COVID-19 Data Science Project

Learner

2023-01-31

Step 1: Importing the Data:

library(tidyverse)

```
\section(table)
── Attaching packages ─────────────────────── tidyverse 1.3.2 ──
✓ ggplot2 3.4.0      ✓ purrr   1.0.1
✓ tibble  3.1.8      ✓ dplyr   1.0.10
✓ tidyr   1.3.0      ✓ stringr 1.5.0
✓ readr   2.1.3      ✓ forcats 0.5.2
── Conflicts ────────────────────────── tidyverse_conflicts() ──
✖ dplyr::filter() masks stats::filter()
✖ dplyr::lag()    masks stats::lag()
```

library(tinytex) library(lubridate)

```
\section(table)
Attaching package: ‘lubridate’

The following objects are masked from ‘package:base’:

    date, intersect, setdiff, union
```

url_in <- “https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_time_series/”

file_names <- c(“time_series_covid19_confirmed_global.csv”, “time_series_covid19_deaths_global.csv”, “time_series_covid19_confirmed_US.csv”, “time_series_covid19_deaths_US.csv”)

urls <- str_c(url_in, file_names)

urls

```
\section(table)
[1] "https://raw.githubusercontent.com/CSSEGISandData/COVID-
19/master/csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_co
nfirmed_global.csv"
[2] "https://raw.githubusercontent.com/CSSEGISandData/COVID-
19/master/csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_de
aths_global.csv"
[3] "https://raw.githubusercontent.com/CSSEGISandData/COVID-
```

```
19/master/csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_co
nfirmed_US.csv"
[4] "https://raw.githubusercontent.com/CSSEGISandData/COVID-
19/master/csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_de
aths_US.csv"
```

global_cases = read_csv(urls[1])

```
section(table)

Rows: 289 Columns: 1109
── Column specification ────────────────────────────────────────
Delimiter: ","
chr    (2): Province/State, Country/Region
dbl (1107): Lat, Long, 1/22/20, 1/23/20, 1/24/20, 1/25/20, 1/...

ℹ Use `spec()` to retrieve the full column specification for this data.
ℹ Specify the column types or set `show_col_types = FALSE` to quiet this
message.
```

global_deaths = read_csv(urls[2])

```
\section(table)
chr    (2): Province/State, Country/Region
dbl (1107): Lat, Long, 1/22/20, 1/23/20, 1/24/20, 1/25/20, 1/...

ℹ Use `spec()` to retrieve the full column specification for this data.
ℹ Specify the column types or set `show_col_types = FALSE` to quiet this
message.
```

US_cases = read_csv(urls[3])

```
\section(table)
Rows: 3342 Columns: 1116
── Column specification ────────────────────────────────────────
Delimiter: ","
chr    (6): iso2, iso3, Admin2, Province_State, Country_Regio...
dbl (1110): UID, code3, FIPS, Lat, Long_, 1/22/20, 1/23/20, 1...

ℹ Use `spec()` to retrieve the full column specification for this data.
ℹ Specify the column types or set `show_col_types = FALSE` to quiet this
message.
```

US_deaths <- read_csv(urls[4])

```
\section(table)

chr    (6): iso2, iso3, Admin2, Province_State, Country_Regio...
dbl (1111): UID, code3, FIPS, Lat, Long_, Population, 1/22/20...

ℹ Use `spec()` to retrieve the full column specification for this data.
```

```
ℹ Specify the column types or set `show_col_types = FALSE` to quiet this
message.
```

