coronavirus

Kevin Cummiskey

4/2/2020

Here is an overview of some of the COVID-19 tools out there.

- Coronovirus package

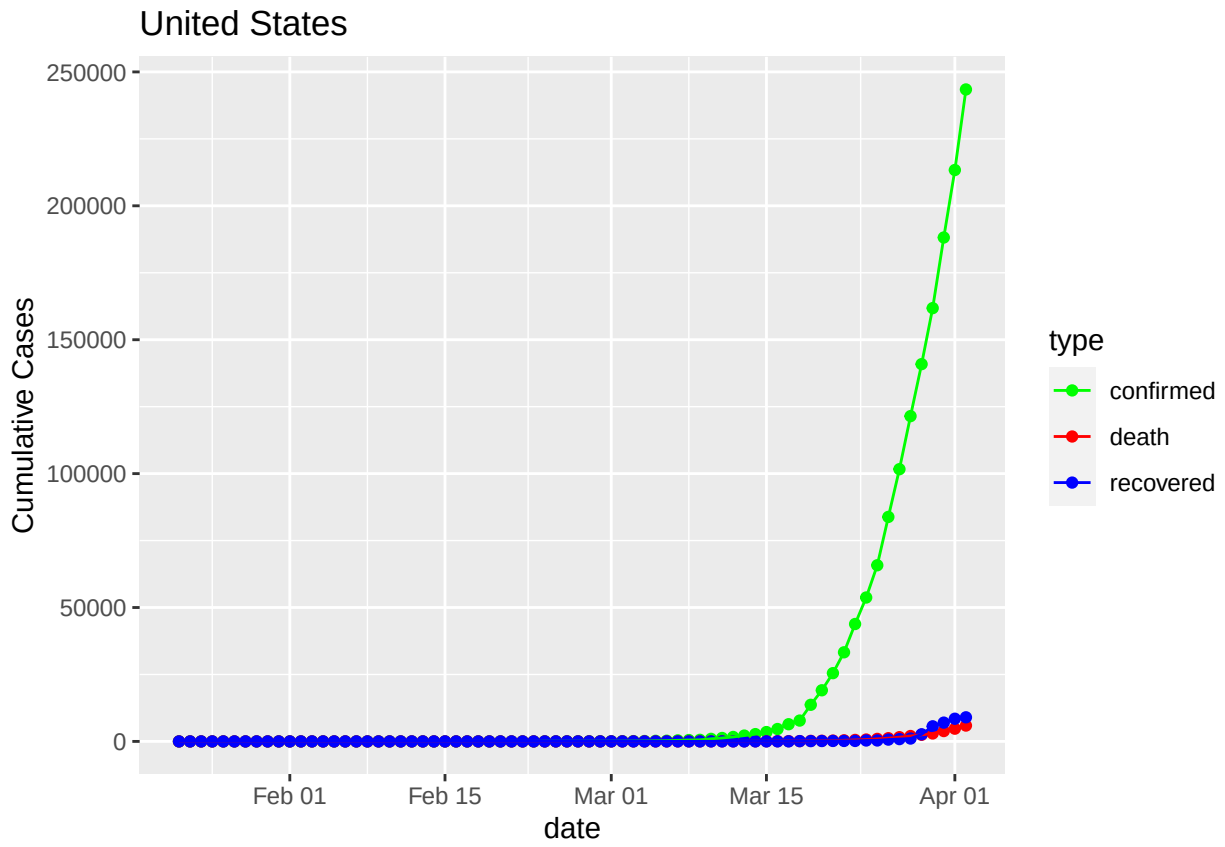
```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.0     v purrr   0.3.3
## v tibble  3.0.0     v dplyr   0.8.5
## v tidyr   1.0.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0

## Warning: package 'tibble' was built under R version 3.6.2

## -- Conflicts ------------------------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(directlabels)

coronavirus <- read_csv(file = "https://raw.githubusercontent.com/RamiKrispin/coronavirus-csv/master/co
```

