

# Homework 1: Estimating Covid-19 Deaths

*Neeraj Sharma*

*4/7/2020*

## Assignment

Please submit: the answers to the questions, your code, and your dataset. The code you provide should reproduce your model. All of these should be submitted via canvas.

The Governor of Illinois, J. B. Pritzker, has decided that a key input to public policy is knowing how many people will die from Covid-19 in the near future. He has asked you to estimate the total number of official Covid-19 deaths that will be officially recorded in the state of Illinois by April 21 and by May 31.

To fulfill that request, you will need to assemble a data set, do estimation based on that data, and have some sort of theoretical model in your mind to extrapolate out to the future.

- Describe the data set that you chose to assemble and the rationale behind the choices you made in deciding what data to use.
- Describe the model(s) that you settled on for estimation. What was your logic for using that/those particular models?
- Provide an exact number which is your prediction for cumulative official Illinois Covid deaths through April 21
- Provide an exact number which is your prediction for cumulative official Illinois Covid deaths through May 31
- How did you get from the estimates in (2) to the predictions in (3) and (4)?
- Provide an exact number of your best guess of the 90-10 confidence interval for your estimates in questions 3 and 4 (those confidence intervals obviously can be different from one another). A 90-10 confidence interval is the range that would encompass the true value 80 percent of the time. I don't want you necessarily to provide the standard error churned out by the computer, but something more thoughtful. Discuss the thought process/rationale underlying the standard errors you choose.
- Make exactly one pretty picture/graph/slide that you would show to the Governor to allow him to easily understand what he should be expecting in terms of Covid deaths.

## Introduction

In December 2019, scientists in China reported the discovery of a novel coronavirus originating from a wild seafood and exotic animal market in the city of Wuhan, Hubei Province, China. Over the subsequent months, the virus spread over the world, infecting individuals on all populated continents and in nearly every country.<sup>1</sup> The assignment given is to provide a prediction of deaths that might occur by April 21 and by May 31 in the state of Illinois for consideration by JB Pritzker.

```
library(tidyverse)
library(readr)
library(modelr)
library(curl)
library(broom)
library(here)
library(lubridate)
library(fable)
library(tsibble)
```

---

<sup>1</sup><https://www.nytimes.com/article/coronavirus-timeline.html>

```
# For security reasons, my personal API key is hidden. Permission to access Census/ACS data
# to reproduce my results can be granted here: https://api.census.gov/data/key\_signup.html
library(tidycensus)
```

Describe the data set that you chose to assemble and the rationale behind the choices you made in deciding what data to use.

My dataset pulls together data from four sources spread across three general categories. The categories I analyze are:

1. COVID
  - i. January 24, 2020 to March 16, 2020 – Data on cases and deaths as reported by the New York Times
  - ii. March 17, 2020 and onwards – Data on cases, deaths, tested, and negative results from the Illinois Department of Public Health
2. Demographics
  - i. County-level demographic data pulled from the 2018 5-year Census American Community Survey
    - a) Population
    - b) Population under 18
    - c) Population enrolled in school
    - d) Median Income
    - e) Number of Households
    - f) Number of Households with people under 18
    - g) Number of Households with people over 60
    - h) Number of Overcrowded Households
3. Hospitals
  - i. County-level capacity, load and utilization data aggregated to the county level from the Illinois Health Facilities and Services Review Board
4. Mobility
  - i. Median distance traveled by cellphone pings.

```
# Pulls in dataset produced by dataset_creat.R
full <- read_csv("20200413_combined_covid_demos_hosp_mobility.csv") %>%
  mutate(per_capita_deaths = deaths / population-E`,
         `socialdistancing?` = if_else(date > as.Date("2020-03-20"), 1, 0),
         overcrowding = `num_hh_morethan1_person_perroom-E` / `num_hh-E`,
         at_icu_cap = if_else(cases > bedsIntensiveCare, 1, 0),
         at_hospital_cap = if_else(cases > bedsTotalCONAuthorizedBeds, 1, 0),
         delta_cases = if_else(is.na(lag(county)) | county != lag(county), 0, as.double(cases - lag(cases))),
         delta_deaths = if_else(is.na(lag(county)) | county != lag(county), 0, as.double(deaths - lag(deaths))),
         cases_rate = if_else(is.na(lag(county)) | county != lag(county), 0, as.double(delta_cases / lag(delta_cases))),
         deaths_rate = if_else(is.na(lag(county)) | county != lag(county), 0, as.double(delta_deaths / lag(delta_deaths))),
         days_since_first_case = date - ymd("2020-01-24"),
         days_since_first_death = date - ymd("2020-03-17"),
         days_since_sheltering = date - mdy("3/20/2020"))

# Note: April 21, 2020 is 88 days after the first IL COVID case was recorded on 2020-01-24
# Note: May 31, 2020 is 128 days after the first IL COVID case was recorded on 2020-01-24
```

