Panel Data Solution

Alicia R. Chen

6/2/2021

Exercise 1

```
packages <- c("tidyverse","data.table", "lubridate", 'ggplot2')
lapply(packages, library, character.only = TRUE)
```

```
## [[1]]
##  [1] "forcats"   "stringr"   "dplyr"     "purrr"     "readr"     "tidyr"
##  [7] "tibble"    "ggplot2"   "tidyverse" "stats"     "graphics"  "grDevices"
## [13] "utils"     "datasets"  "methods"   "base"
##
## [[2]]
##  [1] "data.table" "forcats"    "stringr"    "dplyr"      "purrr"
##  [6] "readr"      "tidyr"      "tibble"     "ggplot2"    "tidyverse"
## [11] "stats"      "graphics"   "grDevices"  "utils"      "datasets"
## [16] "methods"    "base"
##
## [[3]]
##  [1] "lubridate"  "data.table" "forcats"    "stringr"    "dplyr"
##  [6] "purrr"      "readr"      "tidyr"      "tibble"     "ggplot2"
## [11] "tidyverse"  "stats"      "graphics"   "grDevices"  "utils"
## [16] "datasets"   "methods"    "base"
##
## [[4]]
##  [1] "lubridate"  "data.table" "forcats"    "stringr"    "dplyr"
##  [6] "purrr"      "readr"      "tidyr"      "tibble"     "ggplot2"
## [11] "tidyverse"  "stats"      "graphics"   "grDevices"  "utils"
## [16] "datasets"   "methods"    "base"
```

```
# load data
df <- fread("./2. Text-as-Data Exercise/ira_tweets_csv_hashed.csv", fill=TRUE)

# make panel
df$blm_tweet <- as.numeric(str_detect(df$tweet_text, regex('Black Lives Matter|BLM', ignore_case = T)))
df$Date <- as.Date(df$tweet_time)
panel <- df %>%
  group_by(Date) %>%
  summarise(tweet_count = n(),
            quote_count = mean(quote_count),
            reply_count = mean(reply_count),
            retweet_count = mean(retweet_count),
            like_count = mean(like_count),
            blm_count = sum(blm_tweet))
```

```r
# balance panel
panel_balanced <- right_join(panel, data.frame(Date = seq(min(df$Date), max(df$Date), "day")))
panel_balanced[is.na(panel_balanced)] <- 0
panel_balanced <- panel_balanced[order(panel_balanced$Date),]

# number of obs:
cat("Total days in panel:", nrow(panel_balanced))
```

```
## Total days in panel: 3331
```

```r
events <- data.frame(event_date = c("2015-08-19", "2015-07-13", "2016-07-05"))
events$event_date <- as.Date(events$event_date)

# create function to loop over each event:
time_window <- 30
for(i in 1:3){

  # set event date
  event_date <- events$event_date[i]

  # subset by if date is within 14 days of event
  ddf <- panel_balanced
  ddf$diff <- ddf$Date - event_date
  ddf$window <- ifelse(((ddf$diff < time_window)&(ddf$diff >= -time_window)), 1,0)
  ddf <- filter(ddf, window == 1)

  # add the needed variables to run our model:
  ddf$T <- 1:nrow(ddf)
  ddf$X <- ifelse(ddf$diff >= 0,1,0)
  ddf$XT <- ifelse(ddf$diff >= 0, 1:nrow(ddf)/2, 0)

  # loop over every variable
  # and print summary

  for(x in 2:7){
    model <- lm(paste(colnames(ddf)[x], " ~ T + X + XT"), data=ddf)
    print(paste(colnames(ddf)[x]))
    print(summary(model))
  }

}
```

```
## [1] "tweet_count"
##
## Call:
## lm(formula = paste(colnames(ddf)[x], " ~ T + X + XT"), data = ddf)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -497.04 -136.14   45.44  162.51  588.42
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1764.736     86.179  20.478   <2e-16 ***
## T             -2.385      4.854  -0.491    0.625
```

```
## X            393.893    240.784   1.636    0.107
## XT           -20.957     13.730  -1.526    0.133
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 230.1 on 56 degrees of freedom
## Multiple R-squared:  0.2002, Adjusted R-squared:  0.1574
## F-statistic: 4.673 on 3 and 56 DF,  p-value: 0.005536
##
## [1] "quote_count"
##
## Call:
## lm(formula = paste(colnames(ddf)[x], " ~ T + X + XT"), data = ddf)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -1.142e-04 -5.391e-05  0.000e+00  0.000e+00  5.798e-04
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.218e-06  5.040e-05   0.123    0.902
