RA_Task_Final

Kishika Mahajan

2024-03-01

Setting the working directory, loading the libraries and the datasets -

```
setwd("/Users/kishikamahajan/Desktop/RA Task")
library(tidyverse)
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ─
─
## ✔ dplyr     1.1.4     ✔ readr     2.1.4
## ✔ forcats   1.0.0     ✔ stringr   1.5.0
## ✔ ggplot2   3.4.3     ✔ tibble    3.2.1
## ✔ lubridate 1.9.2     ✔ tidyr     1.3.0
## ✔ purrr     1.0.2
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ─
─
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conf
licts to become errors
```

```
library(foreign)
library(countrycode)
library(sf)
```

```
## Linking to GEOS 3.10.2, GDAL 3.4.2, PROJ 8.2.1; sf_use_s2() is TRUE
```

```
library(maps)
```

```
##
## Attaching package: 'maps'
##
## The following object is masked from 'package:purrr':
##
##     map
```

```
library(ggplot2)
library(plm)
```

```
##
## Attaching package: 'plm'
##
## The following objects are masked from 'package:dplyr':
##
##     between, lag, lead
```

```
city_level_data <- readRDS("reg_data.RDS")
country_level_measure <- read.csv("fiw.csv")
```

Converting the country codes to country names , renaming the “countrycode” column to “country” and changing the country names in the city level data with the first letter as capital and rest as small to facilitate merging -

```
country_level_measure$countrycode <- countrycode(country_level_measure$countrycode,
"iso3c" , "country.name")

country_level_measure <- country_level_measure %>%
  rename(country = countrycode)

city_level_data$country <- str_to_title(city_level_data$country)
```

Converting the two datasets into panel data -

```
panel_data_countries <- pdata.frame(country_level_measure , index = c("country" , "
year"))
panel_data_cities <- pdata.frame(city_level_data , index = c("country" , "year"))
```

Merging the two datasets and then converting the merged dataset to a panel data -

```
merged_dataset <- merge(panel_data_cities, panel_data_countries, by = c("country",
"year"))
merged_panel_data <- pdata.frame(merged_dataset , index = c("country" , "year"))
```

Cleaning panel data for any duplicate values -

```
duplicates <- duplicated(row.names(merged_panel_data))
final_panel_data <- merged_panel_data[!duplicates, ]
```

Running regression on merged panel data-

```
fixed_effect_model <- plm(pm2_5 ~ pm2.5_20km, data = final_panel_data, model = "wit
hin", effect = "individual")
summary(fixed_effect_model)
```

```
## Oneway (individual) effect Within Model
## 
## Call:
## plm(formula = pm2_5 ~ pm2.5_20km, data = final_panel_data, effect = "individual"
,
##     model = "within")
## 
## Unbalanced Panel: n = 120, T = 1-13, N = 669
## 
## Residuals:
##       Min.    1st Qu.     Median    3rd Qu.       Max. 
## -40.867768  -2.519894  -0.037493   2.163754 101.960810 
## 
## Coefficients:
##            Estimate Std. Error t-value  Pr(>|t|)    
## pm2.5_20km 0.868093   0.067583  12.845 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Total Sum of Squares:    86932
## Residual Sum of Squares: 66815
## R-Squared:      0.23141
## Adj. R-Squared: 0.063104
## F-statistic: 164.992 on 1 and 548 DF, p-value: < 2.22e-16
```

