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)?
- You don't have to provide exact numbers, but discuss what you think the standard errors associated with your estimates might be, and your rationale for thinking those would be the standard errors.
- 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 Provence, China. Over the subsequent months, the virus spread over the world, infecting individuals on all populated continents and in nearly every country.[1] 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.

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


# 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)
```

[1] https://www.nytimes.com/article/coronavirus-timeline.html

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 Survery
        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.

```r
# 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`) %>%
  mutate(`socialdistancing?` = if_else(date > as.Date("2020-03-20"), 1, 0)) %>%
  mutate(overcrowding = `num_hh_morethan1_person_perroom-E`/`num_hh-E`)

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

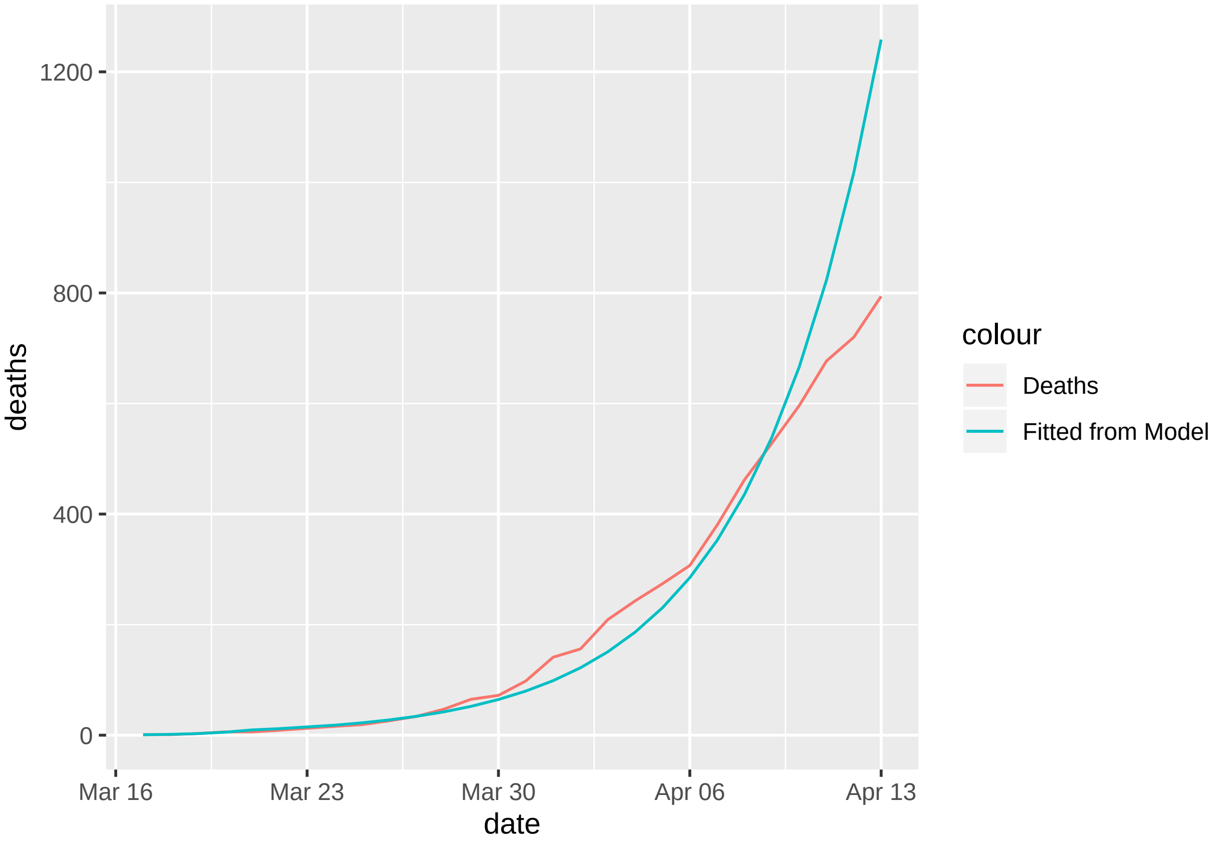
```r
# Hospital data
# https://hifld-geoplatform.opendata.arcgis.com/datasets/6ac5e325468c4cb9b905f1728d6fbf0f_0?selectedAtt
# https://www.chicagobusiness.com/static/section/hospital-beds-database.html

# mobility data
#https://www.google.com/covid19/mobility/
#https://github.com/vitorbaptista/google-covid19-mobility-reports
#https://ai.googleblog.com/2019/11/new-insights-into-human-mobility-with.html
#https://www.nature.com/articles/s41467-019-12809-y

# date implemented social distancing in each county maybe?
# https://www.finra.org/rules-guidance/key-topics/covid-19/shelter-in-place
```

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

```r
apr <- full %>%
  select(date, deaths, `socialdistancing?`) %>%
  group_by(date, `socialdistancing?`) %>%
  summarize(deaths = sum(deaths)) %>%
  ungroup() %>%
  filter(deaths != 0) %>%
  as_tsibble(index = date)

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

apr_regres %>% report()
```

```
## Series: deaths
## Model: TSLM
## Transformation: log(.x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.46575 -0.14162  0.05884  0.18930  0.35728
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)               -1.187e+04  2.077e+03  -5.715 8.07e-06 ***
## date                       6.474e-01  1.133e-01   5.715 8.06e-06 ***
## `socialdistancing?`        7.983e+03  2.082e+03   3.833 0.000850 ***
## date:`socialdistancing?` -4.352e-01  1.135e-01  -3.833 0.000851 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2533 on 23 degrees of freedom
## Multiple R-squared: 0.9863,  Adjusted R-squared: 0.9845
## F-statistic: 550.1 on 3 and 23 DF, p-value: < 2.22e-16
```