Step 2: Tidy & Transform the Data

global_cases <- global_cases %>% pivot_longer(cols = -c('Province/State', 'Country/Region', Lat, Long), names_to = "date", values_to = "cases") %>% select(-c(Lat, Long))

global_cases

```
\section(table)
# A tibble: 319,345 × 4
   `Province/State` `Country/Region` date    cases
   <chr>            <chr>            <chr>   <dbl>
 1 NA               Afghanistan      1/22/20     0
 2 NA               Afghanistan      1/23/20     0
 3 NA               Afghanistan      1/24/20     0
 4 NA               Afghanistan      1/25/20     0
 5 NA               Afghanistan      1/26/20     0
 6 NA               Afghanistan      1/27/20     0
 7 NA               Afghanistan      1/28/20     0
 8 NA               Afghanistan      1/29/20     0
 9 NA               Afghanistan      1/30/20     0
10 NA               Afghanistan      1/31/20     0
# … with 319,335 more rows
# ℹ Use `print(n = ...)` to see more rows
```

global_deaths <- global_deaths %>% pivot_longer(cols = -c("Province/State", "Country/Region", Lat, Long), names_to = "date", values_to = "deaths") %>% select(-c(Lat, Long))

global_deaths

```
\section(table)
# A tibble: 319,345 × 4
   `Province/State` `Country/Region` date    deaths
   <chr>            <chr>            <chr>    <dbl>
 1 NA               Afghanistan      1/22/20      0
 2 NA               Afghanistan      1/23/20      0
 3 NA               Afghanistan      1/24/20      0
 4 NA               Afghanistan      1/25/20      0
 5 NA               Afghanistan      1/26/20      0
 6 NA               Afghanistan      1/27/20      0
 7 NA               Afghanistan      1/28/20      0
 8 NA               Afghanistan      1/29/20      0
 9 NA               Afghanistan      1/30/20      0
10 NA               Afghanistan      1/31/20      0
# … with 319,335 more rows
# ℹ Use `print(n = ...)` to see more rows
```

global <- global_cases %>% full_join(global_deaths) %>% rename(Country_Region = “Country/Region”, Province_State = “Province/State”) %>% mutate(date = mdy(date))

```
\section(table)
Joining, by = c("Province/State", "Country/Region", "date")
```

global

```
\section(table)
# A tibble: 319,345 × 5
   Province_State Country_Region date       cases deaths
   <chr>          <chr>          <date>     <dbl>  <dbl>
 1 NA             Afghanistan    2020-01-22     0      0
 2 NA             Afghanistan    2020-01-23     0      0
 3 NA             Afghanistan    2020-01-24     0      0
 4 NA             Afghanistan    2020-01-25     0      0
 5 NA             Afghanistan    2020-01-26     0      0
 6 NA             Afghanistan    2020-01-27     0      0
 7 NA             Afghanistan    2020-01-28     0      0
 8 NA             Afghanistan    2020-01-29     0      0
 9 NA             Afghanistan    2020-01-30     0      0
10 NA             Afghanistan    2020-01-31     0      0
# … with 319,335 more rows
# ℹ Use `print(n = ...)` to see more rows
```

summary(global)

```
\section(table)
 Province_State     Country_Region          date
 Length:319345      Length:319345      Min.   :2020-01-22
 Class :character   Class :character   1st Qu.:2020-10-24
 Mode  :character   Mode  :character   Median :2021-07-27
                                       Mean   :2021-07-27
                                       3rd Qu.:2022-04-29
                                       Max.   :2023-01-30
     cases              deaths
 Min.   :        0   Min.   :      0
 1st Qu.:      603   1st Qu.:      3
 Median :    13122   Median :    139
 Mean   :   912205   Mean   :  13024
 3rd Qu.:   214636   3rd Qu.:   2897
 Max.   :102310636   Max.   :1107855
```

global <- global %>% filter(cases > 0) summary(global)

```
\section(table)
 Province_State     Country_Region          date
 Length:295921      Length:295921      Min.   :2020-01-22
 Class :character   Class :character   1st Qu.:2020-12-02
 Mode  :character   Mode  :character   Median :2021-08-27
                                       Mean   :2021-08-23
```

```
                                    3rd Qu.:2022-05-17
                                    Max.   :2023-01-30
     cases               deaths
 Min.   :        1   Min.   :      0
 1st Qu.:     1202   1st Qu.:      7
 Median :    18797   Median :    199
 Mean   :   984412   Mean   :  14055
 3rd Qu.:   258540   3rd Qu.:   3547
 Max.   :102310636   Max.   :1107855
```