Extracting the fixed effects from the model -

```
fixed_effects <- summary(fixef(fixed_effect_model, type = "dmean"))
fixed_effects
```

```
##                       Estimate Std. Error t-value  Pr(>|t|)    
## Afghanistan           71.85756    8.06985  8.9044 < 2.2e-16 ***
## Albania               -4.14850    5.65082 -0.7341 0.4631773    
## Algeria                4.99478   11.07859  0.4509 0.6522758    
## Andorra                0.32722    5.56490  0.0588 0.9531319    
## Argentina             -4.28953    5.59890 -0.7661 0.4439235    
## Australia             -3.29214    3.36601 -0.9781 0.3284782    
## Austria               -1.81227    3.60899 -0.5022 0.6157611    
## Bahamas               -5.79968    6.38870 -0.9078 0.3643815    
## Bahrain                9.41065    6.51361  1.4448 0.1490951    
## Bangladesh             3.05725    6.00791  0.5089 0.6110480    
## Belarus                1.96093    4.31230  0.4547 0.6494845    
## Belgium               -0.81982    4.62184 -0.1774 0.8592753    
## Benin                -24.71415   11.25917 -2.1950 0.0285806 *  
## Bhutan                26.47049    4.71879  5.6096 3.221e-08 ***
## Bolivia                9.88762    6.66065  1.4845 0.1382558    
## Brazil                -2.78993    3.81525 -0.7313 0.4649338    
## Brunei                -5.86667   11.06090 -0.5304 0.5960516    
## Bulgaria              -4.08721    4.48031 -0.9123 0.3620316    
## Cambodia             -5.73574    6.54371 -0.8765 0.3811275    
## Cameroon             56.25078   12.35886  4.5515 6.567e-06 ***
## Canada               -2.16960    3.69733 -0.5868 0.5575791    
## Chile                 3.43610    3.49095  0.9843 0.3254087    
## China                 5.43677    4.91491  1.1062 0.2691347    
## Colombia             -2.31638    3.91094 -0.5923 0.5539067    
## Costa Rica            0.40169    4.71945  0.0851 0.9322022    
## Croatia              -5.01888    4.66142 -1.0767 0.2820953    
## Cuba                 22.71741   11.05881  2.0542 0.0404266 *  
## Cyprus               -4.86669    3.55692 -1.3682 0.1718011    
## Czechia               0.58567    5.66793  0.1033 0.9177382    
## Denmark              -0.79512    3.96193 -0.2007 0.8410147    
## Dominican Republic   31.54981    6.42250  4.9124 1.190e-06 ***
## Ecuador              -4.05893    3.51307 -1.1554 0.2484384    
## Egypt                31.14679   11.42869  2.7253 0.0066294 ** 
## El Salvador           3.36331    4.43175  0.7589 0.4482309    
## Estonia              -5.32409    3.23526 -1.6456 0.1004098    
## Ethiopia             -3.07317    5.83812 -0.5264 0.5988245    
## Fiji                 -2.93440    7.82176 -0.3752 0.7076876    
## Finland              -3.43519    4.19397 -0.8191 0.4130975    
## France               -2.06858    3.77902 -0.5474 0.5843368    
## Georgia              -2.40906    6.60722 -0.3646 0.7155429    
## Germany              -2.76551    3.99690 -0.6919 0.4892842    
## Ghana                49.33639    5.18437  9.5164 < 2.2e-16 ***
## Greece               -1.35837    4.62441 -0.2937 0.7690688    
## Guatemala            -2.54456    5.47180 -0.4650 0.6420934    
## Honduras             -3.68397    5.88681 -0.6258 0.5317061    
## Hungary              -2.84693    5.08442 -0.5599 0.5757550    
## Iceland              -4.87242    3.51121 -1.3877 0.1657997    
## India                12.24088    8.25578  1.4827 0.1387283    
## Indonesia            -3.88741    3.96199 -0.9812 0.3269393    
## Iran                 -5.59841    4.64936 -1.2041 0.2290615    
## Iraq                 11.16921    6.55489  1.7040 0.0889573 .  
## Ireland              -2.33065    4.55185 -0.5120 0.6088412    
## Israel               -1.09728    3.93573 -0.2788 0.7805040    
## Italy                -2.55821    3.68097 -0.6950 0.4873612    
## Jamaica              -2.85004    5.63775 -0.5055 0.6133909    
## Japan                -3.24972    4.60243 -0.7061 0.4804338    
## Jordan               -1.56107    5.46021 -0.2859 0.7750635    
## Kazakhstan           11.70743    7.84930  1.4915 0.1363992    
## Kenya               -10.21705    5.67782 -1.7995 0.0724950 .  
