Cross Section Assignment

Jacques Rossouw[a]

[a] *Stellebosch, Western Cape, South Africa*

```
extract_continents()
```

```
## # A tibble: 7 x 1
##   continent
##   <chr>
## 1 Asia
## 2 <NA>
## 3 Europe
## 4 Africa
## 5 North America
## 6 South America
## 7 Oceania
```

```
# number_countries <- read_csv(file = "./data/owid-covid-data.csv",
#                              show_col_types = F) %>%
#     filter(!location %in% c(continents)) %>%
#     filter(date == first(date)) %>%
#     select(location) %>%
#     nrow()
```

```
# quick_check2 <-  read_csv(file = "./data/owid-covid-data.csv",
#                              show_col_types = F) %>%
#     group_by(location) %>%
#     filter(date == first(date)) %>%
#     ggplot() +
#     geom_point(aes(x = location, y = date)) +
```

Email address: gerardrossouw@gmail.com (Jacques Rossouw)

```
#     geom_hline(yintercept = lubridate::ymd(20200430), color = "red")
#
#
# quick_check2
```

```
# number_countries_incl <- read_csv(file = "./data/owid-covid-data.csv",
#                             show_col_types = F) %>%
#     group_by(location) %>%
#     filter(!location %in% c(continents)) %>%
#     filter(date == first(date)) %>%
#     filter(!is.na(continent)) %>%
#     filter(date <= lubridate::ymd(20200430)) %>%
#     select(location) %>%
#     unique()
```

```
# read_csv(file = "./data/owid-covid-data.csv",
#                             show_col_types = F) %>%
#     group_by(location) %>%
#     filter(!location %in% c(continents)) %>%
#     filter(!is.na(continent)) %>%
#     filter(first(date) <= lubridate::ymd(20200430)) %>%
#     filter(date == first(date)) %>%
#     select(location)
```

First, we import the data and remove unnecessary columns. Some transformations are also made to columns to make them more usable for regressio. The variables that are distributed on a wider range, or scale, are also scaled to ensure that the OLS estimation is not biased by this.

```
fmxdat::source_all("./code")
# New vaccinations is transformed to new_vaccinations relative to population size
# i.e. per 1000 people
# Same goes for new_tests
# hosp_paitents, per 1000000
# as well as icu

world_df <- extract_all() %>%
```

```r
    feature_adj_all() %>%
    experiment_aggregate_week() %>%
    experiment_trim() %>%
    relocate(afflicted_rate, .before = reproduction_rate)

world_df
```

```
## # A tibble: 2,050 x 19
## # Groups:   location, date [2,050]
##    location    date       afflicted_rate reproduction_rate new_tests
##    <chr>       <date>              <dbl>             <dbl>     <dbl>
##  1 Afghanistan 2020-03-31           2.38             0.124         0
##  2 Afghanistan 2020-06-30           2.35             1.31          0
##  3 Afghanistan 2020-09-30           9.14             0.876         0
##  4 Afghanistan 2020-12-31           5.60             1.12          0
##  5 Afghanistan 2021-03-31           7.15             0.922         0
##  6 Afghanistan 2021-06-30           3.84             1.29          0
##  7 Afghanistan 2021-09-30           6.39             0.785         0
##  8 Afghanistan 2021-12-31           5.22             1.02          0
##  9 Afghanistan 2022-03-31           1.60             1.13          0
## 10 Afghanistan 2022-06-15           1.15             1.08          0
## # ... with 2,040 more rows, and 14 more variables: new_vaccinations <dbl>,
## #   stringency_index <dbl>, population_density <dbl>, median_age <dbl>,
## #   aged_65_older <dbl>, gdp_per_capita <dbl>, extreme_poverty <dbl>,
## #   cardiovasc_death_rate <dbl>, diabetes_prevalence <dbl>,
## #   handwashing_facilities <dbl>, hosp_beds_1k <dbl>, life_expectancy <dbl>,
## #   human_development_index <dbl>, smokers <dbl>
```