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

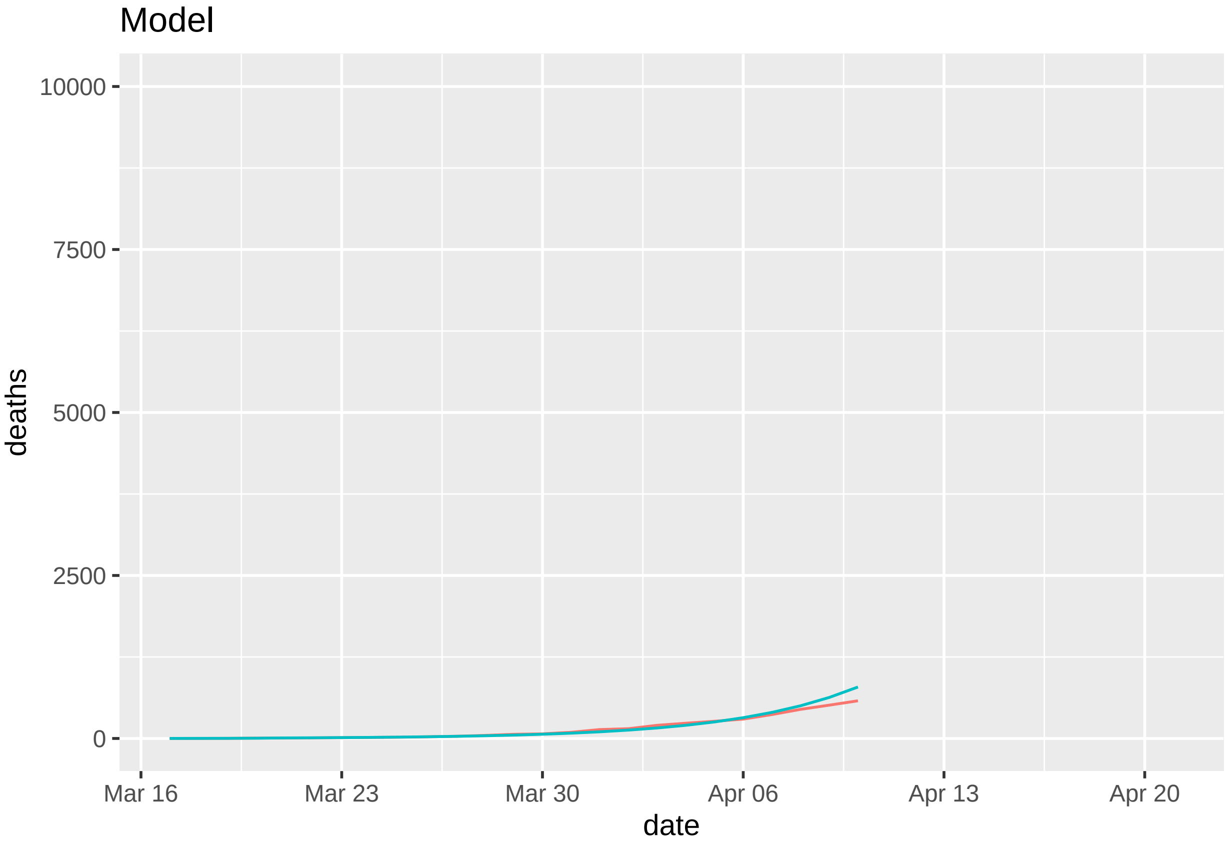


```
# Population of county over 100,000

highpop <- c("Cook", "DuPage","Lake", "Will", "Kane", "McHenry", "Winnebago", "Madison", "St. Clair", "

apr_highpop <- full %>%
  filter(county %in% highpop) %>%
  select(date, deaths, `socialdistancing?`, m50) %>%
  group_by(date, `socialdistancing?`) %>%
  drop_na() %>%
  summarize(deaths = sum(deaths), distance = mean(m50)) %>%
  ungroup() %>%
  filter(deaths != 0) %>%
  as_tsibble(index = date)

apr_highpop_regres <- model(apr_highpop, tslm = TSLM(log(deaths) ~ date * `socialdistancing?`))

apr_highpop_regres %>% report()
```

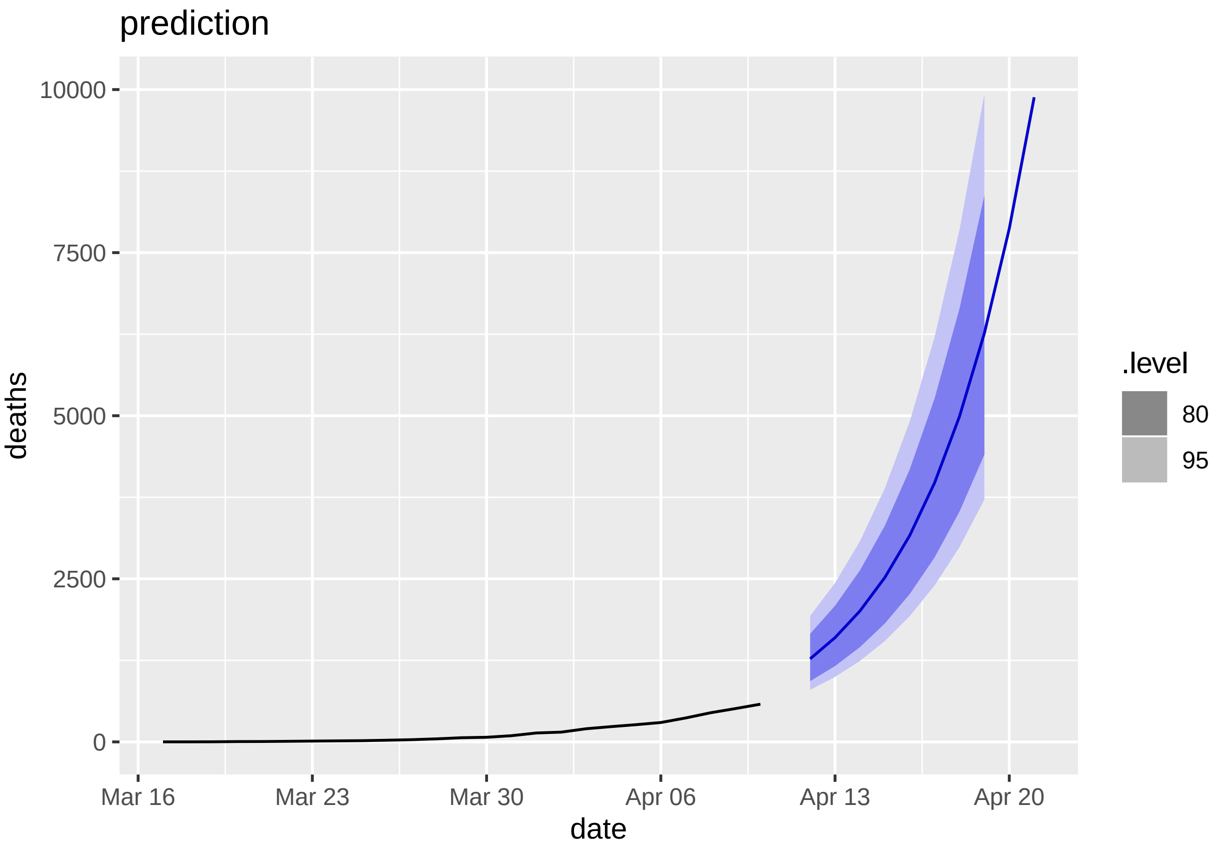
```
## Series: deaths
## Model: TSLM
## Transformation: log(.x)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.344187 -0.125866  0.009134  0.151201  0.289738
##
## Coefficients:
```

```
##                              Estimate Std. Error t value Pr(>|t|)    
## (Intercept)                -1.013e+04  1.656e+03  -6.115 5.64e-06 ***
## date                        5.522e-01  9.029e-02   6.115 5.63e-06 ***
## `socialdistancing?`         5.969e+03  1.662e+03   3.592  0.00182 ** 
## date:`socialdistancing?` -3.254e-01  9.061e-02  -3.591  0.00183 ** 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2019 on 20 degrees of freedom
## Multiple R-squared: 0.9909,  Adjusted R-squared: 0.9895 
## F-statistic: 726.5 on 3 and 20 DF,  p-value: < 2.22e-16
```