## Kuwait               -1.36761    5.28317 -0.2589 0.7958392    
## Kyrgyzstan            6.61758   11.17328  0.5923 0.5539153    
## Latvia               -4.76066    3.83150 -1.2425 0.2145814    
## Lebanon              -6.32845    4.64603 -1.3621 0.1737191    
## Liberia             -13.93552   11.13541 -1.2515 0.2113007    
## Lithuania            -5.41797    4.61339 -1.1744 0.2407445    
## Luxembourg           -2.36517    5.60077 -0.4223 0.6729766    
## Madagascar           24.43259    4.72663  5.1691 3.301e-07 ***
## Malaysia             -3.28020    5.64624 -0.5810 0.5615105    
## Maldives            -12.29375   11.11747 -1.1058 0.2692965    
## Malta                 -4.16765    4.00285 -1.0412 0.2982562    
## Mauritius              3.98137    5.06236  0.7865 0.4319355    
## Mexico                 3.24520    4.42473  0.7334 0.4636134    
## Monaco                -7.26226    5.60751 -1.2951 0.1958324    
## Mongolia              20.29810    5.24419  3.8706 0.0001217 ***
## Montenegro             0.42910    5.68224  0.0755 0.9398315    
## Morocco                1.20896    5.13064  0.2356 0.8138032    
## Nepal                  4.01863    4.87098  0.8250 0.4097221    
## Netherlands           -3.11666    4.26071 -0.7315 0.4647927    
## New Zealand           -1.62900    4.20235 -0.3876 0.6984318    
## Nigeria                1.75719    7.10102  0.2475 0.8046479    
## North Macedonia       12.28209    8.15582  1.5059 0.1326615    
## Norway                -2.04406    3.92434 -0.5209 0.6026686    
## Oman                 -13.77664    8.46779 -1.6269 0.1043232    
## Pakistan              13.96448    5.78802  2.4127 0.0161643 *  
## Panama                -4.38442   11.09517 -0.3952 0.6928749    
## Paraguay               1.75096   11.08277  0.1580 0.8745236    
## Peru                  -6.33987    4.82256 -1.3146 0.1891848    
## Philippines            9.65681    5.30257  1.8212 0.0691281 .  
## Poland                -0.26096    4.16531 -0.0627 0.9500671    
## Portugal              -5.34309    3.95841 -1.3498 0.1776351    
## Qatar                -36.02724    7.33341 -4.9128 1.188e-06 ***
## Romania               -1.95678    5.66303 -0.3455 0.7298243    
## Russia                -5.58483    7.93329 -0.7040 0.4817480    
## Saudi Arabia           0.43801    7.98614  0.0548 0.9562811    
## Senegal               -8.78645    4.83859 -1.8159 0.0699301 .  
## Serbia                -0.15153    8.03343 -0.0189 0.9849579    
## Singapore             -3.05983    3.28622 -0.9311 0.3522061    
## Slovakia              -1.70919    4.09243 -0.4176 0.6763692    
## Slovenia              -2.84532    4.33192 -0.6568 0.5115676    
## South Africa          -3.95274    4.79541 -0.8243 0.4101419    
## South Korea            0.66515    4.39450  0.1514 0.8797469    
## Spain                 -4.98521    4.55279 -1.0950 0.2740070    
## Sri Lanka              1.66486    5.26940  0.3159 0.7521621    
## Sweden                -4.55697    4.95932 -0.9189 0.3585685    
## Switzerland           -3.67982    3.98380 -0.9237 0.3560522    
## Tajikistan            39.52763   11.24067  3.5165 0.0004735 ***
## Tanzania               1.97700    7.91589  0.2498 0.8028734    
## Thailand              -4.64444    6.58018 -0.7058 0.4805988    
## Tunisia               16.05777   11.12599  1.4433 0.1495167    
## Turkey                 4.05632    6.68751  0.6066 0.5443997    
## Turkmenistan          -7.62653   11.21466 -0.6801 0.4967599    
## Uganda                31.58001    6.09115  5.1846 3.050e-07 ***
## Ukraine                2.79839    6.48244  0.4317 0.6661379    
## United Arab Emirates -10.47264    6.28406 -1.6665 0.0961770 .  
## United Kingdom        -3.46286    4.22582 -0.8195 0.4128846    
## United States         -0.06883    3.12958 -0.0220 0.9824612    
## Uruguay               -3.29928    6.43788 -0.5125 0.6085221    
## Uzbekistan            -0.63671    8.35485 -0.0762 0.9392809    
## Venezuela              6.24727    7.95124  0.7857 0.4323840    
## Vietnam               -8.57696    4.14528 -2.0691 0.0390058 *  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Creating the dataframe with countries as a column -