1. Cumulative Values

```r
options(scipen=999)
fmxdat::source_all("./code")
cols_range(world_df)
```

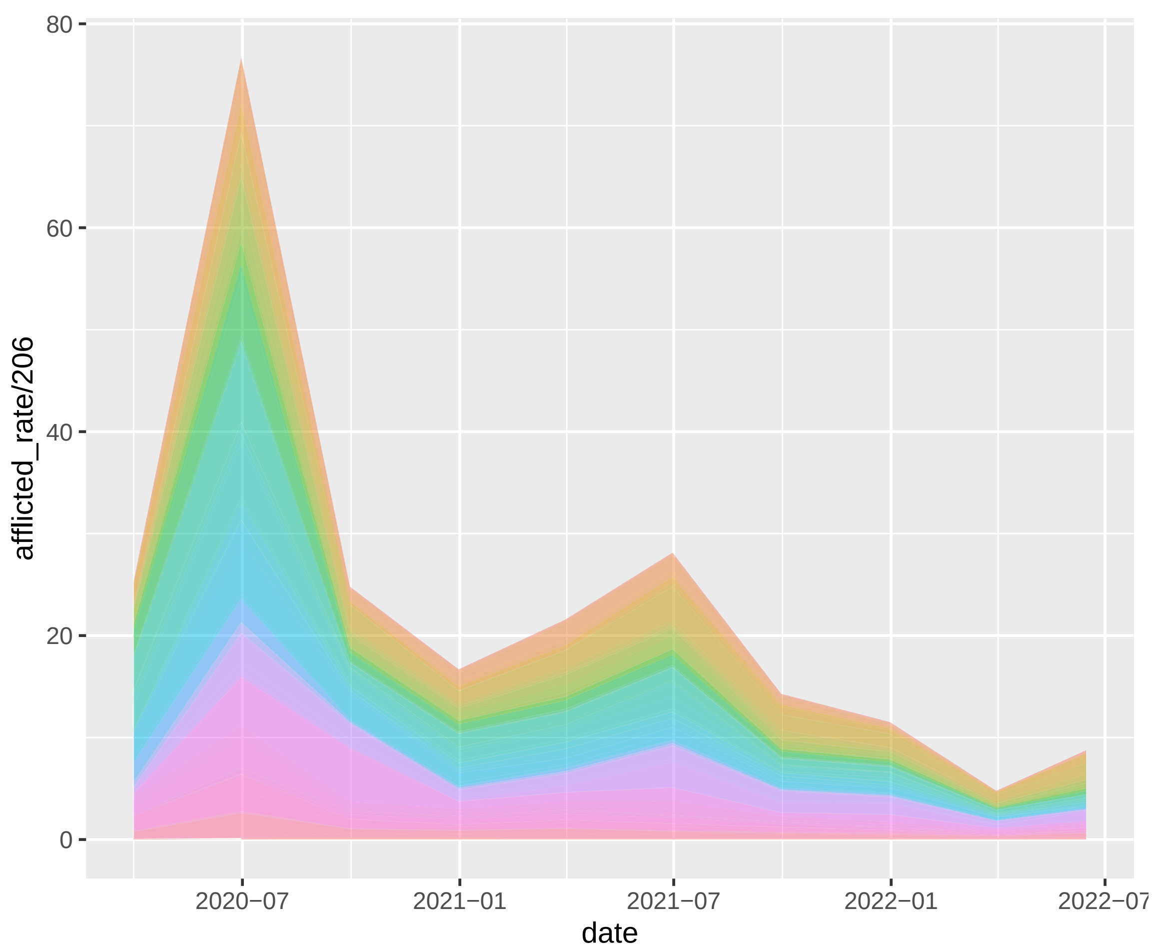
```
##                        mean      sd    min       max     range
```

```
## afflicted_rate      23.37  88.83  0.00  1479.96  1479.96
## reproduction_rate    0.77   0.44 -0.01     2.06     2.08
## new_tests          117.76 460.31  0.00 10142.16 10142.16
## new_vaccinations    70.79 166.06  0.00  1689.38  1689.38
## stringency_index    44.81  25.15  0.00    99.06    99.06
```

```
options(scipen=0)
```

Plotting to see whether there is any irregularity in the distribution of the dependent variable.

```
world_df %>% ggplot(aes(fill=location, y = afflicted_rate/206, x = date)) +
    geom_area(position = "stack", stat = "identity", alpha = 0.5) +
    theme(legend.position="none")
```



```r
fmxdat::source_all("./code")

world_df <- world_df %>% scale_bigs_cumsum(.)
```