global %>% filter(cases > 28000000)

```
\section(table)
# A tibble: 2,244 × 5
   Province_State Country_Region date          cases deaths
   <chr>          <chr>          <date>        <dbl>  <dbl>
 1 NA             Brazil         2022-02-18 28072238 643340
 2 NA             Brazil         2022-02-19 28177367 644195
 3 NA             Brazil         2022-02-20 28218180 644592
 4 NA             Brazil         2022-02-21 28258458 644918
 5 NA             Brazil         2022-02-22 28361951 645735
 6 NA             Brazil         2022-02-23 28493336 646714
 7 NA             Brazil         2022-02-24 28589235 647703
 8 NA             Brazil         2022-02-25 28679671 648496
 9 NA             Brazil         2022-02-26 28749552 649184
10 NA             Brazil         2022-02-27 28776794 649437
# … with 2,234 more rows
# ℹ Use `print(n = ...)` to see more rows
```

lets wprk on US cases:

US_cases

```
\section(table)
# A tibble: 3,342 × 1,116
        UID iso2  iso3  code3  FIPS Admin2 Provi…¹ Count…²   Lat Long_
      <dbl> <chr> <chr> <dbl> <dbl> <chr>  <chr>   <chr>   <dbl> <dbl>
 1 84001001 US    USA     840  1001 Autau… Alabama US       32.5 -86.6
 2 84001003 US    USA     840  1003 Baldw… Alabama US       30.7 -87.7
 3 84001005 US    USA     840  1005 Barbo… Alabama US       31.9 -85.4
 4 84001007 US    USA     840  1007 Bibb   Alabama US       33.0 -87.1
 5 84001009 US    USA     840  1009 Blount Alabama US       34.0 -86.6
 6 84001011 US    USA     840  1011 Bullo… Alabama US       32.1 -85.7
 7 84001013 US    USA     840  1013 Butler Alabama US       31.8 -86.7
 8 84001015 US    USA     840  1015 Calho… Alabama US       33.8 -85.8
 9 84001017 US    USA     840  1017 Chamb… Alabama US       32.9 -85.4
10 84001019 US    USA     840  1019 Chero… Alabama US       34.2 -85.6
# … with 3,332 more rows, 1,106 more variables: Combined_Key <chr>,
#   `1/22/20` <dbl>, `1/23/20` <dbl>, `1/24/20` <dbl>,
#   `1/25/20` <dbl>, `1/26/20` <dbl>, `1/27/20` <dbl>,
#   `1/28/20` <dbl>, `1/29/20` <dbl>, `1/30/20` <dbl>,
```

```
#   `1/31/20` <dbl>, `2/1/20` <dbl>, `2/2/20` <dbl>, `2/3/20` <dbl>,
#   `2/4/20` <dbl>, `2/5/20` <dbl>, `2/6/20` <dbl>, `2/7/20` <dbl>,
#   `2/8/20` <dbl>, `2/9/20` <dbl>, `2/10/20` <dbl>, …
# i Use `print(n = ...)` to see more rows, and `colnames()` to see all
variable names
```