Describe the model(s) that you settled on for estimation. What was your logic for using that/those particular models?

```

highpop <- c("Cook", "DuPage", "Lake", "Will", "Kane", "McHenry", "Winnebago", "Madison", "St. Clair", "
apr <- full %>%
  select(date, county, cases, deaths, delta_cases, delta_deaths, cases_rate, deaths_rate, days_since_fi
    `num_hh-E`, `num_hh_morethan1_person_perroom-E`, bedsIntensiveCare, bedsTotalCONAuthorizedBeds
    at_hospital_cap) %>%
  filter(deaths != 0, county %in% highpop) %>%
  as_tsibble(index = date, key = county)

regres1 <- model(apr, lm = TSLM(log(deaths) ~ date*`socialdistancing?` + at_icu_cap))

## Warning: 1 error encountered for lm
## [1] 0 (non-NA) cases

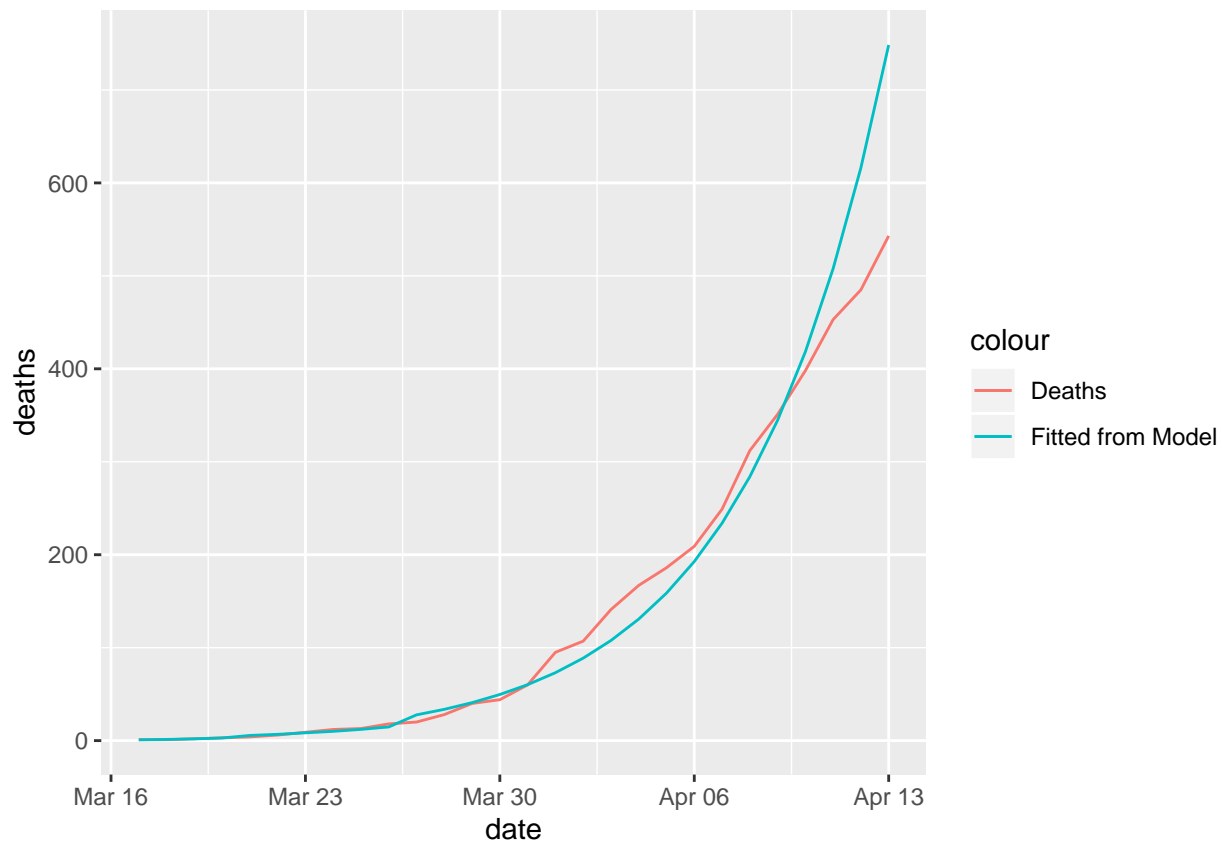
regres2 <- model(apr %>% filter(!is.na(m50))), lm = TSLM(log(deaths) ~ delta_deaths * cases + date + `
## Warning: 1 error encountered for lm
## [1] 0 (non-NA) cases

regres1 %>% filter(county == "Cook") %>% report()

## Series: deaths
## Model: TSLM
## Transformation: log(.x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.33334 -0.12072  0.04576  0.15457  0.26944
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -7.315e+03  1.665e+03  -4.393 0.000231 ***
## date              3.989e-01  9.081e-02   4.393 0.000231 ***
## `socialdistancing?`  3.761e+03  1.674e+03   2.247 0.034980 *
## at_icu_cap        4.396e-01  1.489e-01   2.952 0.007370 **
## date:`socialdistancing?` -2.050e-01  9.126e-02  -2.247 0.035014 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2031 on 22 degrees of freedom
## Multiple R-squared:  0.9912, Adjusted R-squared:  0.9895
## F-statistic: 616.4 on 4 and 22 DF, p-value: < 2.22e-16

ggplot(augment(regres1 %>% filter(county == "Cook")), aes(x = date)) +
  geom_line(aes(y = deaths, color = "Deaths")) +
  geom_line(aes(y = .fitted, color = "Fitted from Model"))