## T            3.599e-06  2.839e-06   1.268    0.210
## X           -6.218e-06  1.408e-04  -0.044    0.965
## XT          -7.199e-06  8.030e-06  -0.896    0.374
##
## Residual standard error: 0.0001346 on 56 degrees of freedom
## Multiple R-squared:  0.07881,    Adjusted R-squared:  0.02946
## F-statistic: 1.597 on 3 and 56 DF,  p-value: 0.2003
##
## [1] "reply_count"
##
## Call:
## lm(formula = paste(colnames(ddf)[x], " ~ T + X + XT"), data = ddf)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.138786 -0.023983  0.001735  0.028600  0.116246
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.0959205  0.0163274   5.875 2.43e-07 ***
## T           0.0003983  0.0009197   0.433    0.667
## X           0.0184209  0.0456187   0.404    0.688
## XT          0.0003674  0.0026013   0.141    0.888
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0436 on 56 degrees of freedom
## Multiple R-squared:  0.1816, Adjusted R-squared:  0.1377
## F-statistic: 4.141 on 3 and 56 DF,  p-value: 0.01013
##
## [1] "retweet_count"
##
## Call:
```

```
## lm(formula = paste(colnames(ddf)[x], " ~ T + X + XT"), data = ddf)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.83205 -0.16097 -0.05218  0.13352  1.34501
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.737204   0.126914   5.809 3.11e-07 ***
## T           -0.003557   0.007149  -0.498    0.621
## X           -0.047417   0.354598  -0.134    0.894
## XT           0.013889   0.020220   0.687    0.495
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3389 on 56 degrees of freedom
## Multiple R-squared:  0.06499,    Adjusted R-squared:  0.0149
## F-statistic: 1.298 on 3 and 56 DF,  p-value: 0.2843
##
## [1] "like_count"
##
## Call:
## lm(formula = paste(colnames(ddf)[x], " ~ T + X + XT"), data = ddf)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.26350 -0.04728 -0.00837  0.03771  0.32078
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.074e-01  3.520e-02   5.891 2.28e-07 ***
## T           -8.646e-05  1.983e-03  -0.044    0.965
## X           -5.242e-02  9.834e-02  -0.533    0.596
## XT           5.342e-03  5.608e-03   0.953    0.345
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09399 on 56 degrees of freedom
## Multiple R-squared:  0.1413, Adjusted R-squared:  0.09526
## F-statistic: 3.071 on 3 and 56 DF,  p-value: 0.03506
##
## [1] "blm_count"
##
## Call:
## lm(formula = paste(colnames(ddf)[x], " ~ T + X + XT"), data = ddf)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.1410 -0.8243 -0.0264  0.3269  4.1352
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.12644    0.43060   2.616  0.01141 *
## T           -0.01246    0.02426  -0.514  0.60953
```

```
## X            4.70664    1.20310   3.912  0.00025 ***
## XT          -0.18020    0.06860  -2.627  0.01110 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.15 on 56 degrees of freedom
## Multiple R-squared:  0.2509, Adjusted R-squared:  0.2108
## F-statistic: 6.253 on 3 and 56 DF,  p-value: 0.0009748
##
## [1] "tweet_count"
##
## Call:
## lm(formula = paste(colnames(ddf)[x], " ~ T + X + XT"), data = ddf)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -465.7 -237.7  -47.7  189.6  734.3
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1224.685    113.965  10.746 3.16e-15 ***
## T              4.287      6.420   0.668   0.5070
## X            648.910    318.420   2.038   0.0463 *
## XT           -15.706     18.157  -0.865   0.3907
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 304.3 on 56 degrees of freedom