2. Scaling

```r
options(scipen=999)
fmxdat::source_all("./code")

cols_range(world_df)
```

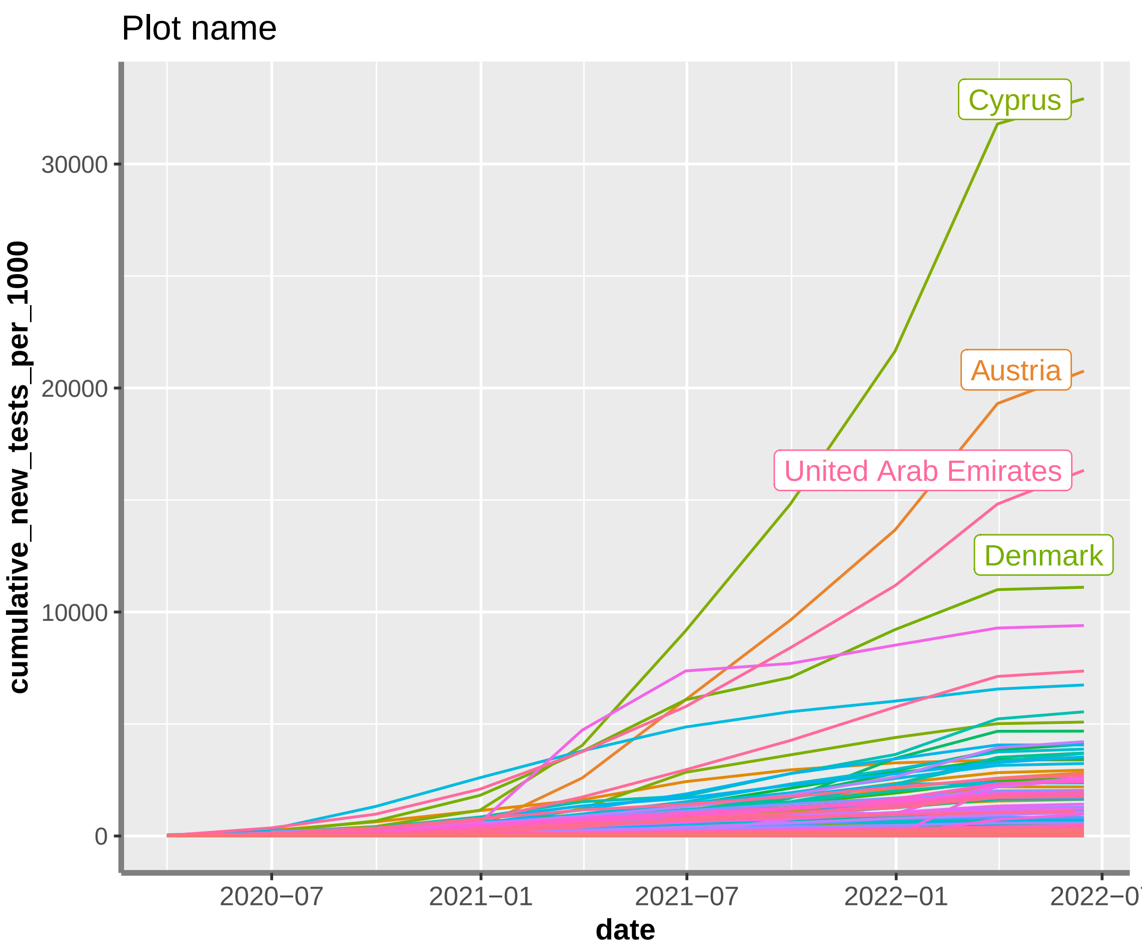
```
##                        mean      sd   min      max    range
## afflicted_rate        23.37   88.83  0.00  1479.96  1479.96
## reproduction_rate      0.77    0.44 -0.01     2.06     2.08
## new_tests            487.10 1789.23  0.00 32919.30 32919.30
## new_vaccinations     275.41  558.44  0.00  3041.92  3041.92
## stringency_index      44.81   25.15  0.00    99.06    99.06
```

```r
options(scipen=0)
```