```
fixed_effects <- fixed_effects %>%
  as.data.frame() %>%  # Convert to dataframe (if not already)
  mutate(country = rownames(fixed_effects))
```

Loading the shapefile and renaming the country column -

```
world_shapefile <- st_read(dsn = "/Users/kishikamahajan/Desktop/RA Task/world-administrative-boundaries/world-administrative-boundaries.shx")
```

```
## Reading layer `world-administrative-boundaries' from data source
##   `/Users/kishikamahajan/Desktop/RA Task/world-administrative-boundaries/world-administrative-boundaries.shx'
##   using driver `ESRI Shapefile'
## Simple feature collection with 256 features and 8 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -180 ymin: -58.49861 xmax: 180 ymax: 83.6236
## Geodetic CRS:  WGS 84
```

```
world_shapefile <- world_shapefile %>%
  rename(country = name)
```

Merging the shapefile and the final panel dataset to get boundaries -

```
boundaries_dataset <- left_join(world_shapefile , final_panel_data , by = "country")
```

Creating the categories -

```
boundaries_dataset<- boundaries_dataset %>%
  mutate(country_type = case_when(
    fiw_pr < 3 ~ "Free",
    fiw_pr >= 3 & fiw_pr <= 5 ~ "Partially Free",
    fiw_pr > 5 ~ "Not Free"
  ))
```

Merging the boundaries database and the fixed effects database -

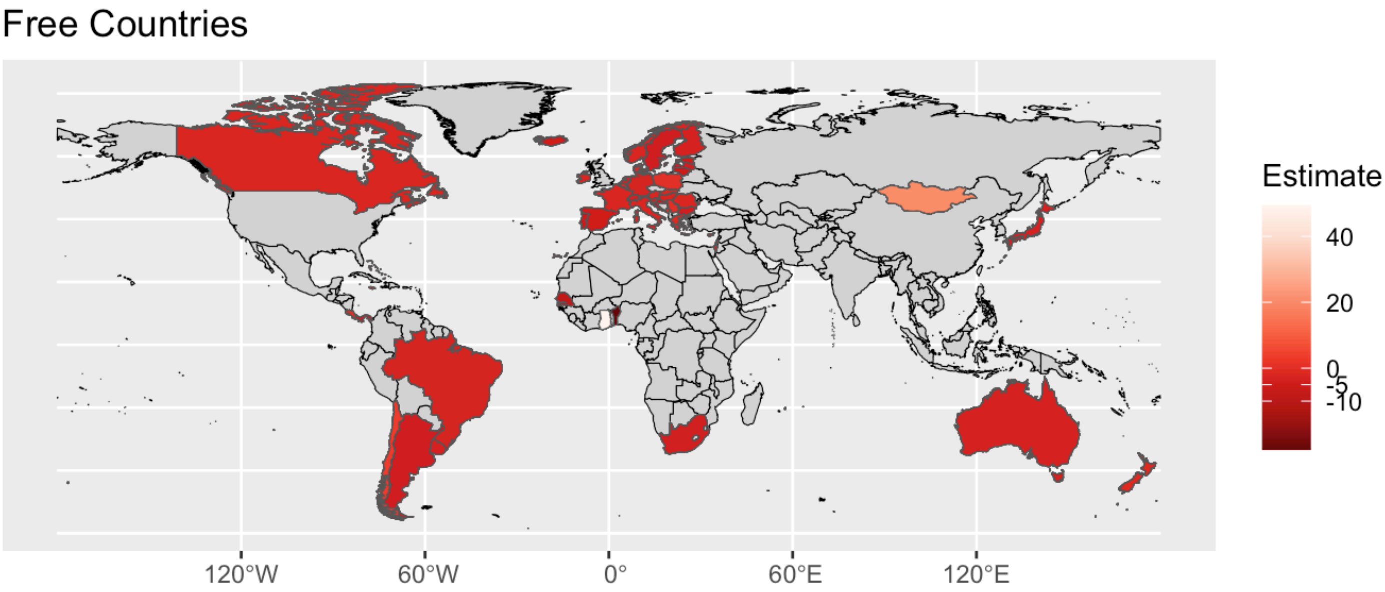
```
merged_regression_database <- merge(boundaries_dataset , fixed_effects , by = "country")
```

Creating subsets for each country type -

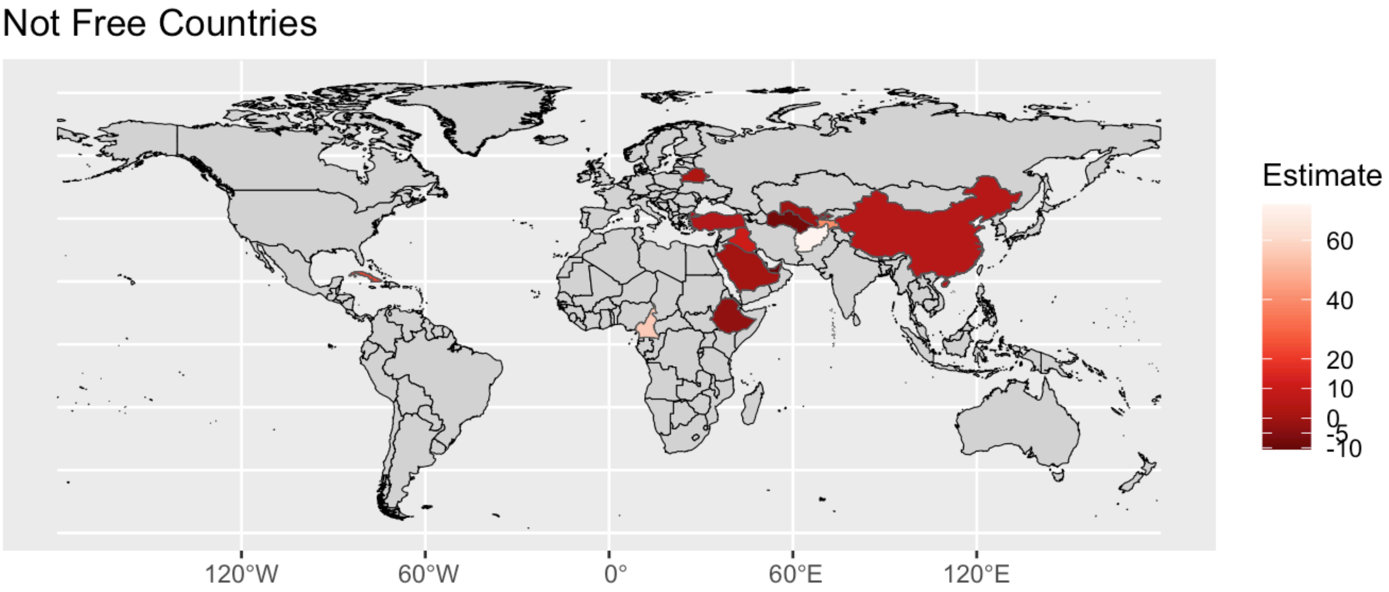
```
#For free countries -
free_countries <- merged_regression_database %>%
  subset(country_type == "Free")

#For not free countries -
not_free_countries <- merged_regression_database %>%
  subset(country_type == "Not Free")