## Multiple R-squared:  0.3439, Adjusted R-squared:  0.3087
## F-statistic: 9.783 on 3 and 56 DF,  p-value: 2.749e-05
##
## [1] "quote_count"
##
## Call:
## lm(formula = paste(colnames(ddf)[x], " ~ T + X + XT"), data = ddf)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -3.281e-04 -6.922e-05 -5.462e-05 -1.928e-05  1.246e-03
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.434e-04  9.878e-05   3.476  0.00099 ***
## T           -1.525e-05  5.564e-06  -2.741  0.00821 **
## X           -3.302e-04  2.760e-04  -1.196  0.23656
## XT           3.241e-05  1.574e-05   2.059  0.04413 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0002638 on 56 degrees of freedom
## Multiple R-squared:  0.1262, Adjusted R-squared:  0.07944
## F-statistic: 2.697 on 3 and 56 DF,  p-value: 0.05446
##
## [1] "reply_count"
```

```
##
## Call:
## lm(formula = paste(colnames(ddf)[x], " ~ T + X + XT"), data = ddf)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.118176 -0.020482 -0.001387  0.021195  0.095098
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.1243321  0.0143499   8.664 6.32e-12 ***
## T           -0.0002198  0.0008083  -0.272    0.787
## X           -0.0633994  0.0400937  -1.581    0.119
## XT           0.0024010  0.0022863   1.050    0.298
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.03832 on 56 degrees of freedom
## Multiple R-squared:  0.06604,    Adjusted R-squared:  0.016
## F-statistic:  1.32 on 3 and 56 DF,  p-value: 0.277
##
## [1] "retweet_count"
##
## Call:
## lm(formula = paste(colnames(ddf)[x], " ~ T + X + XT"), data = ddf)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.66314 -0.11811 -0.02301  0.09573  0.62029
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.634e-01  9.306e-02   7.129 2.12e-09 ***
## T           -9.033e-06  5.242e-03  -0.002    0.999
## X           -2.481e-01  2.600e-01  -0.954    0.344
## XT           1.205e-02  1.483e-02   0.813    0.420
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2485 on 56 degrees of freedom
## Multiple R-squared:  0.02572,    Adjusted R-squared:  -0.02648
## F-statistic: 0.4927 on 3 and 56 DF,  p-value: 0.6888
##
## [1] "like_count"
##
## Call:
## lm(formula = paste(colnames(ddf)[x], " ~ T + X + XT"), data = ddf)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.178355 -0.037629 -0.002667  0.026562  0.227893
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)  0.1827810  0.0268709   6.802 7.34e-09 ***
## T           -0.0005803  0.0015136  -0.383   0.7029
## X           -0.1573942  0.0750775  -2.096   0.0406 *
## XT           0.0088091  0.0042811   2.058   0.0443 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.07176 on 56 degrees of freedom
## Multiple R-squared:  0.131,  Adjusted R-squared:  0.08442
## F-statistic: 2.813 on 3 and 56 DF,  p-value: 0.04748
##
## [1] "blm_count"
##
## Call:
## lm(formula = paste(colnames(ddf)[x], " ~ T + X + XT"), data = ddf)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.6001 -0.5021 -0.2857  0.2519  3.4493
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.05287    0.40167   0.132    0.896
## T            0.02024    0.02263   0.895    0.375
## X           -1.36670    1.12227  -1.218    0.228
## XT           0.05829    0.06399   0.911    0.366
##
## Residual standard error: 1.073 on 56 degrees of freedom
## Multiple R-squared:  0.1483, Adjusted R-squared:  0.1027
## F-statistic: 3.251 on 3 and 56 DF,  p-value: 0.02839
##
## [1] "tweet_count"
##
## Call:
## lm(formula = paste(colnames(ddf)[x], " ~ T + X + XT"), data = ddf)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -370.23 -191.21  -69.23  123.10  675.22
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  729.444    101.105   7.215 1.53e-09 ***