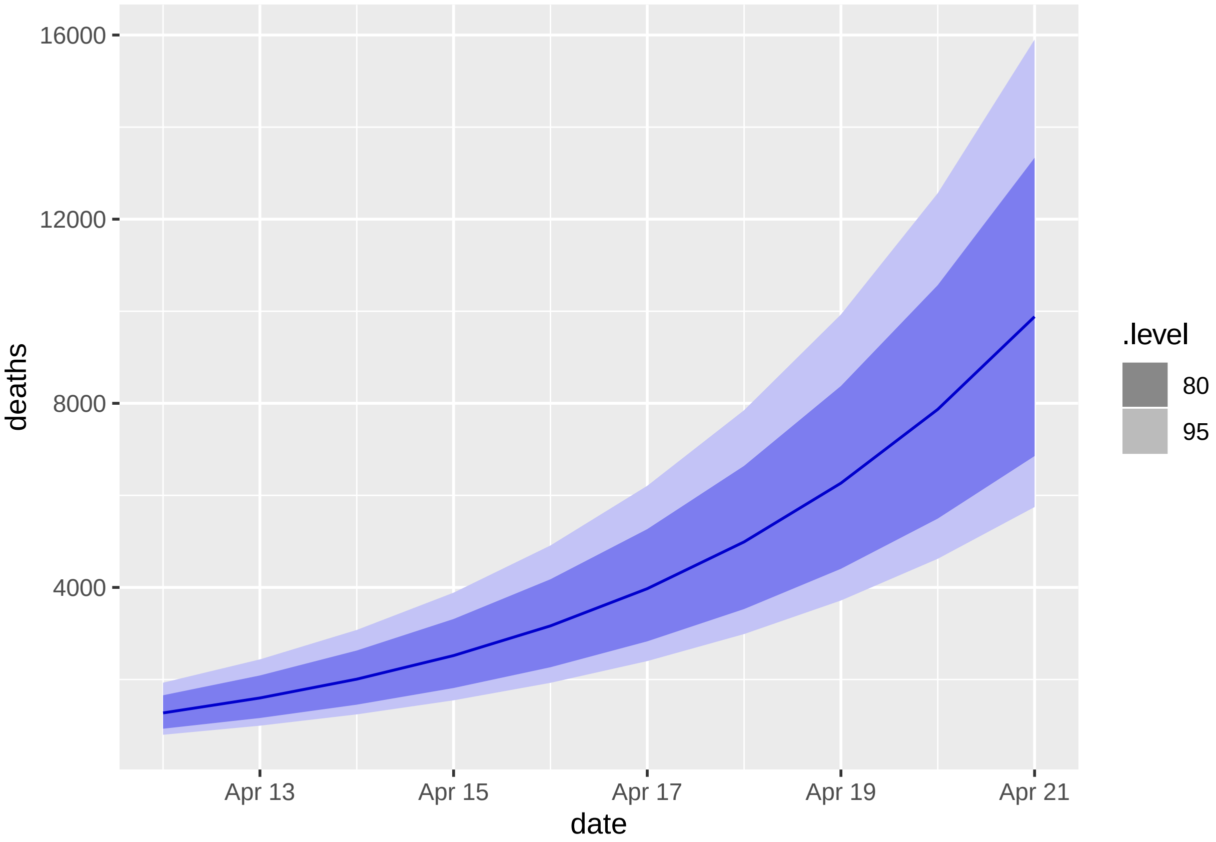
```
ggplot(augment(apr_highpop_regres), aes(x = date)) +
  geom_line(aes(y = deaths, color = "Deaths")) +
  geom_line(aes(y = .fitted, color = "Fitted from Model")) +
  labs(title = "Model") +
  xlim(as.Date("2020-03-17"), as.Date("2020-04-21")) +
  ylim(0, 10000) +
  theme(legend.position = "none")
```



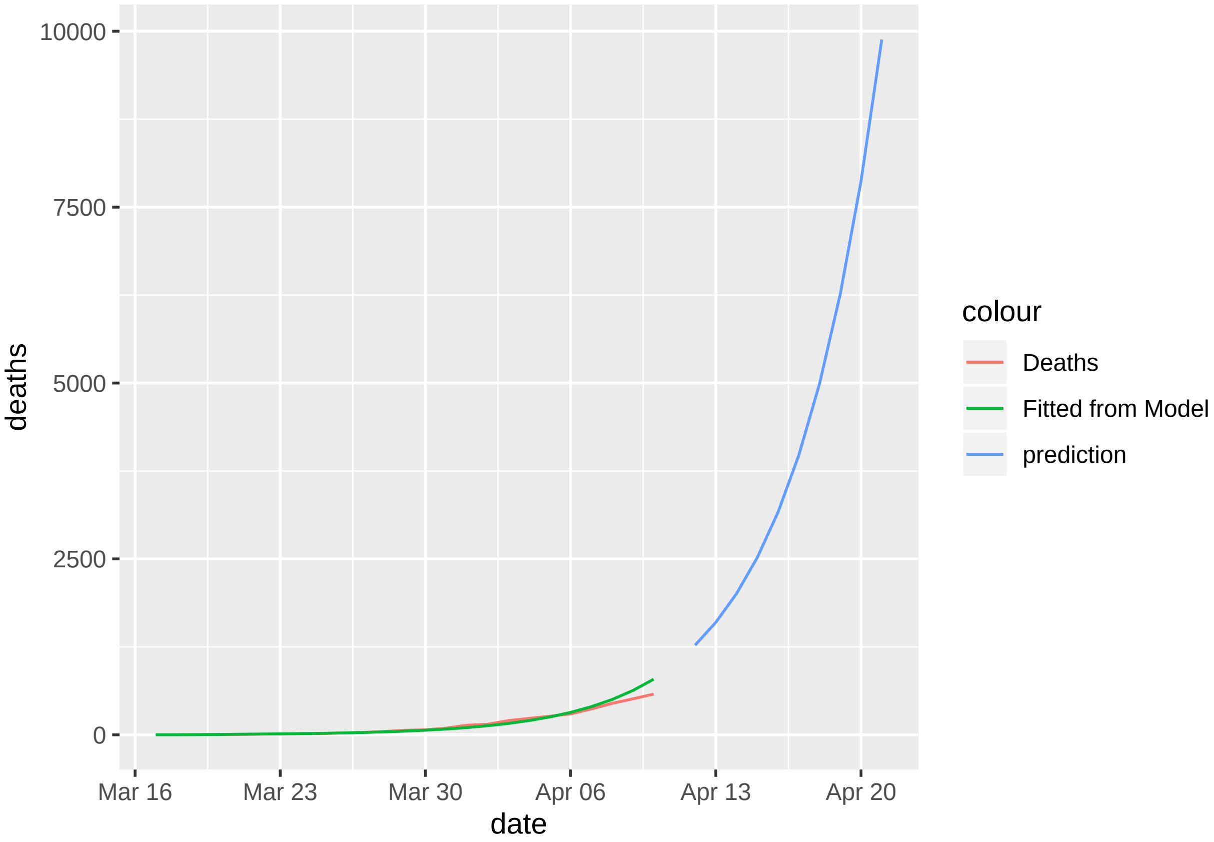
```
apr_highpop_forecast <- forecast(new_data = tsibble::tsibble(date = seq(ymd('2020-04-12'),ymd('2020-04-
  mutate(day = date - ymd("2020-01-24"))


autoplot(apr_highpop_forecast, apr_highpop) +
  xlim(as.Date("2020-03-17"), as.Date("2020-04-21")) +
  ylim(0, 10000) +
  labs(title = "prediction")
```



```r
apr_highpop_forecast %>%
  autoplot()
```