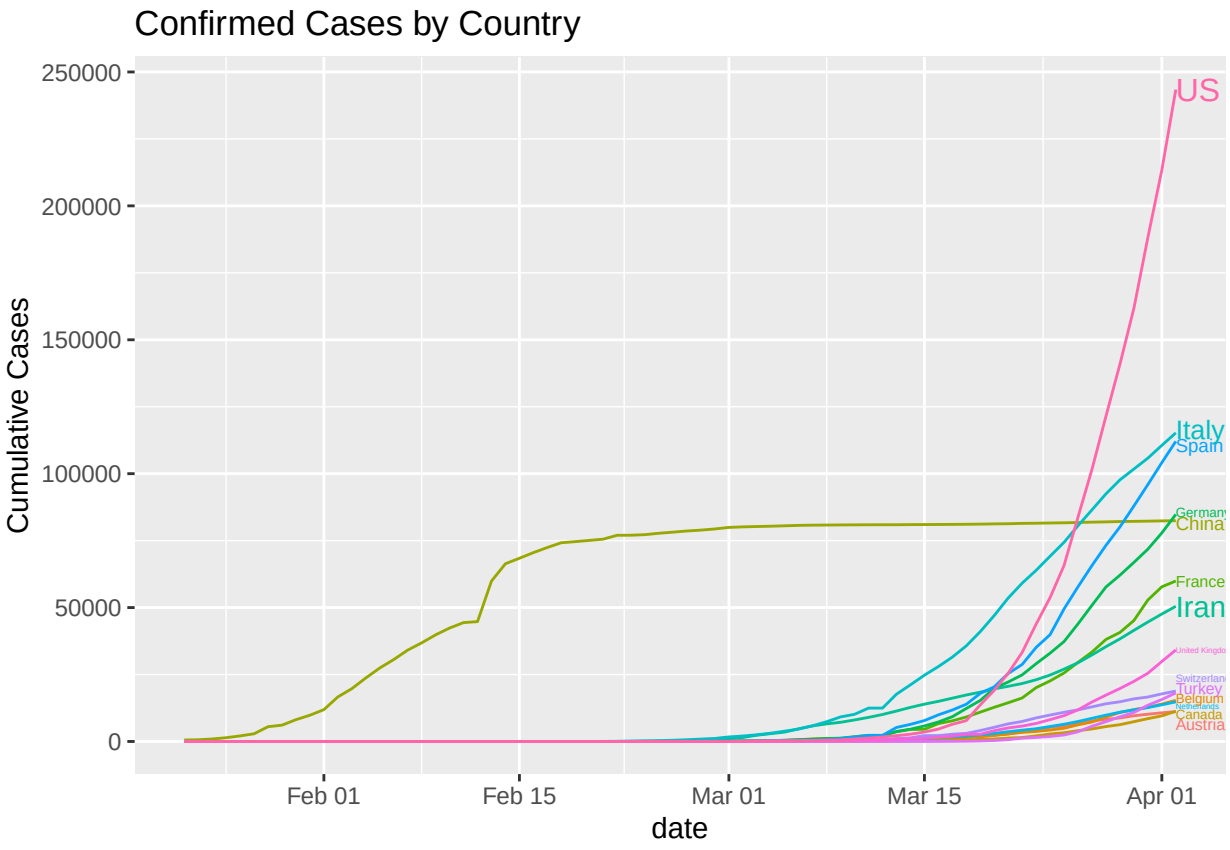
```
## Parsed with column specification:
## cols(
##   Province.State = col_logical(),
##   Country.Region = col_character(),
##   Lat = col_double(),
##   Long = col_double(),
##   date = col_date(format = ""),
##   cases = col_double(),
##   type = col_character()
## )

## Warning: 16200 parsing failures.
##   row           col           expected  actual
## 12817 Province.State 1/0/T/F/TRUE/FALSE Alberta 'https://raw.githubusercontent.com/RamiKrispin/corona
## 12818 Province.State 1/0/T/F/TRUE/FALSE Alberta 'https://raw.githubusercontent.com/RamiKrispin/corona
## 12819 Province.State 1/0/T/F/TRUE/FALSE Alberta 'https://raw.githubusercontent.com/RamiKrispin/corona
## 12820 Province.State 1/0/T/F/TRUE/FALSE Alberta 'https://raw.githubusercontent.com/RamiKrispin/corona
## 12821 Province.State 1/0/T/F/TRUE/FALSE Alberta 'https://raw.githubusercontent.com/RamiKrispin/corona
## ..... ............. ................. ....... ................................................
## See problems(...) for more details.
```

```r
coronavirus %>%
  group_by(Country.Region, type, date) %>%
  summarise(cases = sum(cases)) %>%
  group_by(Country.Region, type) %>%
  mutate(cases_cum = cumsum(cases))  -> covid_country

#Other States
covid_country %>%
  filter(Country.Region == "US") %>%
  ggplot(aes(x = date, y = cases_cum, color = type)) +
  geom_point() +
  geom_line() +
  labs(title = "United States", y = "Cumulative Cases") +
  scale_color_manual(values =  c("green", "red", "blue"))
```




```r
covid_country %>%
  filter(type == "confirmed") %>%
  group_by(Country.Region) %>%
  arrange(-cases_cum) %>%
  top_n(1, cases_cum) %>%
  filter(cases_cum > 10000) %>%
  pull(Country.Region) -> top.countries


covid_country %>%
```

```
  filter(type == "confirmed", Country.Region %in% top.countries) %>%
  ggplot(aes(x = date, y = cases_cum, group = Country.Region, color = Country.Region)) +
  geom_line() +
  labs(title = "Confirmed Cases by Country", y = "Cumulative Cases") +
  geom_dl(aes(label = Country.Region), method="last.qp") +
  theme(legend.position = "none")
```