## T             -4.459      5.695  -0.783   0.4370
## X           -560.735    282.488  -1.985   0.0521 .
## XT            25.406     16.108   1.577   0.1204
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 270 on 56 degrees of freedom
## Multiple R-squared:  0.08945,    Adjusted R-squared:  0.04067
## F-statistic: 1.834 on 3 and 56 DF,  p-value: 0.1515
##
## [1] "quote_count"
```

```
##
## Call:
## lm(formula = paste(colnames(ddf)[x], " ~ T + X + XT"), data = ddf)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.3635 -0.3426 -0.1366  0.0779  5.4338
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.07593    0.34871  -0.218  0.82842
## T            0.03199    0.01964   1.629  0.10902
## X            3.24622    0.97429   3.332  0.00153 **
## XT          -0.16121    0.05556  -2.902  0.00530 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9312 on 56 degrees of freedom
## Multiple R-squared:  0.1976, Adjusted R-squared:  0.1546
## F-statistic: 4.597 on 3 and 56 DF,  p-value: 0.006027
##
## [1] "reply_count"
##
## Call:
## lm(formula = paste(colnames(ddf)[x], " ~ T + X + XT"), data = ddf)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.91537 -0.20358 -0.00786  0.22926  0.92550
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.382337   0.151489   2.524   0.0145 *
## T            0.013715   0.008533   1.607   0.1136
## X            1.040310   0.423261   2.458   0.0171 *
## XT          -0.054129   0.024136  -2.243   0.0289 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4045 on 56 degrees of freedom
## Multiple R-squared:  0.1448, Adjusted R-squared:  0.09897
## F-statistic:  3.16 on 3 and 56 DF,  p-value: 0.03157
##
## [1] "retweet_count"
##
## Call:
## lm(formula = paste(colnames(ddf)[x], " ~ T + X + XT"), data = ddf)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.1494  -4.2387  -0.1563   2.9311  20.2146
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)   7.1590     2.6793   2.672  0.00986 **
## T             0.1411     0.1509   0.935  0.35391
## X            22.0348     7.4860   2.943  0.00472 **
## XT           -1.0213     0.4269  -2.393  0.02011 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.155 on 56 degrees of freedom
## Multiple R-squared:  0.1459, Adjusted R-squared:  0.1001
## F-statistic: 3.188 on 3 and 56 DF,  p-value: 0.03055
##
## [1] "like_count"
##
## Call:
## lm(formula = paste(colnames(ddf)[x], " ~ T + X + XT"), data = ddf)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.4217  -3.1878  -0.0625   2.4202  17.8854
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   5.2893     2.1104   2.506   0.0151 *
## T             0.1305     0.1189   1.098   0.2769
## X            15.2874     5.8965   2.593   0.0121 *
## XT           -0.7429     0.3362  -2.209   0.0312 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.636 on 56 degrees of freedom
## Multiple R-squared:  0.1225, Adjusted R-squared:  0.07547
## F-statistic: 2.605 on 3 and 56 DF,  p-value: 0.0607
##
## [1] "blm_count"
##
## Call:
## lm(formula = paste(colnames(ddf)[x], " ~ T + X + XT"), data = ddf)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.5883 -1.4279 -0.5735  0.5138 12.1114
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.848276   0.869994   2.124   0.0381 *
## T           -0.028921   0.049006  -0.590   0.5575
## X            2.151576   2.430766   0.885   0.3799
## XT          -0.002225   0.138609  -0.016   0.9873
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.323 on 56 degrees of freedom
## Multiple R-squared:  0.08123,    Adjusted R-squared:  0.03201
## F-statistic:  1.65 on 3 and 56 DF,  p-value: 0.1881
```

Exercise #2

```
# Load violence and reconstruction spending data
violence <- readstata13::read.dta13("./3. Panel Data Exercise/ESOC-I_Replication_V2/sigact_dist_month_0
cerp <- readstata13::read.dta13("./3. Panel Data Exercise/ESOC-I_Replication_V2/CERP_dist_month_0308_HA

# Load population data
pop <- readstata13::read.dta13("./3. Panel Data Exercise/ESOC-I_Replication_V2/population_HAM.dta")
pop <- select(pop, District = district, Governorate = governorate, year, pop)