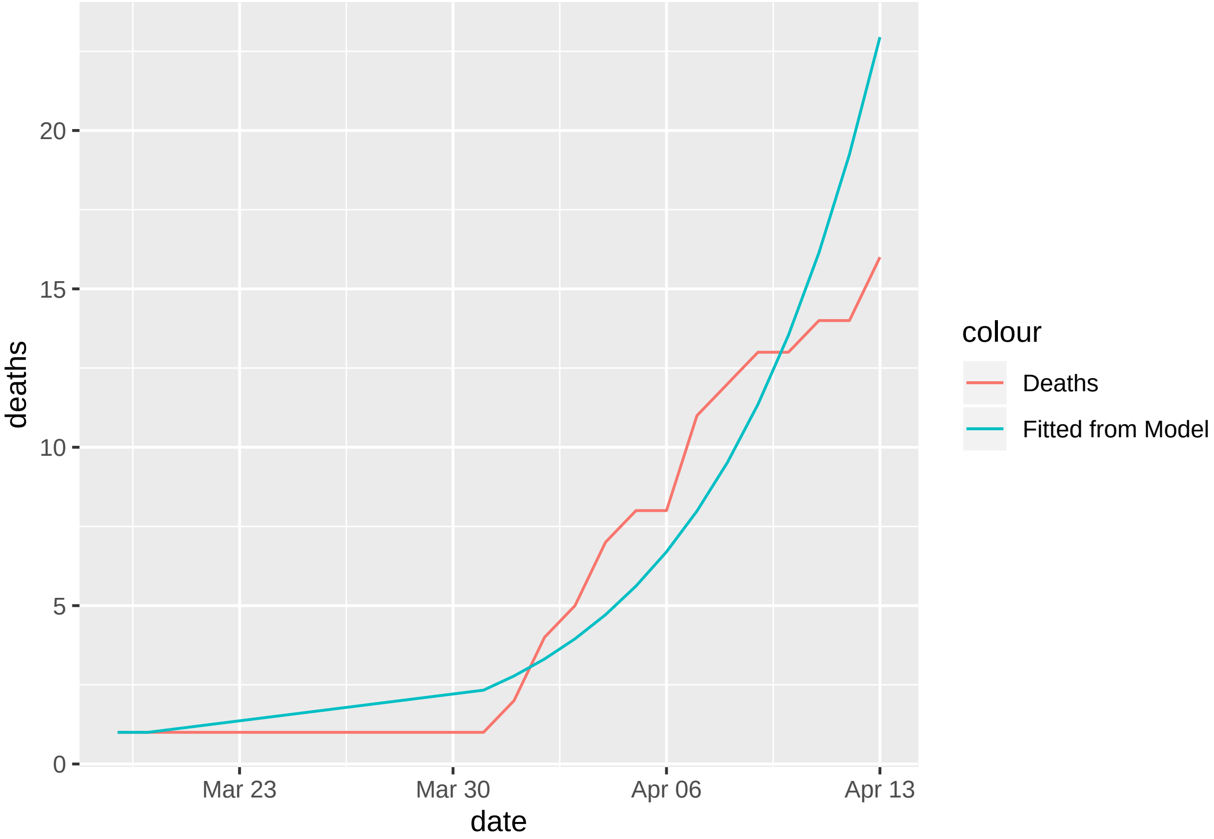
```
ggplot() +
  geom_line(augment(apr_highpop_regres), mapping = aes(x = date, y = deaths, color = "Deaths")) +
  geom_line(augment(apr_highpop_regres), mapping = aes(x = date, y = .fitted, color = "Fitted from Model
  geom_line(apr_highpop_forecast, mapping = aes(x = date, y = deaths, color = "prediction"))
```



```
# social distancing is not correlated with predicitng deaths becsue

apr_lowpop <- full %>%
  filter(!(county %in% highpop)) %>%
  select(date, deaths, `socialdistancing?`) %>%
  group_by(date, `socialdistancing?`) %>%
  summarize(deaths = sum(deaths)) %>%
  ungroup() %>%
  filter(deaths != 0) %>%
  as_tsibble(index = date)

apr_lowpop_regres <- model(apr_lowpop, lm = TSLM(log(deaths) ~ date * `socialdistancing?`))

apr_lowpop_regres %>% report()
```

```
## Series: deaths
## Model: TSLM
## Transformation: log(.x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.84633 -0.18634  0.06768  0.23227  0.39591
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)               -6.271e-08  9.636e+03   0.000    1.000
## date                       3.419e-12  5.254e-01   0.000    1.000
## `socialdistancing?`      -3.228e+03  9.647e+03  -0.335    0.744
## date:`socialdistancing?`  1.759e-01  5.260e-01   0.334    0.744
##
## Residual standard error: 0.3715 on 12 degrees of freedom
## Multiple R-squared: 0.894,   Adjusted R-squared: 0.8675
## F-statistic: 33.74 on 3 and 12 DF, p-value: 3.9654e-06
```