Thus, want to scale: `new_test`, `new_vaccinations`

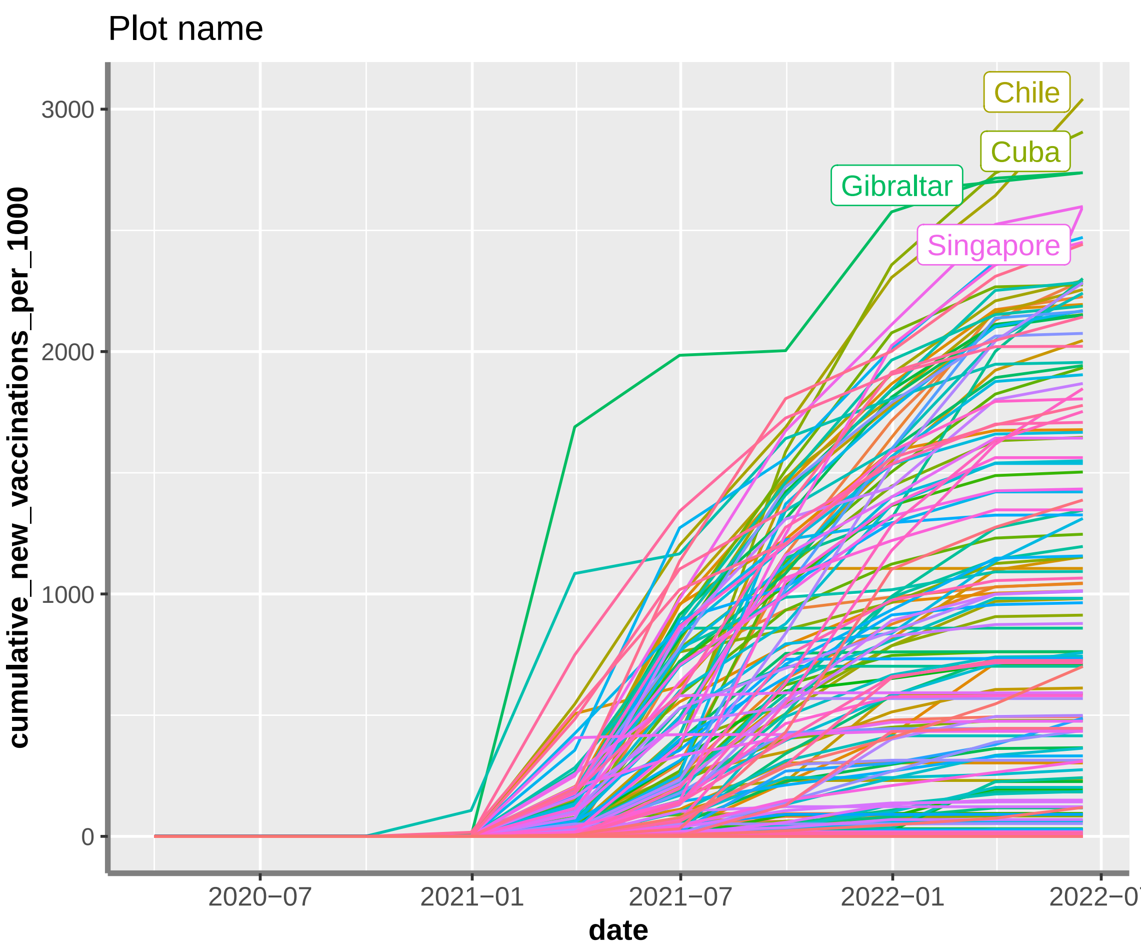
```r
# world_df %>% gather(big, big_cols,
#                       c("icu_patients", "hosp_patients",
#                         "new_tests", "new_vaccinations")) %>%
#     ggplot()




world_df %>% ungroup() %>% group_by(location) %>%
    mutate(label = if_else(date == last(date), as.character(location), NA_character_)) %>%
    # filter(date == last(date)) %>%
    ggplot(aes(x = date, y = new_tests, group = location, col = location)) +
    geom_line() +
```

```
    theme(axis.text.x = element_text(size = 10),
      axis.title = element_text(face = "bold"),
      axis.line = element_line(colour = "grey50", size = 1)) +
    scale_y_continuous("cumulative_new_tests_per_1000") +
    labs(title = "Plot name") +
    geom_label_repel(aes(label = label),
              nudge_x = 1,
              na.rm = TRUE) +
    theme(legend.position="none")
```




```
world_df %>% ungroup() %>% group_by(location) %>%
    mutate(label = if_else(date == last(date), as.character(location), NA_character_)) %>%
    # filter(date == last(date)) %>%
    ggplot(aes(x = date, y = new_vaccinations, group = location, col = location)) +
```

```
    geom_line() +
    theme(axis.text.x = element_text(size = 10),
      axis.title = element_text(face = "bold"),
      axis.line = element_line(colour = "grey50", size = 1)) +
    scale_y_continuous("cumulative_new_vaccinations_per_1000") +
    labs(title = "Plot name") +
    geom_label_repel(aes(label = label),
              nudge_x = 1,
              na.rm = TRUE) +
    theme(legend.position="none")
```




```
world_df <- world_df %>% scale_bigs_scale()
```

2.1. Country specific feature scaling

Now, to check the scales of the features that remain constant per country:

```
cols_range_constant(world_df)
```

```
##                               mean       sd min       max     range
## gdp_per_capita            17697.35 20539.28   0 116935.60 116935.60
## population_density          444.44  2094.60   0  20546.77  20546.77
## median_age                   27.58    12.79   0     48.20     48.20
## aged_65_older                 7.90     6.48   0     27.05     27.05
## extreme_poverty               7.83    16.76   0     77.60     77.60
## cardiovasc_death_rate       226.19   135.19   0    724.42    724.42
## diabetes_prevalence           7.52     4.59   0     23.36     23.36
## handwashing_facilities       21.89    32.72   0     99.00     99.00
## hosp_beds_1k                  2.38     2.51   0     13.80     13.80
## life_expectancy              73.36     9.08   0     86.75     86.75
## human_development_index       0.63     0.28   0      0.96      0.96
## smokers                      14.38    12.75   0     45.95     45.95
```

Additional features that need to be scaled are this

- `gdp_per_capita`
- `population_density`
- `cardiovasc_death_rate`

```
g1 <- world_df %>% ungroup() %>% group_by(location) %>%
    filter(date == last(date)) %>%
    ggplot() +
    geom_point(aes(x = reorder(location, cardiovasc_death_rate, mean), y = cardiovasc_death_rate
    theme(axis.text.x = element_blank(),
      axis.title = element_text(face = "bold"),
      axis.line = element_line(colour = "grey50", size = 1)) +
    scale_y_continuous("Cardiovascular Death Rate") +
    scale_x_discrete("Country") +
```