```

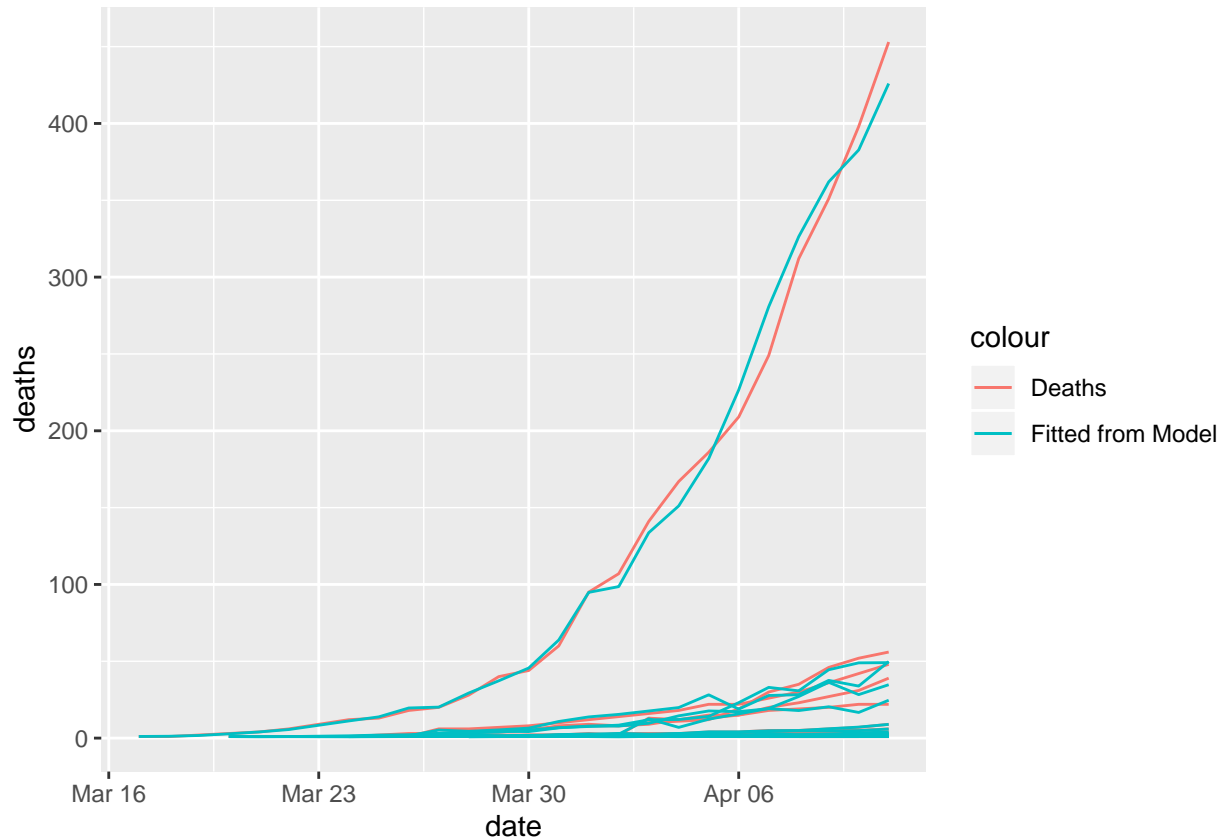


```
regres2 %>% filter(county == "Cook") %>% report()
```

```
## Series: deaths
## Model: TSLM
## Transformation: log(.x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.1836125 -0.0534001 -0.0006865  0.0699137  0.1614388
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -6.539e+03  5.636e+02 -11.603  3.34e-09 ***
## delta_deaths    1.767e-02  4.814e-03   3.672  0.00206 **
## cases         -1.878e-04  5.347e-05  -3.512  0.00289 **
## date           3.566e-01  3.073e-02  11.604  3.34e-09 ***
## `socialdistancing?` 4.792e-01  2.263e-01   2.118  0.05020 .
## m50            2.170e-01  1.180e-01   1.838  0.08465 .
## at_icu_cap     -2.130e-01  9.996e-02  -2.131  0.04896 *
## delta_deaths:cases -1.556e-06  5.257e-07  -2.961  0.00921 **
## `socialdistancing?:m50` -3.071e-01  2.303e-01  -1.333  0.20104
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09733 on 16 degrees of freedom
## Multiple R-squared:  0.9983, Adjusted R-squared:  0.9975
## F-statistic: 1180 on 8 and 16 DF, p-value: < 2.22e-16
```

```
ggplot(augment(regres2), aes(x = date, group = county)) +
  geom_line(aes(y = deaths, color = "Deaths")) +
  geom_line(aes(y = .fitted, color = "Fitted from Model"))
```