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



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

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   `Province/State` = col_character(),
##   `Country/Region` = col_character()
## )
```

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

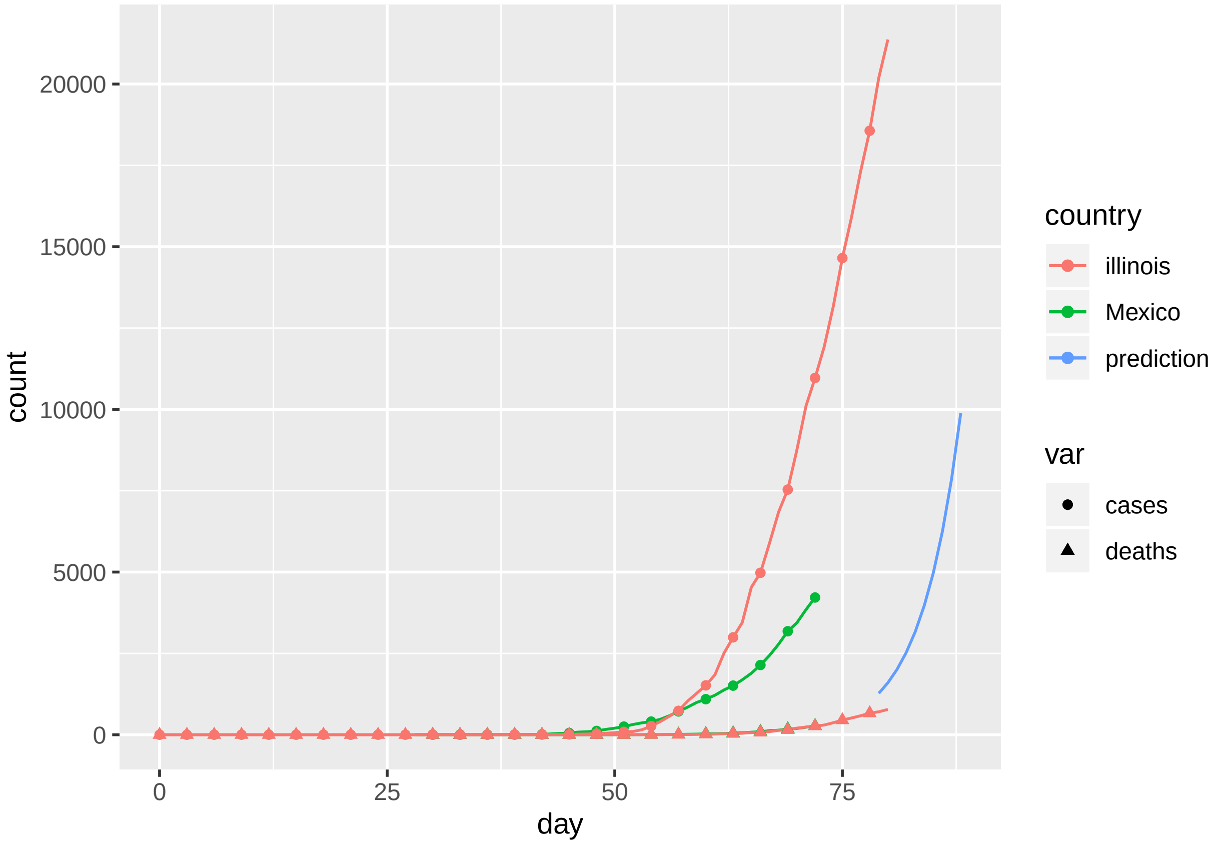
```
global_deaths_raw <- read_csv(curl("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/css
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   `Province/State` = col_character(),
##   `Country/Region` = col_character()
## )
## See spec(...) for full column specifications.
```

```
country <- "Mexico"


versus <- left_join(global_cases_raw %>%
  filter(`Country/Region` == country) %>%
  select(-`Province/State`, -Lat, -Long) %>%
  pivot_longer(-`Country/Region`, names_to = "date", values_to = "cases"), global_deaths_raw %>%
  filter(`Country/Region` == country) %>%
  select(-`Province/State`, -Lat, -Long) %>%
  pivot_longer(-`Country/Region`, names_to = "date", values_to = "deaths")) %>%
  filter(cases != 0) %>%
  pivot_longer(-c(`Country/Region`, date), names_to = "var", values_to = "count") %>%
  mutate(date = as.Date(date, "%m/%d/%y")) %>%
  select(-`Country/Region`) %>%
  mutate(day = as.double(date - ymd("2020-01-31")))
```

```
## Joining, by = c("Country/Region", "date")
```

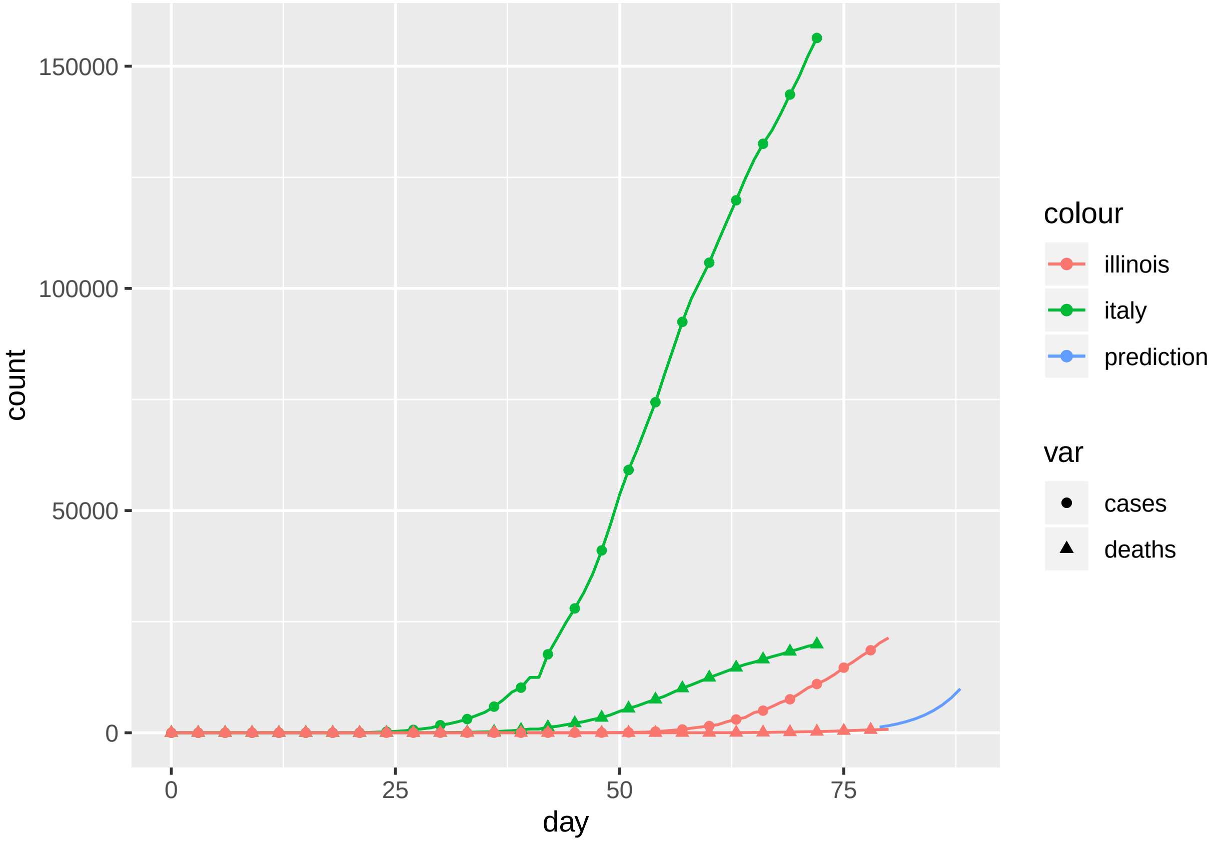
```
il <- full %>%
  filter(county %in% highpop) %>%
  select(date, cases, deaths, delta_cases, delta_deaths, `socialdistancing?`) %>%
  group_by(date, `socialdistancing?`) %>%
  summarize(deaths = sum(deaths), cases = sum(cases)) %>%
  ungroup() %>%
  filter(cases != 0) %>%
  select(-`socialdistancing?`) %>%
  pivot_longer(-date, names_to = "var", values_to = "count")  %>%
  mutate(day = as.double(date - ymd("2020-01-24")))