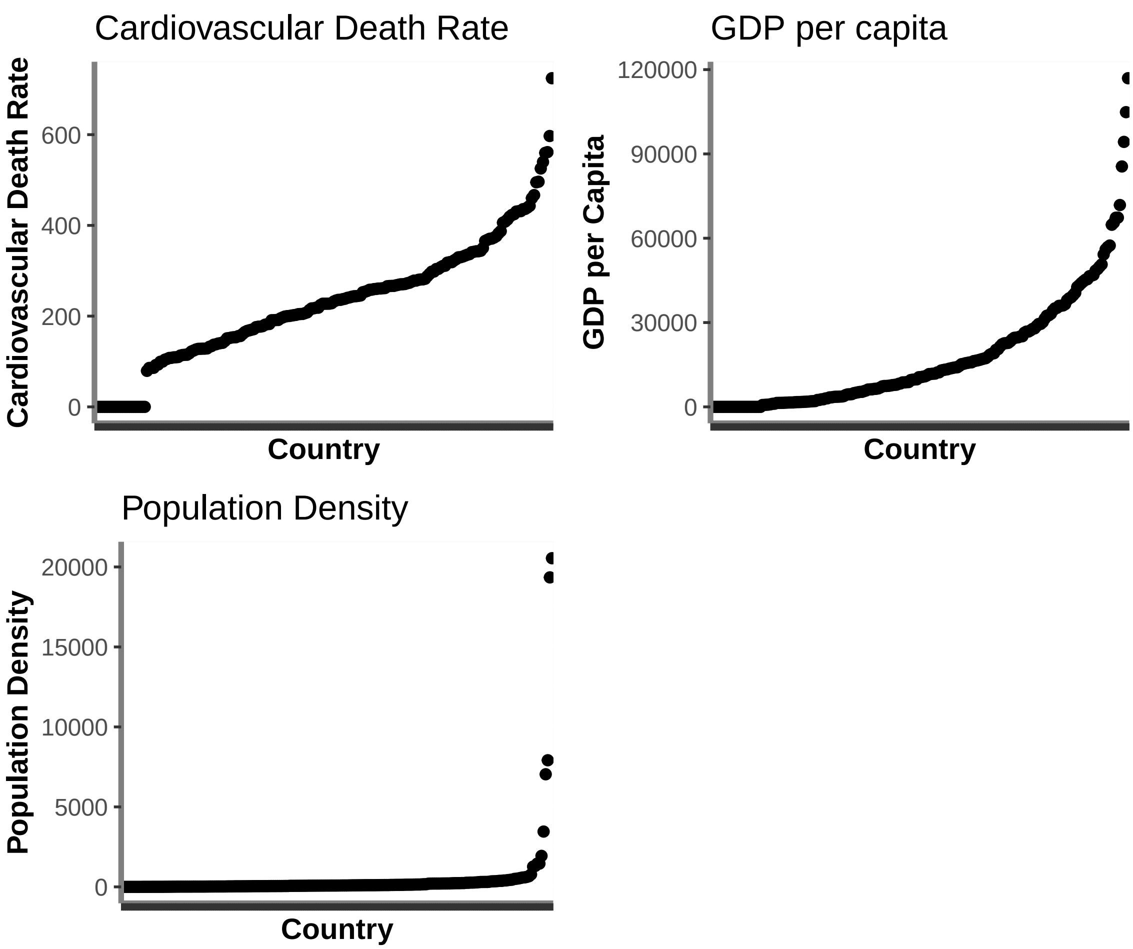
```
    labs(title = "Cardiovascular Death Rate")
         # subtitle = "More or less linear i.e. normalisation scaling")
```

```
g2 <- world_df %>% ungroup() %>% group_by(location) %>%
    filter(date == last(date)) %>%
    ggplot() +
    geom_point(aes(x = reorder(location, gdp_per_capita, mean), y = gdp_per_capita)) +
    theme(axis.text.x = element_blank(),
      axis.title = element_text(face = "bold"),
      axis.line = element_line(colour = "grey50", size = 1)) +
    scale_y_continuous("GDP per Capita") +
        scale_x_discrete("Country") +
    labs(title = "GDP per capita")
         # subtitle = "Nonlinear distribution suggests a\nlog transformation")
```

```
g3 <- world_df %>% ungroup() %>% group_by(location) %>%
    filter(date == last(date)) %>%
    ggplot() +
    geom_point(aes(x = reorder(location, population_density, mean), y = population_density)) +
    theme(axis.text.x = element_blank(),
      axis.title = element_text(face = "bold"),
      axis.line = element_line(colour = "grey50", size = 1)) +
    scale_y_continuous("Population Density") +
    scale_x_discrete("Country") +
    labs(title = "Population Density")
         # subtitle = "Presence of outliers suggests scaling\nsuch that outliers remain\nrelati
```