#For partially free countries -
partially_free_countries <- merged_regression_database %>%
  subset(country_type == "Partially Free")
```

Creating respective maps -

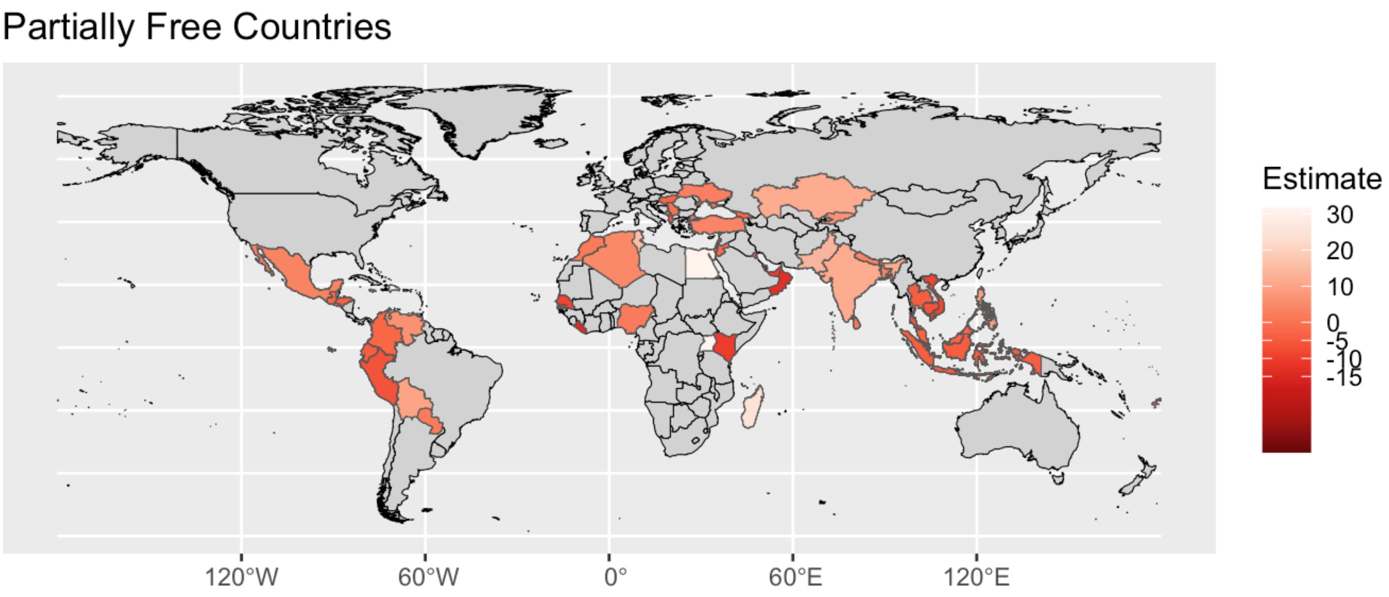
```
#For free countries -
ggplot() +
  geom_sf(data = world_shapefile, aes(geometry = geometry), fill = "lightgray", col
or = "black") +
  geom_sf(data = free_countries, aes(geometry = geometry, fill = Estimate)) +
  scale_fill_gradientn(colours = rev(RColorBrewer::brewer.pal(9, "Reds")),
                       breaks = c(-10, -5, 0, 20, 40, 60)) +
  ggtitle("Free Countries")
```