ggplot() +
  geom_line(versus, mapping = aes(x = day, y = count, group = var, color = country)) +
  geom_line(il, mapping = aes(x = day, y = count, group = var, color = "illinois")) +
  geom_point(versus %>% filter(day %% 3 == 0), mapping = aes(x = day, y = count, group = var, color = co
  geom_point(il %>% filter(day %% 3 == 0), mapping = aes(x = day, y = count, group = var, color = "illi
  geom_line(apr_highpop_forecast, mapping = aes(x = day, y = deaths, color = "prediction"))
```



```r
# compare to italy deaths

italy_cases <- global_cases_raw %>%
  filter(`Country/Region` == "Italy") %>%
  select(-`Province/State`, -Lat, -Long) %>%
  pivot_longer(-`Country/Region`, names_to = "date", values_to = "cases")

italy_deaths <- global_deaths_raw %>%
  filter(`Country/Region` == "Italy") %>%
  select(-`Province/State`, -Lat, -Long) %>%
  pivot_longer(-`Country/Region`, names_to = "date", values_to = "deaths")

italy <- left_join(italy_cases, italy_deaths) %>%
  filter(cases != 0) %>%
  pivot_longer(-c(`Country/Region`, date), names_to = "var", values_to = "count") %>%
  mutate(date = as.Date(date, "%m/%d/%y")) %>%
  select(-`Country/Region`) %>%
  mutate(day = as.double(date - ymd("2020-01-31")))
```

```
## Joining, by = c("Country/Region", "date")
```

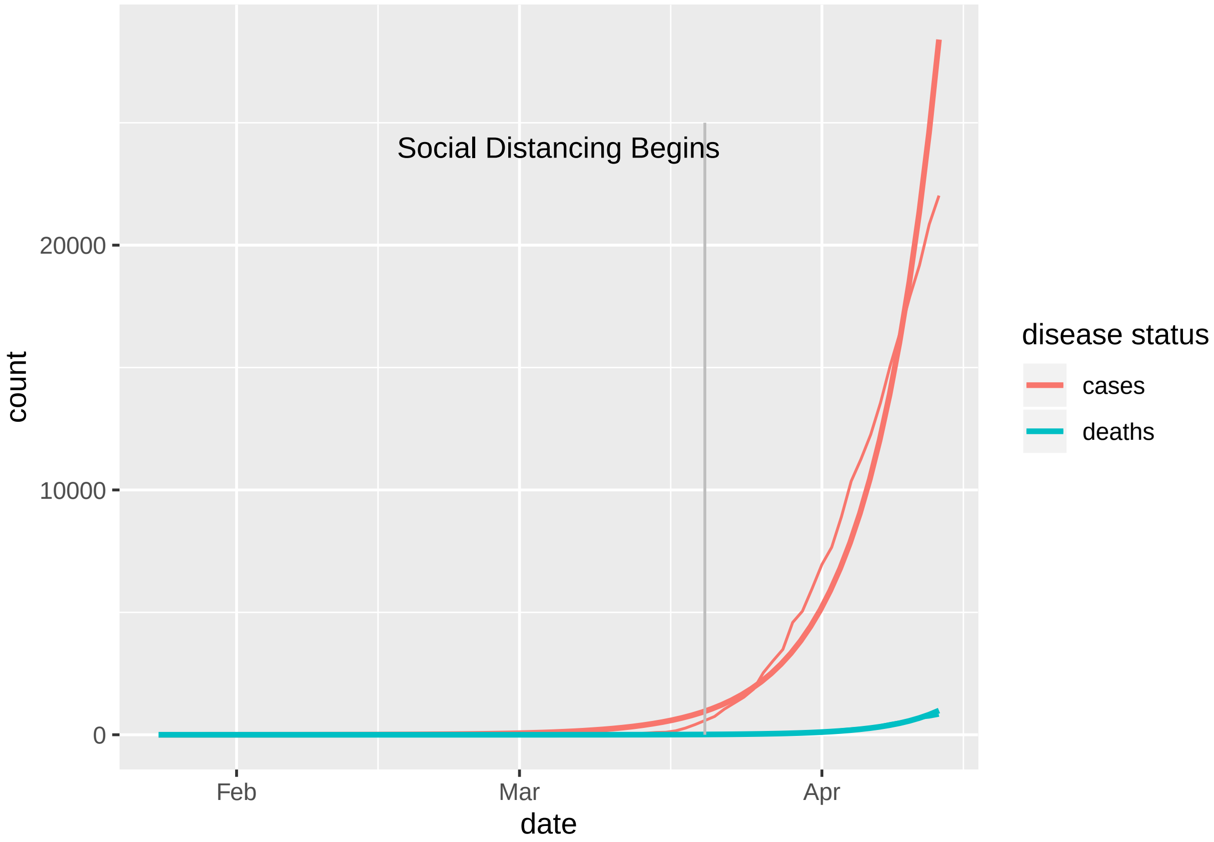
```r
il <- full %>%
  filter(county %in% highpop) %>%
  select(date, cases, deaths, delta_cases, delta_deaths, `socialdistancing?`) %>%
  group_by(date, `socialdistancing?`) %>%
  summarize(deaths = sum(deaths), cases = sum(cases)) %>%
  ungroup() %>%
  filter(cases != 0) %>%
  select(-`socialdistancing?`) %>%
  pivot_longer(-date, names_to = "var", values_to = "count")  %>%
  mutate(day = as.double(date - ymd("2020-01-24")))