## Warning: Removed 14 rows containing missing values (geom\_path).



```
# forecast_newdata <- tsibble(date = seq(ymd("2020-04-14"), ymd("2020-04-21"), by = "1 day"), `socialdi
```

```
global_cases_raw <- read_csv(curl("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/css
```

```
## Parsed with column specification:
```

```
## cols(
##   .default = col_double(),
##   `Province/State` = col_character(),
##   `Country/Region` = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```
global_deaths_raw <- read_csv(curl("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/css
```

```
## Parsed with column specification:
```

```
## cols(
##   .default = col_double(),
##   `Province/State` = col_character(),
##   `Country/Region` = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```

first_reported_cases <- global_cases_raw %>%
  select(-`Province/State`, -Lat, -Long) %>%
  pivot_longer(-`Country/Region`, names_to = "date", values_to = "cases") %>%
  mutate(first_case_date = as.Date(date, "%m/%d/%y")) %>%
  group_by(first_case_date, `Country/Region`) %>%
  summarize(cases = sum(cases)) %>%
  arrange(-desc(`Country/Region`)) %>%
  ungroup() %>%
  filter((cases != 0 & lag(cases) == 0 & cases <= lead(cases)) | (first_case_date == as.Date("2020-01-22"))) %>%
  select(-cases, country = `Country/Region`)

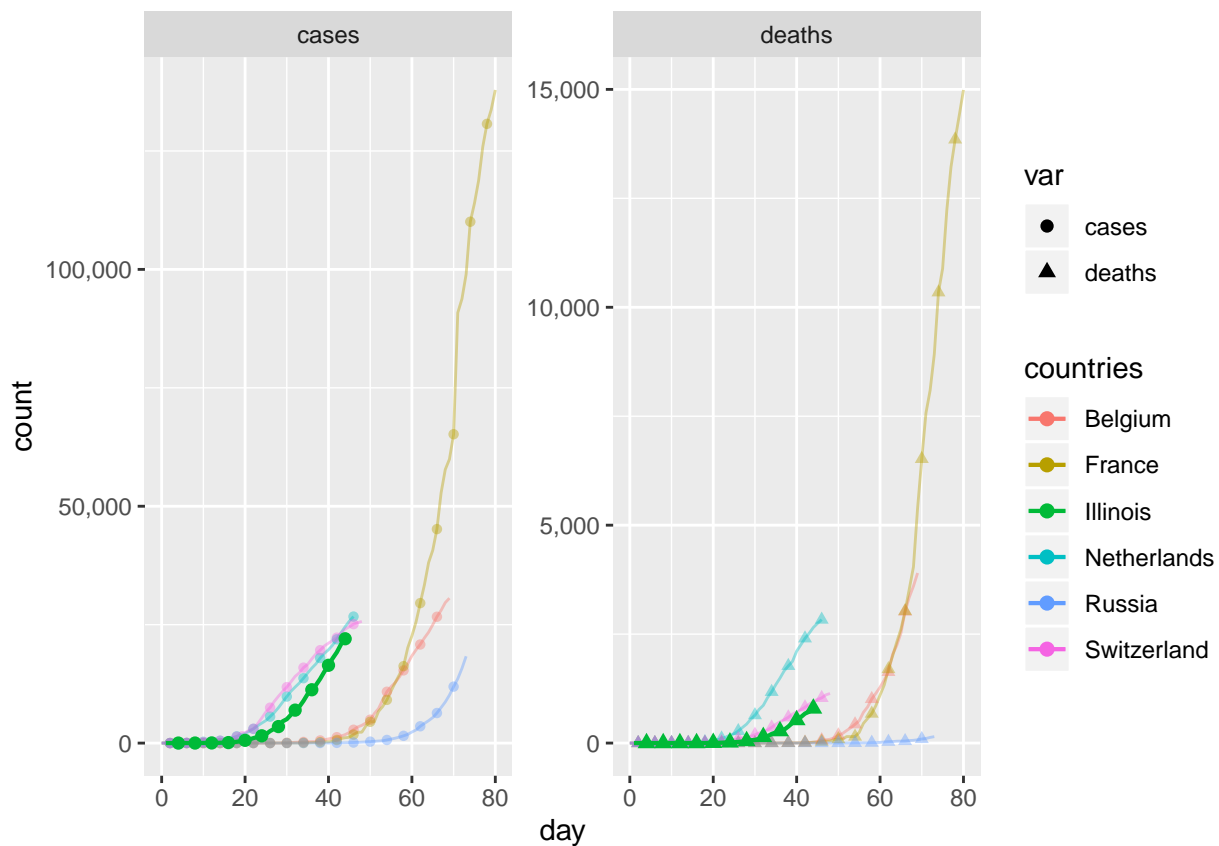
il <- full %>%
# filter(county %in% highpop) %>%
  select(date, cases, deaths, delta_deaths, `socialdistancing?`) %>%
  group_by(date, `socialdistancing?`) %>%
  summarize(deaths = sum(deaths), cases = sum(cases)) %>%
  ungroup() %>%
  filter(cases != 0) %>%
  select(-`socialdistancing?`) %>%
  pivot_longer(-date, names_to = "var", values_to = "count") %>%
  mutate(day = as.double(date - ymd("2020-02-29"))) %>%
  filter(day > 0)

country <- c("Russia", "Netherlands", "Switzerland", "Belgium", "France")

versus <- left_join(global_cases_raw %>%
  filter(`Country/Region` %in% country) %>%
  select(-`Province/State`, -Lat, -Long) %>%
  pivot_longer(-`Country/Region`, names_to = "date", values_to = "cases") %>%
  group_by(`Country/Region`, date) %>%
  summarize(cases = sum(cases)),
  global_deaths_raw %>%
  filter(`Country/Region` %in% country) %>%
  select(-`Province/State`, -Lat, -Long) %>%