US_cases %>% pivot_longer(cols = -(UID:Combined_Key), names_to = “date”, values_to = “cases”)

```
\section(table)
# A tibble: 3,692,910 × 13
        UID iso2  iso3  code3  FIPS Admin2 Provi…¹ Count…²   Lat Long_
      <dbl> <chr> <chr> <dbl> <dbl> <chr>  <chr>   <chr>   <dbl> <dbl>
 1 84001001 US    USA     840  1001 Autau… Alabama US       32.5 -86.6
 2 84001001 US    USA     840  1001 Autau… Alabama US       32.5 -86.6
 3 84001001 US    USA     840  1001 Autau… Alabama US       32.5 -86.6
 4 84001001 US    USA     840  1001 Autau… Alabama US       32.5 -86.6
 5 84001001 US    USA     840  1001 Autau… Alabama US       32.5 -86.6
 6 84001001 US    USA     840  1001 Autau… Alabama US       32.5 -86.6
 7 84001001 US    USA     840  1001 Autau… Alabama US       32.5 -86.6
 8 84001001 US    USA     840  1001 Autau… Alabama US       32.5 -86.6
 9 84001001 US    USA     840  1001 Autau… Alabama US       32.5 -86.6
10 84001001 US    USA     840  1001 Autau… Alabama US       32.5 -86.6
# … with 3,692,900 more rows, 3 more variables: Combined_Key <chr>,
#   date <chr>, cases <dbl>, and abbreviated variable names
#   ¹Province_State, ²Country_Region
# i Use `print(n = ...)` to see more rows, and `colnames()` to see all
variable names
```

US_cases <- US_cases %>% pivot_longer(cols= -(UID:Combined_Key), names_to = “date”, values_to = “cases”) %>% select(Admin2:cases) %>% mutate(date=mdy(date)) %>% select(-c(Lat, Long_))

US_cases

```
\section(table)
# A tibble: 3,692,910 × 6
   Admin2  Province_State Country_Region Combined_Key date       cases
   <chr>   <chr>          <chr>          <chr>        <date>     <dbl>
 1 Autauga Alabama        US             Autauga, Al… 2020-01-22     0
 2 Autauga Alabama        US             Autauga, Al… 2020-01-23     0
 3 Autauga Alabama        US             Autauga, Al… 2020-01-24     0
 4 Autauga Alabama        US             Autauga, Al… 2020-01-25     0
 5 Autauga Alabama        US             Autauga, Al… 2020-01-26     0
 6 Autauga Alabama        US             Autauga, Al… 2020-01-27     0
 7 Autauga Alabama        US             Autauga, Al… 2020-01-28     0
 8 Autauga Alabama        US             Autauga, Al… 2020-01-29     0
 9 Autauga Alabama        US             Autauga, Al… 2020-01-30     0
10 Autauga Alabama        US             Autauga, Al… 2020-01-31     0
# … with 3,692,900 more rows
# i Use `print(n = ...)` to see more rows
```

Same of above to be done with US deaths:

US_deaths <- US_deaths %>% pivot_longer(cols = -(UID:Population), names_to = “date”, values_to = “deaths”) %>% select(Admin2:deaths) %>% mutate(date = mdy(date)) %>% select(-c(Lat, Long_))

US_deaths

```
\section(table)
# A tibble: 3,692,910 × 7
   Admin2  Province_State Country_…¹ Combi…² Popul…³ date       deaths
   <chr>   <chr>          <chr>      <chr>     <dbl> <date>      <dbl>
 1 Autauga Alabama        US         Autaug…   55869 2020-01-22      0
 2 Autauga Alabama        US         Autaug…   55869 2020-01-23      0
 3 Autauga Alabama        US         Autaug…   55869 2020-01-24      0
 4 Autauga Alabama        US         Autaug…   55869 2020-01-25      0
 5 Autauga Alabama        US         Autaug…   55869 2020-01-26      0
 6 Autauga Alabama        US         Autaug…   55869 2020-01-27      0
 7 Autauga Alabama        US         Autaug…   55869 2020-01-28      0
 8 Autauga Alabama        US         Autaug…   55869 2020-01-29      0
 9 Autauga Alabama        US         Autaug…   55869 2020-01-30      0
10 Autauga Alabama        US         Autaug…   55869 2020-01-31      0
# … with 3,692,900 more rows, and abbreviated variable names
#   ¹Country_Region, ²Combined_Key, ³Population
# i Use `print(n = ...)` to see more rows
```