ggplot() +
  geom_line(italy, mapping = aes(x = day, y = count, group = var, color = "italy")) +
  geom_line(il, mapping = aes(x = day, y = count, group = var, color = "illinois")) +
  geom_point(italy %>% filter(day %% 3 == 0), mapping = aes(x = day, y = count, group = var, color = "it
  geom_point(il %>% filter(day %% 3 == 0), mapping = aes(x = day, y = count, group = var, color = "illin
  geom_line(apr_highpop_forecast, mapping = aes(x = day, y = deaths, color = "prediction"))
```



```
fit <- augment(apr_highpop_regres)

il_highpop <- full_join(as_tibble(apr_highpop_forecast), fit) %>%
  arrange(desc(date)) %>%
  select(-.model, -`socialdistancing?`)
```

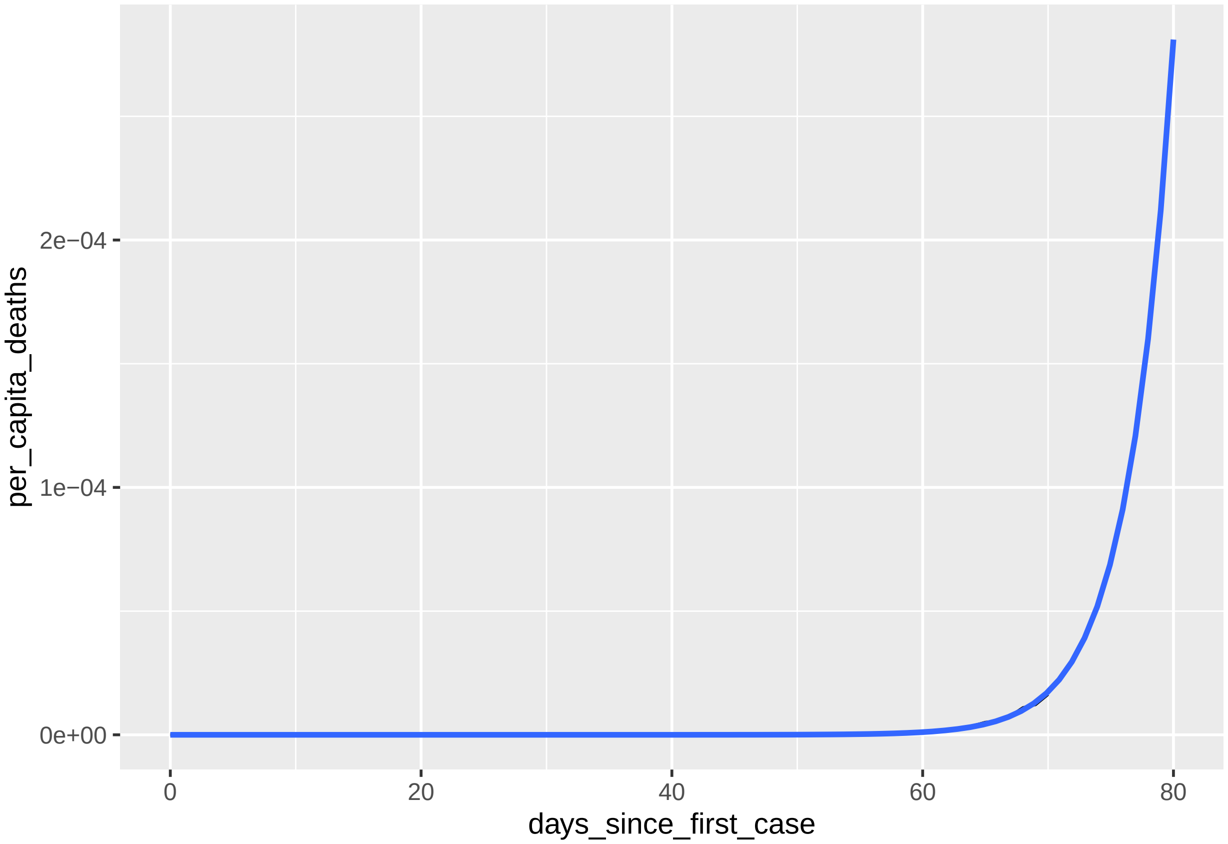
```
## Joining, by = c(".model", "date", "deaths")
```