```
#For not free countries -
ggplot() +
  geom_sf(data = world_shapefile, aes(geometry = geometry), fill = "lightgray", color = "black") +
  geom_sf(data = not_free_countries, aes(geometry = geometry, fill = Estimate)) +
  scale_fill_gradientn(colours = rev(RColorBrewer::brewer.pal(9, "Reds")),
                       breaks = c(-15, -10, -5, 0, 10, 20, 40, 60, 80)) +
  ggtitle("Not Free Countries")
```



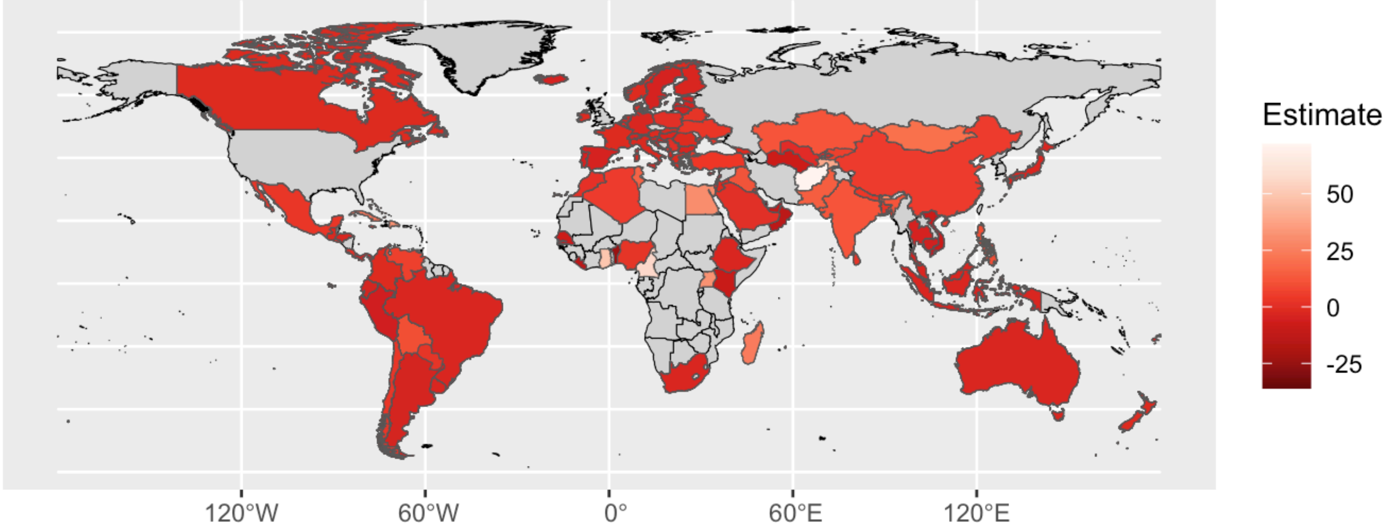
```
#For partially free countries -
ggplot() +
  geom_sf(data = world_shapefile, aes(geometry = geometry), fill = "lightgray", color = "black") +
  geom_sf(data = partially_free_countries, aes(geometry = geometry, fill = Estimate)) +
  scale_fill_gradientn(colours = rev(RColorBrewer::brewer.pal(9, "Reds")),
                       breaks = c(-15, -10, -5, 0, 10, 20, 30)) +
  ggtitle("Partially Free Countries")
```



Plotting all countries -

```
ggplot() +
  geom_sf(data = world_shapefile, aes(geometry = geometry), fill = "lightgray", col
or = "black") +
  geom_sf(data = merged_regression_database, aes(geometry = geometry, fill = Estima
te)) +
  scale_fill_gradientn(colours = rev(RColorBrewer::brewer.pal(9, "Reds"))) +
  ggtitle("All countries")
```

All countries



Conclusion:

A negative value of the coefficient (Estimate) suggests that a same-sized increase in satellite PM2.5 measurements is associated with a smaller increase in reported PM2.5 pollution levels in the country. Essentially, it indicates whether there is underreporting of PM2.5 pollution levels by governments compared to satellite measurements. More negative values of βi suggest higher levels of underreporting.

What can be seen is that for free and not free countries, the estimate is mostly negative with slightly higher negative values being in not free countries as compared to free countries. This means the level of underreporting (whenever there is any) is slightly higher in not free countries. However, this difference is minuscle. Moreover, more free countries underreport as compared to not free countries. Even for partially free countries, there is some level of underreporting (like free and not free countries), but here, there are alot of countries in this category that are not underreporting.

Conclusively, there is no stark pattern of underreporting / overreporting that can be derived solely on the basis of democracy levels.