US <- US_cases %>% full_join(US_deaths)

```
\section(table)
Joining, by = c("Admin2", "Province_State", "Country_Region",
"Combined_Key", "date")
```

US

```
\section(table)
# A tibble: 3,692,910 × 8
   Admin2  Province_…¹ Count…² Combi…³ date       cases Popul…⁴ deaths
   <chr>   <chr>       <chr>   <chr>   <date>     <dbl>   <dbl>  <dbl>
 1 Autauga Alabama     US      Autaug… 2020-01-22     0   55869      0
 2 Autauga Alabama     US      Autaug… 2020-01-23     0   55869      0
 3 Autauga Alabama     US      Autaug… 2020-01-24     0   55869      0
 4 Autauga Alabama     US      Autaug… 2020-01-25     0   55869      0
 5 Autauga Alabama     US      Autaug… 2020-01-26     0   55869      0
 6 Autauga Alabama     US      Autaug… 2020-01-27     0   55869      0
 7 Autauga Alabama     US      Autaug… 2020-01-28     0   55869      0
 8 Autauga Alabama     US      Autaug… 2020-01-29     0   55869      0
 9 Autauga Alabama     US      Autaug… 2020-01-30     0   55869      0
10 Autauga Alabama     US      Autaug… 2020-01-31     0   55869      0
# … with 3,692,900 more rows, and abbreviated variable names
#   ¹Province_State, ²Country_Region, ³Combined_Key, ⁴Population
# i Use `print(n = ...)` to see more rows
```

global <- global %>% unite("Combined_Key", c(Province_State, Country_Region), sep = ",", na.rm=TRUE, remove=FALSE)

global

```
\section(table)
# A tibble: 295,921 × 6
   Combined_Key Province_State Country_Region date       cases deaths
   <chr>        <chr>          <chr>          <date>     <dbl>  <dbl>
 1 Afghanistan  NA             Afghanistan    2020-02-24     5      0
 2 Afghanistan  NA             Afghanistan    2020-02-25     5      0
 3 Afghanistan  NA             Afghanistan    2020-02-26     5      0
 4 Afghanistan  NA             Afghanistan    2020-02-27     5      0
 5 Afghanistan  NA             Afghanistan    2020-02-28     5      0
 6 Afghanistan  NA             Afghanistan    2020-02-29     5      0
 7 Afghanistan  NA             Afghanistan    2020-03-01     5      0
 8 Afghanistan  NA             Afghanistan    2020-03-02     5      0
 9 Afghanistan  NA             Afghanistan    2020-03-03     5      0
10 Afghanistan  NA             Afghanistan    2020-03-04     5      0
# … with 295,911 more rows
# ℹ Use `print(n = ...)` to see more rows
```

Step 3: Visualizaing

US_by_State <- US %>% group_by(Province_State, Country_Region, date) %>% summarize(cases = sum(cases), deaths = sum(deaths), Population = sum(Population)) %>% mutate(deaths_per_mill = deaths*1000000 / Population) %>% ungroup()

```
\section(table)
`summarise()` has grouped output by
'Province_State', 'Country_Region'.
You can override using the `.groups`
argument.
```