2.1.1. Plot

```
grid.arrange(g1, g2, g3, nrow=2)
```




```r
world_df <- world_df %>% scale_bigs_constant(.)
```

Now we can check all the descriptive stats for all the columns

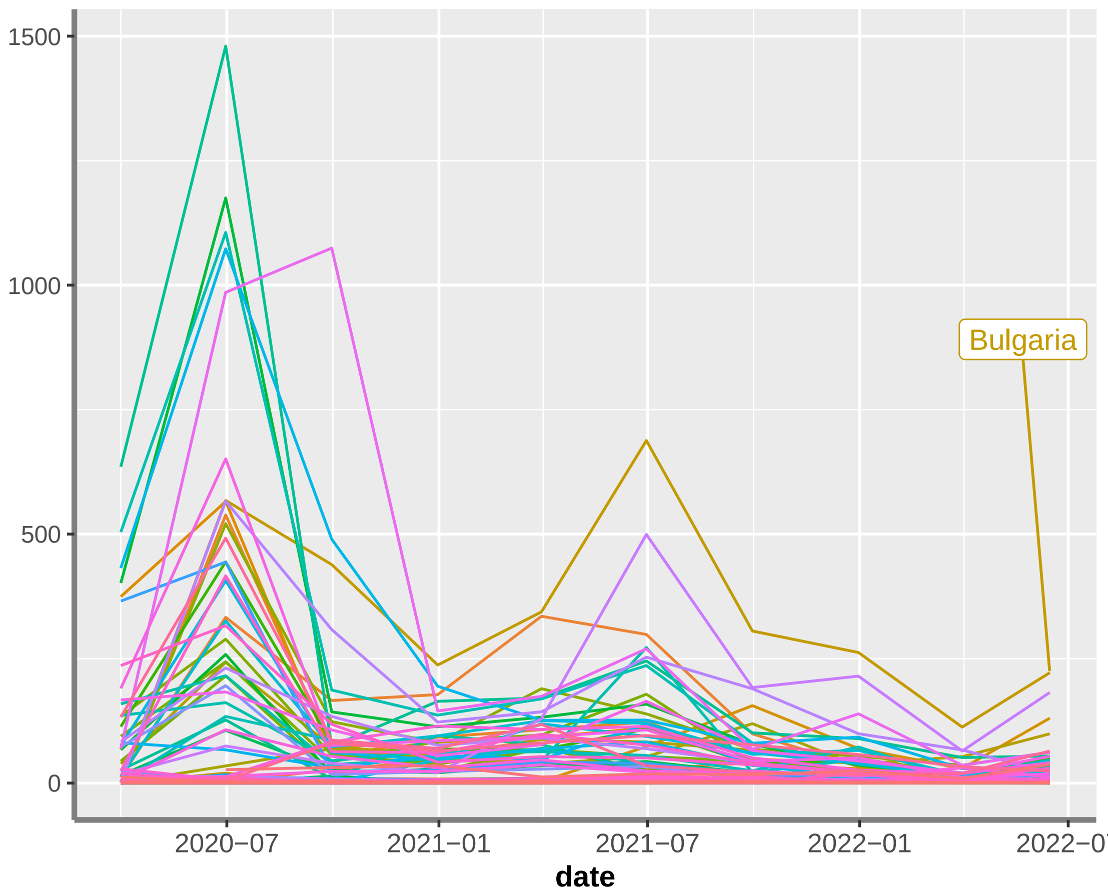
```r
options(scipen=999)
world_df %>% cols_range(df = ., constant_features = c("location", "date"))
```

```
##                          mean    sd   min     max   range
## afflicted_rate          23.37 88.83  0.00 1479.96 1479.96
## reproduction_rate        0.77  0.44 -0.01    2.06    2.08
## stringency_index        44.81 25.15  0.00   99.06   99.06
## median_age              27.58 12.79  0.00   48.20   48.20
## aged_65_older            7.90  6.48  0.00   27.05   27.05
## extreme_poverty          7.83 16.76  0.00   77.60   77.60
```

```
## diabetes_prevalence             7.52  4.59  0.00    23.36    23.36
## handwashing_facilities         21.89 32.72  0.00    99.00    99.00
## hosp_beds_1k                    2.38  2.51  0.00    13.80    13.80
## life_expectancy                73.36  9.08  0.00    86.75    86.75
## human_development_index        62.94 28.48  0.00    95.70    95.70
## smokers                        14.38 12.75  0.00    45.95    45.95
## new_vaccinations_cum_per_1000   0.00  1.00 -0.49     4.95     5.45
## new_tests_cum_per_1000          0.00  1.00 -0.27    18.13    18.40
## population_density_norm         0.00  1.00 -0.21     9.60     9.81
## cardiovasc_death_rate_norm      0.00  1.00 -1.67     3.69     5.36
## gdp_per_capita_log              8.21  3.22  0.00    11.67    11.67
```

```
options(scipen=0)
```

