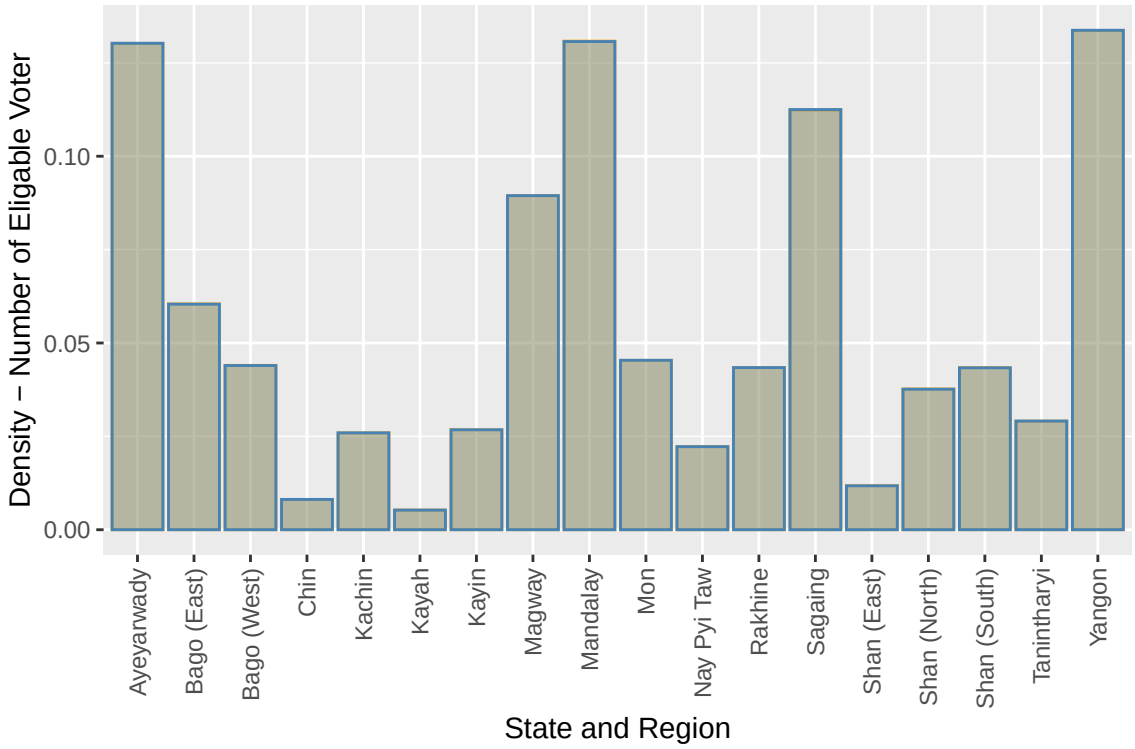

Figure 2: Comparision between placebo 1 vs 2

```
placebo_at_bw <- diftrans(pre_main = placebo_1,
                          post_main = placebo_2,
                          var = ST,
                          bandwidth_seq = bandwidths) %>%
  mutate(cat = "placebo")
## ====
## Error in support[i] - support: non-numeric argument to binary operator


placebo_at_bw
## Error in eval(expr, envir, enclos): object 'placebo_at_bw' not found
```

(b) compute the transport cost between the observed distributions for 2015 and 2015 election eligible voters number

```
df_2015 <- df_2015 %>%
  mutate(count = num_eligible_voters)

df_2020 <- df_2020 %>%
  mutate(count = num_eligible_voters)

emprical_at_bw <- diftrans(pre_main = df_2015,
                          post_main = df_2020,
                          var = ST,
                          bandwidth_seq = bandwidths) %>%
  mutate(cat = "emprical")
```

```
## ====
## Error in support[i] - support: non-numeric argument to binary operator


emprical_at_bw
## Error in eval(expr, envir, enclos): object 'emprical_at_bw' not found
```

(c)

```
df_merged <- rbind(placebo_at_bw, emprical_at_bw)
## Error in rbind(placebo_at_bw, emprical_at_bw): object 'placebo_at_bw' not found

ggplot(df_merged, aes(x = bandwidth, y = main, color = cat)) +
  geom_line() +
  labs(x = "Bandwidths",
       y = "the fraction of \n optimal transport cost",
       color = "between two distributions of") +
  theme(legend.position = "bottom")
## Error in ggplot(df_merged, aes(x = bandwidth, y = main, color = cat)): object 'df_merged'
```

(d) values of d, the placebo cost less than 0.05%

```
placebo_at_bw %>%
  arrange(-main) %>%
  filter(main < 0.0005)
## Error in arrange(., -main): object 'placebo_at_bw' not found
```

(e) The empirical transport cost at lowest value of d

```
emprical_at_bw %>%
  arrange(main) %>%
  filter(bandwidth == 45200)
## Error in arrange(., main): object 'emprical_at_bw' not found
```