US_by_State

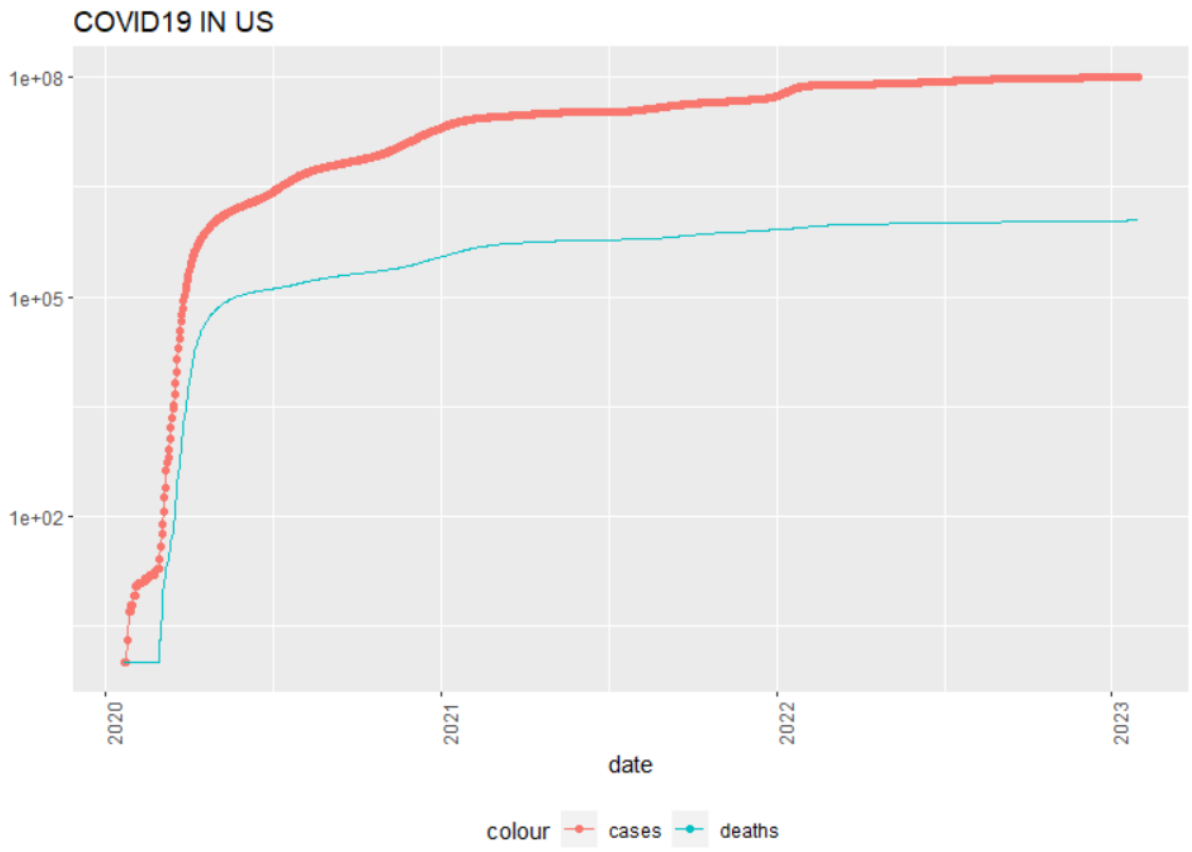
```
\section(table)
# A tibble: 64,090 × 7
   Provinc…¹ Count…² date       cases
   <chr>     <chr>   <date>     <dbl>
 1 Alabama   US      2020-01-22     0
 2 Alabama   US      2020-01-23     0
 3 Alabama   US      2020-01-24     0
 4 Alabama   US      2020-01-25     0
 5 Alabama   US      2020-01-26     0
 6 Alabama   US      2020-01-27     0
 7 Alabama   US      2020-01-28     0
 8 Alabama   US      2020-01-29     0
 9 Alabama   US      2020-01-30     0
10 Alabama   US      2020-01-31     0
# … with 64,080 more rows, 3 more
```

```
#   variables: deaths <dbl>,
#   Population <dbl>,
#   deaths_per_mill <dbl>, and
#   abbreviated variable names
#   ¹Province_State, ²Country_Region
# ℹ Use `print(n = ...)` to see more rows, and `colnames()` to see all
variable name
```

US_totals <- US_by_State %>% group_by(Country_Region, date) %>% summarize(cases = sum(cases), deaths = sum(deaths), Population = sum(Population)) %>% select(Country_Region, date, cases, deaths, Population) %>% ungroup()