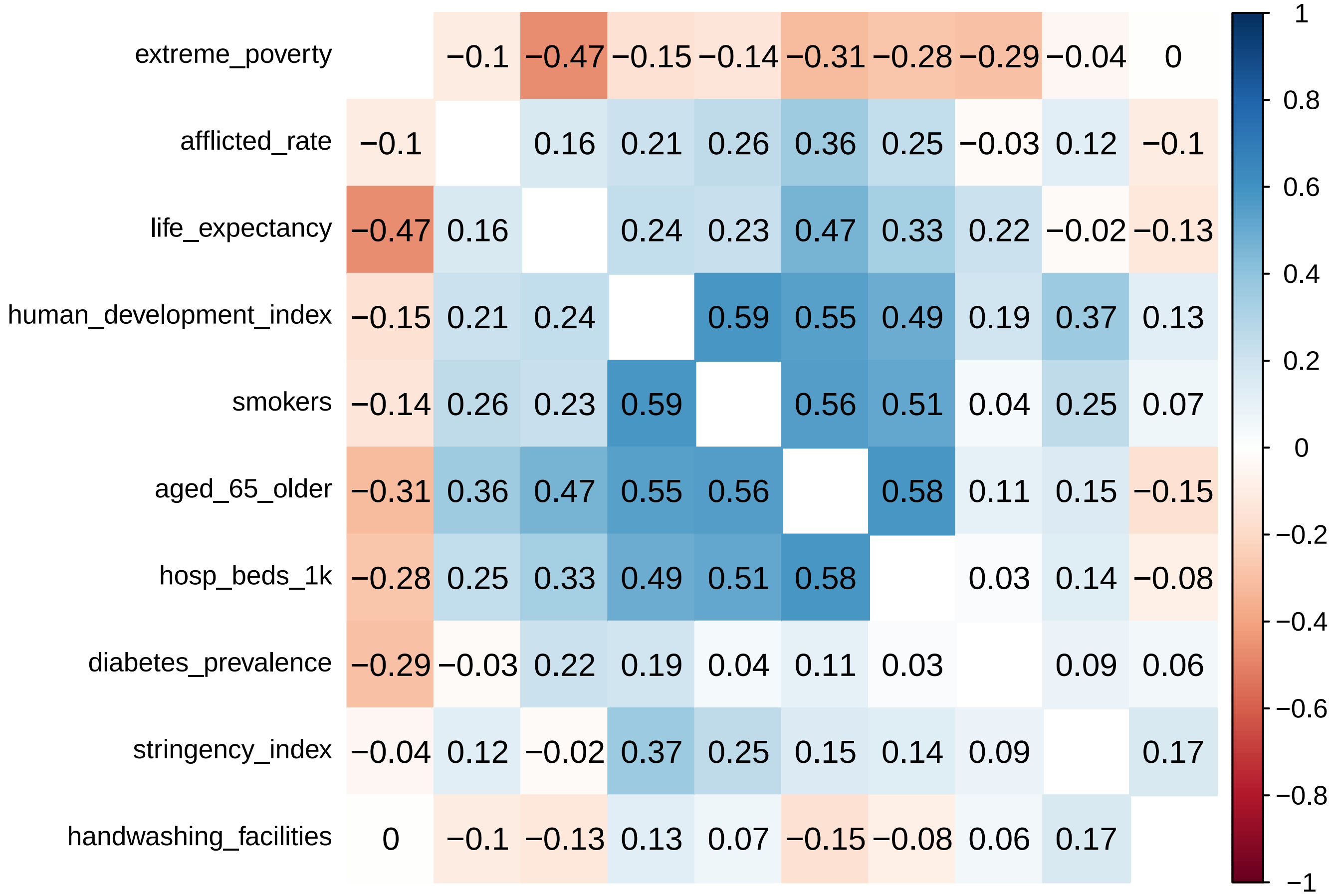
```
world_df %>% ungroup() %>% group_by(location) %>%
    mutate(label = if_else(date == last(date), as.character(location), NA_character_)) %>%
    # filter(date == last(date)) %>%
    ggplot(aes(x = date, y = afflicted_rate, group = location, col = location)) +
    geom_line() +
    theme(axis.text.x = element_text(size = 10),
      axis.title = element_text(face = "bold"),
      axis.line = element_line(colour = "grey50", size = 1)) +
    scale_y_continuous("") +
    labs(title = "") +
    geom_label_repel(aes(label = label),
              nudge_x = 1,
              na.rm = TRUE) +
    theme(legend.position="none")
```



3. Correlation

```
world_df %>% ungroup() %>% select(-c(location, date,
                                     gdp_per_capita_log, population_density_norm,
                                     cardiovasc_death_rate_norm,
                                     reproduction_rate, new_tests_cum_per_1000,
                                     new_vaccinations_cum_per_1000,
                                     median_age)) %>%
    cor(.) %>%
    corrplot(., method = "color", order = "hclust", tl.srt=0, diag = F,
          tl.col = "black", addCoef.col = "black",
          tl.pos = "l",
          tl.cex = 0.8,
```

```
            number.font = 8)
```