  pivot_longer(-`Country/Region`, names_to = "date", values_to = "deaths") %>%
  group_by(`Country/Region`, date) %>%
  summarize(deaths = sum(deaths))) %>%
  filter(cases != 0) %>%
  pivot_longer(-c(`Country/Region`, date), names_to = "var", values_to = "count") %>%
  mutate(date = as.Date(date, "%m/%d/%y")) %>%
  left_join(first_reported_cases, by = c("Country/Region" = "country")) %>%
  mutate(day = as.double(date - first_case_date)) %>%
  rename(countries = `Country/Region`) %>%
  group_by(countries, date, var, day) %>%
  summarize(count = sum(count))

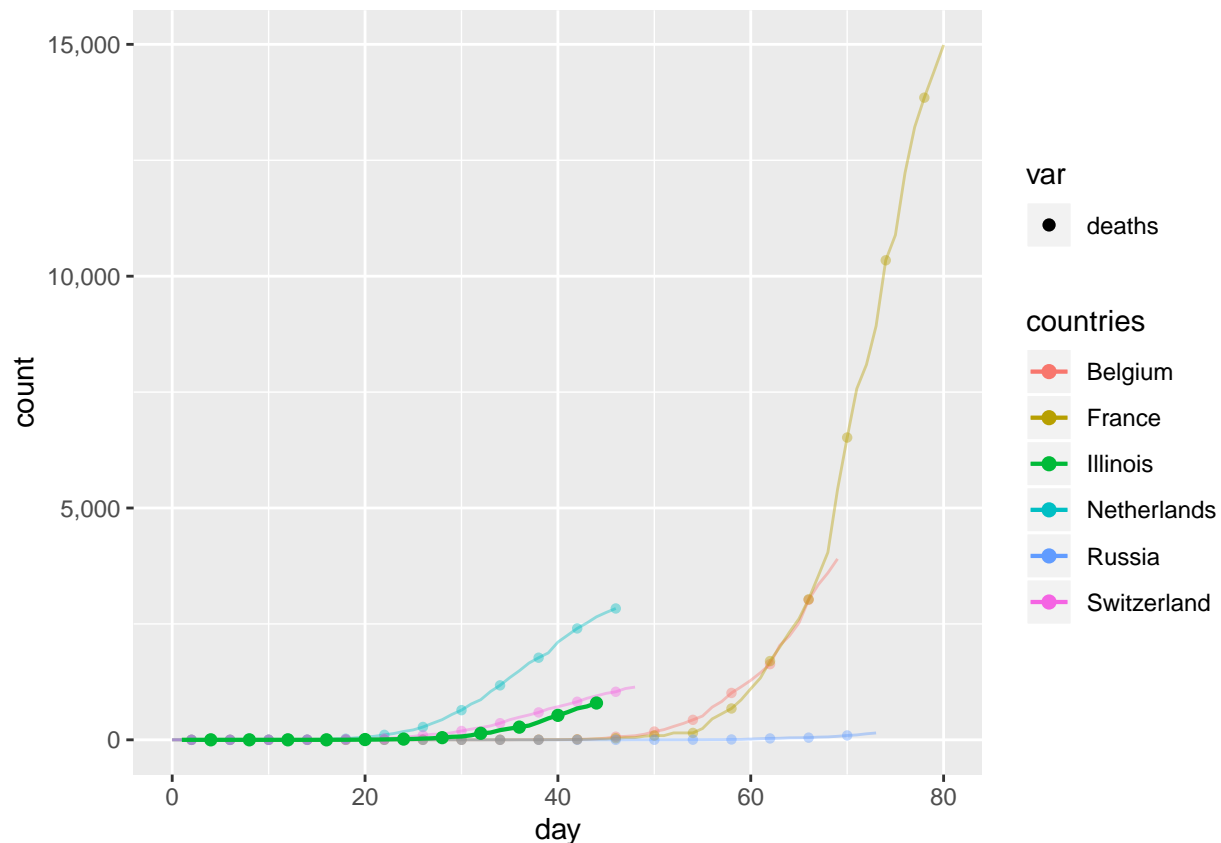
ggplot() +
  geom_line(versus, mapping = aes(x = day, y = count, group = interaction(countries, var), color = countries)) +
  geom_point(versus %>% filter(day %% 4 == 2), mapping = aes(x = day, y = count, group = interaction(countries, var), color = countries)) +
  geom_line(il, mapping = aes(x = day, y = count, group = var, color = "Illinois"), size = .75) +
  geom_point(il %>% filter(day %% 4 == 0), mapping = aes(x = day, y = count, group = var, color = "Illinois")) +
  scale_y_continuous(labels = scales::comma) +
  facet_wrap(~var, scales = "free")