```
ggplot(full %>% group_by(date) %>% summarise_at(c("cases", "deaths"), sum) %>% pivot_longer(-date, names
  geom_line() +
  geom_smooth(method = "glm",
              method.args = list(family = "poisson"),
              se = FALSE) +
  annotate("segment", x = as.Date("2020-03-20"), xend = as.Date("2020-03-20"), y = 0, yend = 25000, colo
  annotate("text", x = as.Date("2020-03-05"), y = 24000, label = "Social Distancing Begins")
```



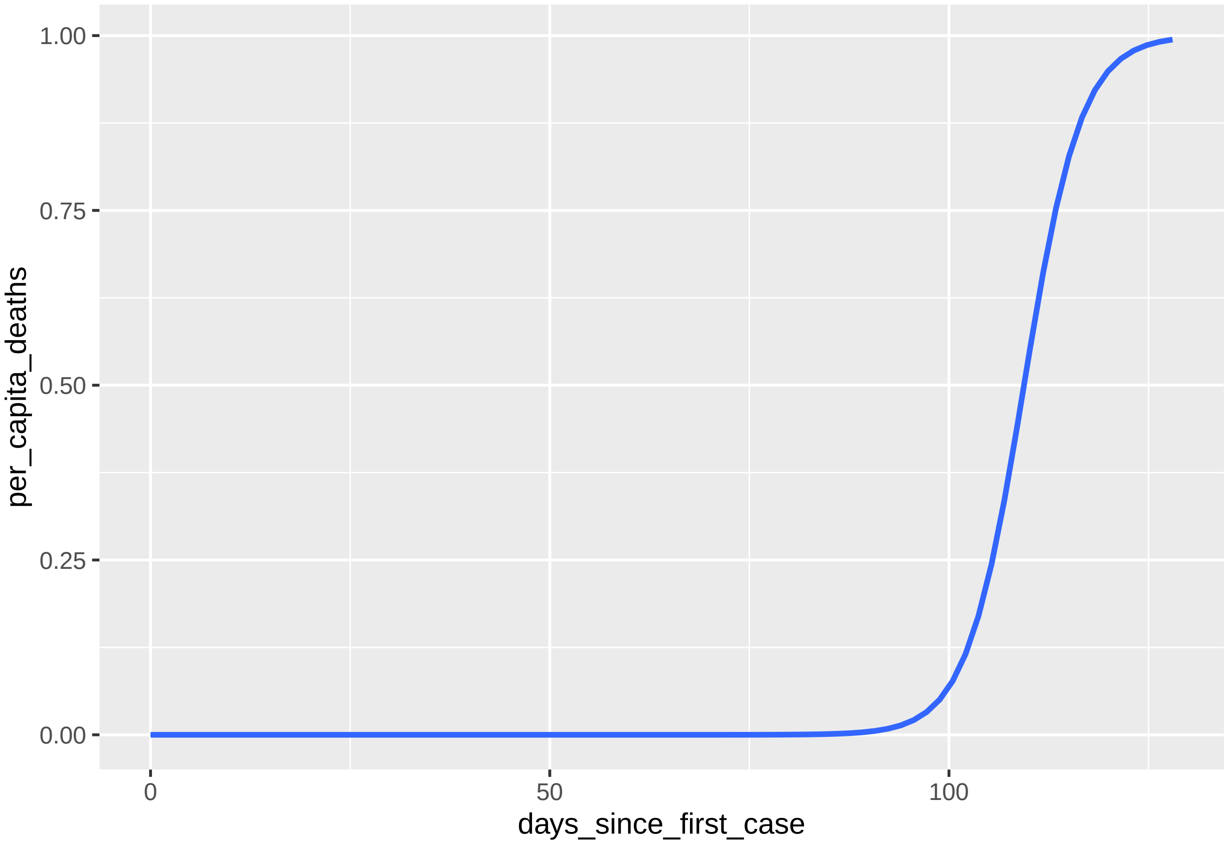
```r
ggplot(full %>% group_by(days_since_first_case) %>% summarize_at(c("cases", "deaths", "population-E"), s
  geom_path() +
  geom_smooth(method = "glm", method.args = list(family = "binomial"),
              se = FALSE, fullrange = TRUE)
```

```
## Warning: Removed 10 rows containing non-finite values (stat_smooth).

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: Removed 10 rows containing missing values (geom_path).
```



```r
ggplot(full %>% group_by(days_since_first_case) %>% summarize_at(c("cases", "deaths", "population-E"), s
  geom_path() +
  geom_smooth(method = "glm", method.args = list(family = "binomial"),
              se = FALSE, fullrange = TRUE) +
  xlim(0, 128)
```

```
## Warning: Removed 10 rows containing non-finite values (stat_smooth).

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: Removed 10 rows containing missing values (geom_path).
```

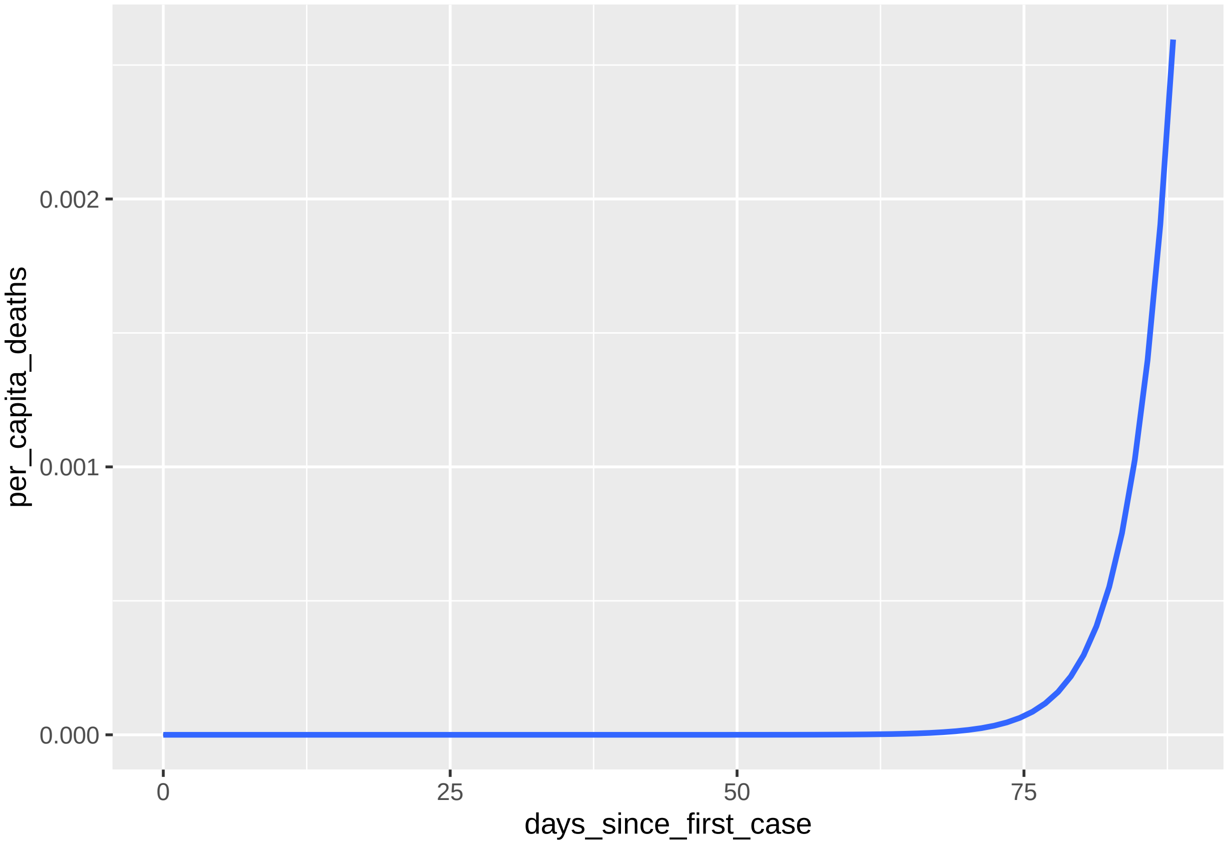


```
ggplot(full %>% group_by(days_since_first_case) %>% summarize_at(c("cases", "deaths", "population-E"), s
  geom_path() +
  geom_smooth(method = "glm", method.args = list(family = "binomial"),
              se = FALSE, fullrange = TRUE) +
  xlim(0, 88)
```

```
## Warning: Removed 10 rows containing non-finite values (stat_smooth).
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
## Warning: Removed 10 rows containing missing values (geom_path).
```


0.002
0.001
0.000
per_capita_deaths
0
25
50
75
days_since_first_case