4. Regressions

4.1. OLS

```
mod_ols_1 <- plm(afflicted_rate ~ stringency_index +
                  handwashing_facilities,
                   index = c("location", "date"), data = world_df,
                  model = "pooling")

mod_ols_2 <- plm(afflicted_rate ~ stringency_index + smokers
```

```
                 + handwashing_facilities + gdp_per_capita_log +
                 diabetes_prevalence,
                  index = c("location", "date"), data = world_df,
                 model = "pooling")

mod_ols_3 <- plm(afflicted_rate ~ stringency_index + smokers
                 + handwashing_facilities + gdp_per_capita_log
                 + aged_65_older + extreme_poverty + diabetes_prevalence
                 + life_expectancy,
                 data = world_df,
                 index = c("location", "date"), model = "pooling")

robustse_ols1 <- sqrt(diag(vcovHC(mod_ols_1, type = "HC1")))
robustse_ols2 <- sqrt(diag(vcovHC(mod_ols_2, type = "HC1")))
robustse_ols3 <- sqrt(diag(vcovHC(mod_ols_3, type = "HC1")))
```

```
stargazer(mod_ols_1, mod_ols_2, mod_ols_3, header = F, font.size = "footnotesize",
    se = list(robustse_ols1, robustse_ols2, robustse_ols3))
```

Table 4.1

	Dependent variable:		
	afflicted_rate		
	(1)	(2)	(3)
stringency_index	0.503^{***}	0.234^{***}	0.266^{***}
	(0.104)	(0.088)	(0.087)
smokers		1.562^{***}	0.568^{*}
		(0.388)	(0.307)
handwashing_facilities	-0.345^{**}	-0.360^{***}	-0.192
	(0.135)	(0.122)	(0.119)
gdp_per_capita_log		1.850^{***}	-0.226
		(0.650)	(0.967)
aged_65_older			4.027^{***}
			(0.871)
extreme_poverty			-0.060
			(0.064)
diabetes_prevalence		-0.929	-1.293^{*}
		(0.711)	(0.691)
life_expectancy			0.081
			(0.242)
Constant	8.386^{***}	-9.886^{*}	-18.268
	(2.775)	(5.472)	(18.268)
Observations	2,050	2,050	2,050
R^2	0.030	0.092	0.146
Adjusted R^2	0.030	0.089	0.142
F Statistic	32.166^{***} (df = 2; 2047)	41.231^{***} (df = 5; 2044)	43.556^{***} (df = 8; 2041)
Note:			*p<0.1; **p<0.05; ***p<0.01

4.2. Fixed Effects

```r
mod_fe_1 <- plm(afflicted_rate ~ stringency_index + smokers
                 + handwashing_facilities + gdp_per_capita_log
                 + aged_65_older + extreme_poverty + diabetes_prevalence
                 + life_expectancy + human_development_index,
                 data = world_df,
                 index = c("location"),
                model = "within", effect = "individual")


robustse_fe1 <- sqrt(diag(vcovHC(mod_fe_1, type = "HC1")))
```

```r
stargazer(mod_fe_1, header = F, font.size = "small", se = list(robustse_fe1))
```

Table 4.2

	Dependent variable:
	afflicted_rate
stringency_index	0.624***
	(0.136)
Observations	2,050
R^2	0.018
Adjusted R^2	−0.092
F Statistic	33.786*** (df = 1; 1843)
Note:	*p<0.1; **p<0.05; ***p<0.01

4.3. Random Effects

```r
mod_re_1 <- plm(afflicted_rate ~ stringency_index + smokers
                 + handwashing_facilities + gdp_per_capita_log
                 + aged_65_older + extreme_poverty + diabetes_prevalence
                 + life_expectancy + human_development_index
                + reproduction_rate,
                 data = world_df,
```

```
                 index = c("location"),
                model = "random")


robustse_re1 <- sqrt(diag(vcovHC(mod_re_1, type = "HC1")))
```

```
stargazer(mod_re_1, header = F, font.size = "small", se = list(robustse_re1))
```

Table 4.3

	Dependent variable:
	afflicted_rate
stringency_index	0.510***
	(0.112)
smokers	0.549
	(0.336)
handwashing_facilities	−0.204*
	(0.123)
gdp_per_capita_log	−0.558
	(1.098)
aged_65_older	4.055***
	(0.933)
extreme_poverty	−0.046
	(0.073)
diabetes_prevalence	−1.428**
	(0.710)
life_expectancy	0.043
	(0.241)
human_development_index	0.093
	(0.158)
reproduction_rate	−14.911***
	(4.354)
Constant	−16.759
	(18.079)
Observations	2,050
R^2	0.066
Adjusted R^2	0.061
F Statistic	143.344***
Note:	*p<0.1; **p<0.05; ***p<0.01

It is important to note that with a lot of Covid data, the reliability of measurement error and false estimates is questionable. The fixed, and random effects might exacerbate this problem. In this case, it seems that OLS pooled regression, might perform better in explaining the behaviour in the defined ‘afflicted_rate’ variable.