```



```
# geom_line(apr_highpop_forecast, mapping = aes(x = day, y = deaths, color = "prediction"))
```

```
ggplot() +
  geom_line(versus %>% filter(var == "deaths"), mapping = aes(x = day, y = count, group = interaction(c
  geom_point(versus %>% filter(day %% 4 == 2, var == "deaths"), mapping = aes(x = day, y = count, group
  geom_line(il %>% filter(var == "deaths"), mapping = aes(x = day, y = count, group = var, color = "Ill
  geom_point(il %>% filter(day %% 4 == 0, var == "deaths"), mapping = aes(x = day, y = count, group = v
  scale_y_continuous(labels = scales::comma)
```



A Chicago resident contracted the first confirmed case of COVID-19 in Illinois. The Chicago case was announced Jan. 24, and the first cases outside of Chicago and Cook County were reported March 11, in Kane and McHenry counties. Governor Pritzker issued a disaster proclamation on March 9, which provides the state access to federal and state resources to mitigate the spread of COVID-19.

On Jan 24, IL had it's first COVID patients. Those two individuals had literally recovered by the time another patient was diagnosed in IL. I remove the days that there were those two random early stage cases and instead start my day 0 for Illinois on February 29, 2020. On February 29, 2020, the first COVID patient in IL not affiliated with the two early cases was diagnosed.<sup>2</sup> I count that as the real first COVID case in IL and thus make it in the initial start date for IL.

Additionally, I also move Russia's curve to begin on March 2, 2020. Prior to that date, the only cases in Russia were Chinese Nationals who had recently entered Russia from China and eight Russians from the Diamond Princess Cruise Ship who were immediately quarantined. All these individuals recovered prior to another case appearing in Russia. On March 2nd, a man was diagnosed with COVID after vacationing in Italy, so I mark that as the first Russian COVID case.

<sup>2</sup><https://wgntv.com/news/illinois-health-officials-announce-new-case-of-coronavirus/>