Make a map

```
library(maps)
```

```
##
## Attaching package: 'maps'
```

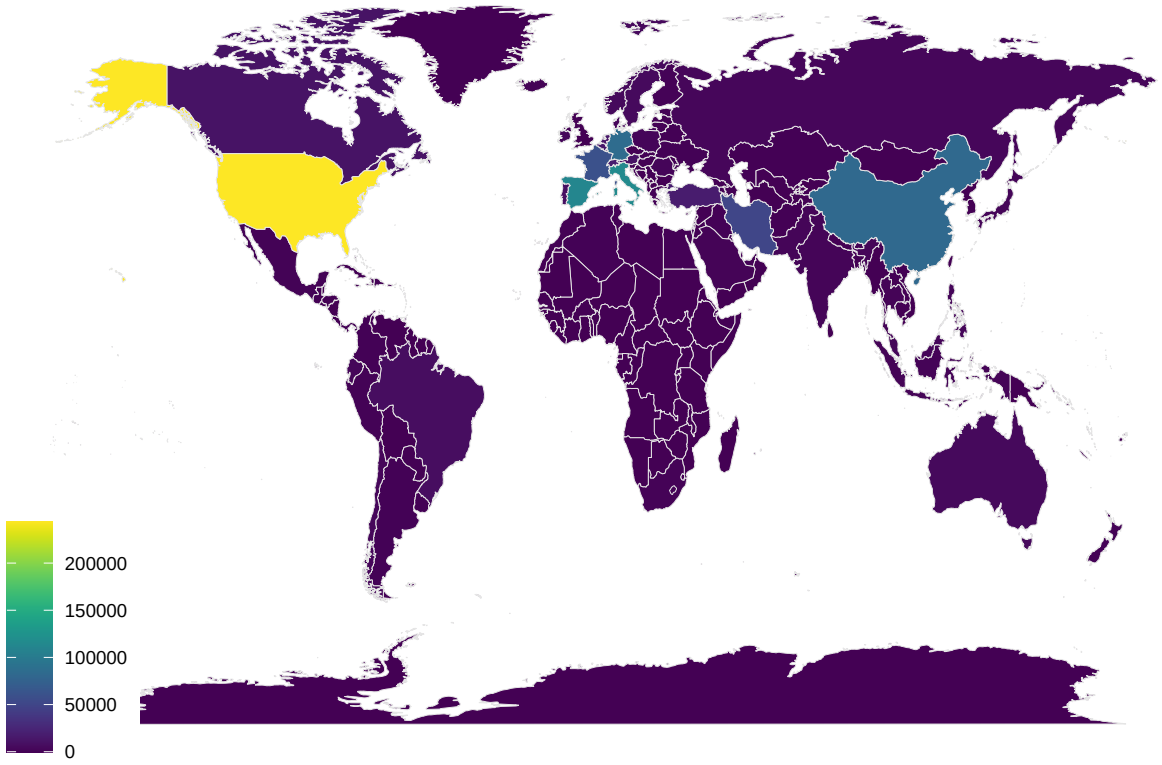
```
## The following object is masked from 'package:purrr':
##
##     map
```

```
library(mapproj)
library(ggthemes)
library(viridis)
```

```
## Loading required package: viridisLite
```

```
# Map Data
world <- map_data("world")


# Add coronavirus data
coronavirus %>%
  filter(type == "confirmed") %>%
  group_by(Country.Region, type) %>%
  summarize(cases = sum(cases, na.rm = TRUE)) %>%
  group_by() %>%
  mutate(Country.Region = replace(Country.Region, Country.Region == "US","USA")) -> country.cases



world %>%
  left_join(country.cases, by = c("region" = "Country.Region")) %>%
  replace_na(list(cases = 0)) -> world

ggplot(data = world,
            mapping = aes(x = long, y = lat,
                          group = group,
                          fill = cases)) +
  geom_polygon(color = "gray90", size = 0.1) +
  labs(title = "Election Results 2016", fill = NULL) +
  theme_map() +
  scale_fill_viridis()
```

Election Results 2016



Modeling

Modeling COVID-19 Spread vs Healthcare Capacity

https://alhill.shinyapps.io/COVID19seir/