# Note that there are some mistakes with the assignment
# of district to governorate in the violence data.
# For example, Afaq is assigned to both Baghdad and Qadissiya.
# This is an issue also in the CERP data, which has lots of
# missing governorates per district.
# Population data is the most complete and from WFP so let's use that
districts <- unique(pop[,c('District', 'Governorate')])

# Delete the Governorate column from violence and cerp datasets
violence <- select(violence, -Governorate)
cerp <- select(cerp, -Governorate)

# Merge them together at the district-half year level
colnames(violence)
```

```
## [1] "District" "MONTH"    "SIGACT"   "SIG_1"
```

```
colnames(cerp)
```

```
##  [1] "District"               "monthyr"                "month_spent"
##  [4] "ms_c"                   "ms_u"                   "ms_democracy"
##  [7] "ms_education"           "ms_electricity"         "ms_healthcare"
## [10] "ms_transport"           "ms_watersan"            "ms_large"
## [13] "ms_pubbuild"            "ms_democracy_noncerp"   "ms_education_noncerp"
## [16] "ms_electricity_noncerp" "ms_healthcare_noncerp"  "ms_large_noncerp"
## [19] "ms_pubbuild_noncerp"    "np_cerp"                "np_c"
## [22] "np_u"                   "np_democracy"           "np_education"
## [25] "np_electricity"         "np_healthcare"          "np_transport"
## [28] "np_watersan"            "np_large"               "np_pubbuild"
## [31] "np_democracy_noncerp"   "np_education_noncerp"   "np_electricity_noncerp"
## [34] "np_healthcare_noncerp"  "np_transport_noncerp"   "np_watersan_noncerp"
## [37] "np_large_noncerp"       "np_pubbuild_noncerp"    "projcount"
## [40] "YEAR"                   "MONTH"
```

```
colnames(violence)[2] <- "monthyr"

df <- full_join(cerp, violence) # Note that you will need to full join as there are missing districts i
df$SIG_1[is.na(df$SIG_1)] <- 0 # Don't forget to fill in violence data as 0 for Choman
df$MONTH2 = ymd("1960-01-01") + months(df$monthyr) # Convert stata months
# Create half-year variable
df$halfyr <- ifelse(month(df$MONTH2) >= 7,
                    paste0(year(df$MONTH2),"h2"),
                    paste0(year(df$MONTH2),"h1"))
df <- df %>%
  plyr::join(., districts) %>%
```

```
  filter(year(MONTH2) >= 2004, year(MONTH2) <= 2008) %>%
  group_by(District, Governorate, halfyr) %>%
  summarise(ms_c = sum(ms_c),
            SIG_1 = sum(SIG_1)) %>%
  mutate(year = as.numeric(substr(halfyr, start=1, stop=4)),
         half = as.numeric(substr(halfyr, start=6, stop=6)))

# Create per capita (1000 pop.) measures
df <- plyr::join(df, pop)

df$p_S1 <- df$SIG_1 / df$pop*1000
df$p_ms_c <- df$ms_c / df$pop

cat("Number of districts: ", length(unique(df$District)))
```

```
## Number of districts:  104
```

```
cat("Number of half-years: ", length(unique(df$halfyr)))
```

```
## Number of half-years:  10
```

```
cat("Number of observations: ", nrow(df))
```

```
## Number of observations:  1040
```

```
# base model
mod1 <- lm(p_S1 ~ p_ms_c, df, weights=pop)
lmtest::coeftest(mod1, sandwich::vcovCL(mod1, type="HC1", cluster = ~District))
```

```
##
## t test of coefficients:
##
##              Estimate Std. Error t value  Pr(>|t|)
## (Intercept) 0.3611243  0.0852531  4.2359 2.478e-05 ***
## p_ms_c      0.0212734  0.0037942  5.6069 2.640e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
### adding control vars

# sunni & shia vote shares:
elections <- readstata13::read.dta13("./3. Panel Data Exercise/ESOC-I_Replication_V2/Dec_2005_Vote_Data
df <- plyr::join(df, elections[,c("Governorate", "su_v", "sh_v")])

# unemployment rate:
econ <- readstata13::read.dta13("./3. Panel Data Exercise/ESOC-I_Replication_V2/econfactors_HAM.dta")
colnames(econ)[1] <- "District"
df <- plyr::join(df, econ[,c("District", "year", "urate_2008")])

# income variables:
econ$b2_prop <- econ$hhinc_i1_2008 + econ$hhinc_i2_2008
df <- plyr::join(df, econ[,c("District", "year", "b2_prop")])

# mean change in HHI quintiles between 2002 and 2004:
community <- readstata13::read.dta13("./3. Panel Data Exercise/ESOC-I_Replication_V2/ILCS_district.dta")
df <- plyr::join(df, community[,c("District", 'dif_02_04_qcap')])
```

```r
# with basic controls
mod2 <- lm(p_S1 ~ p_ms_c + su_v + sh_v + urate_2008 + dif_02_04_qcap + b2_prop, df, weights=pop)
lmtest::coeftest(mod2, sandwich::vcovCL(mod2, type="HC1", cluster = ~District))
```

```
##
## t test of coefficients:
##
##                  Estimate Std. Error t value  Pr(>|t|)
## (Intercept)     0.3540683  0.1492578  2.3722  0.017873 *
## p_ms_c          0.0137916  0.0037949  3.6343  0.000293 ***
## su_v            2.2011208  0.2898376  7.5943 7.127e-14 ***
## sh_v            0.3629594  0.1823771  1.9902  0.046847 *
## urate_2008     -1.9528416  1.1225390 -1.7397  0.082228 .
## dif_02_04_qcap -0.2371980  0.1767470 -1.3420  0.179896
## b2_prop        -0.0082629  0.0035457 -2.3304  0.019984 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# with time controls
df <- fastDummies::dummy_cols(df, "year")
df$year_2005_su_v <- df$year_2005 * df$su_v
df$year_2006_su_v <- df$year_2006 * df$su_v
df$year_2007_su_v <- df$year_2007 * df$su_v
df$year_2008_su_v <- df$year_2008 * df$su_v

mod3 <- lm(p_S1 ~ p_ms_c + su_v + sh_v + urate_2008 + dif_02_04_qcap + b2_prop + year + year:su_v, df, w
lmtest::coeftest(mod3, sandwich::vcovCL(mod3, type="HC1", cluster = ~District))
```

```
##
## t test of coefficients:
##
##                   Estimate  Std. Error t value  Pr(>|t|)
## (Intercept)     50.6258801  57.9995252  0.8729 0.3829469
## p_ms_c           0.0143951   0.0037721  3.8162 0.0001439 ***
## su_v            -5.4623516 203.2405744 -0.0269 0.9785638
## sh_v             0.3685476   0.1830640  2.0132 0.0443617 *
## urate_2008      -2.1191509   1.2543034 -1.6895 0.0914374 .
## dif_02_04_qcap  -0.2361800   0.1762328 -1.3402 0.1805007
## b2_prop         -0.0079498   0.0035486 -2.2403 0.0252943 *
## year            -0.0250610   0.0288897 -0.8675 0.3858945
## su_v:year        0.0038168   0.1013002  0.0377 0.9699520
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# create FD:
df <- df[order(df$District, df$year, df$half),]
df <- df %>%
  group_by(District) %>%
  mutate(d.p_S1 = p_S1 - lag(p_S1),
         d.p_ms_c = p_ms_c - lag(p_ms_c))

mod4 <- lm(d.p_S1 ~ d.p_ms_c + year_2005 + year_2006 + year_2007 + year_2008 +
             year_2005_su_v + year_2006_su_v + year_2007_su_v + year_2008_su_v, df, weights=pop)
lmtest::coeftest(mod4, sandwich::vcovCL(mod4, type="HC1", cluster = ~District))
```

```
##
## t test of coefficients:
##
##                  Estimate Std. Error t value  Pr(>|t|)
## (Intercept)     0.2172913  0.0461283  4.7106 2.848e-06 ***
## d.p_ms_c       -0.0094509  0.0042964 -2.1998  0.028071 *
## year_2005      -0.2522461  0.0533684 -4.7265 2.639e-06 ***
## year_2006      -0.1622804  0.0632027 -2.5676  0.010396 *
## year_2007      -0.0977735  0.0760045 -1.2864  0.198619
## year_2008      -0.3548905  0.0877166 -4.0459 5.647e-05 ***
## year_2005_su_v  0.6750848  0.2055273  3.2846  0.001059 **
## year_2006_su_v  0.8391445  0.2867263  2.9266  0.003510 **
## year_2007_su_v -1.0307333  0.5424841 -1.9000  0.057740 .
## year_2008_su_v -0.7525954  0.1620027 -4.6456 3.882e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# control for pre-existing trends:
df <- df %>%
  group_by(District) %>%
  mutate(l1_p_S1 = lag(p_S1),
         l2_p_S1 = lag(l1_p_S1),
         ld.p_S1 = l1_p_S1 - l2_p_S1)

mod5 <- lm(d.p_S1 ~ d.p_ms_c + ld.p_S1 + year_2005 + year_2006 + year_2007 + year_2008 +
             year_2005_su_v + year_2006_su_v + year_2007_su_v + year_2008_su_v, df, weights=pop)
lmtest::coeftest(mod5, sandwich::vcovCL(mod5, type="HC1", cluster = ~District))
```

```
##
## t test of coefficients:
##
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -0.1236831  0.0408982 -3.0242 0.002571 **
## d.p_ms_c       -0.0111449  0.0043213 -2.5791 0.010080 *
## ld.p_S1         0.1948104  0.0803293  2.4251 0.015517 *
## year_2005       0.0966691  0.0503405  1.9203 0.055165 .
## year_2006       0.1792451  0.0654569  2.7384 0.006308 **
## year_2007       0.2171810  0.0976939  2.2231 0.026482 *
## year_2005_su_v  0.5513319  0.2002971  2.7526 0.006043 **
## year_2006_su_v  0.7364528  0.2493742  2.9532 0.003235 **
## year_2007_su_v -1.0698338  0.4898257 -2.1841 0.029236 *
## year_2008_su_v -0.5518250  0.1733516 -3.1833 0.001511 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```