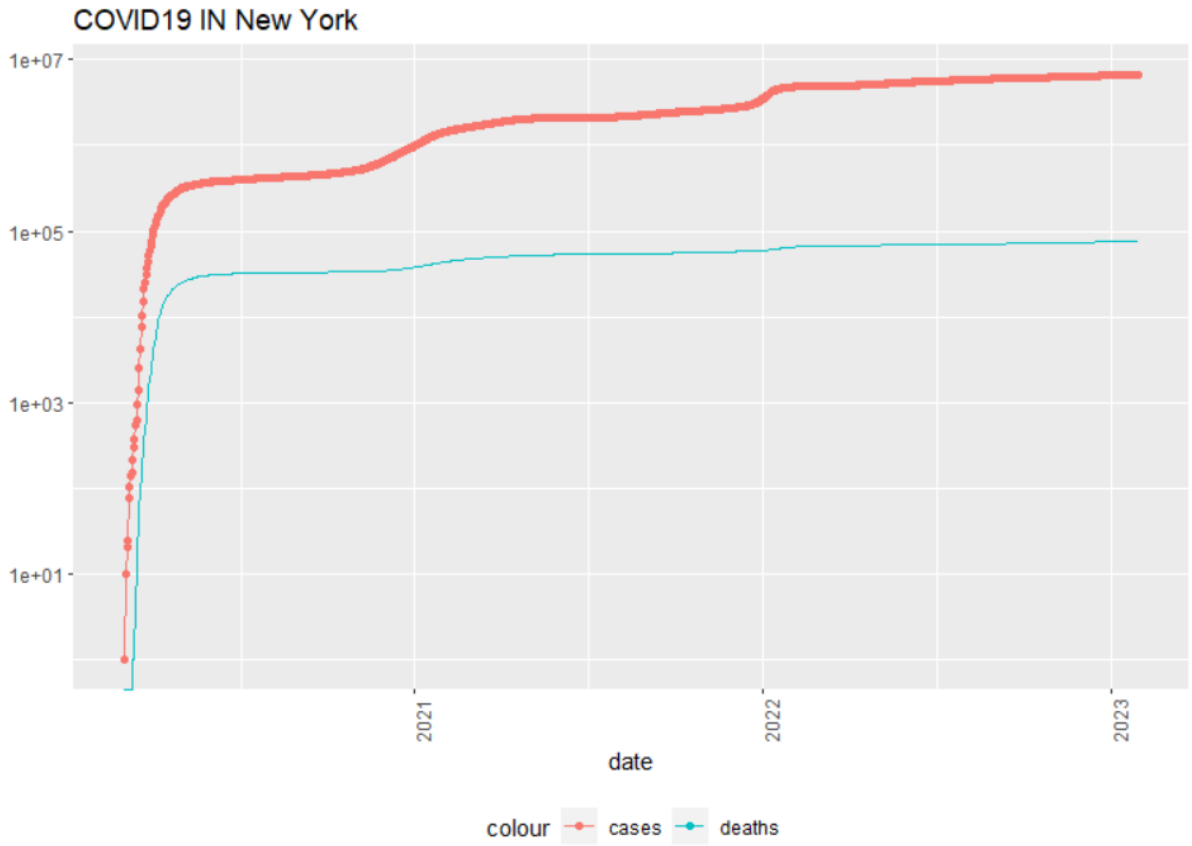
```
\section(table)

`summarise()` has grouped output by
'Country_Region'. You can override using the
`.groups` argument.
```

US_totals %>% filter(cases > 0) %>% ggplot(aes(x = date, y = cases)) + geom_line(aes(color = "cases")) + geom_point(aes(color = "cases")) + geom_line(aes(y = deaths, color = "deaths")) + scale_y_log10() + theme(legend.position="bottom", axis.text.x = element_text(angle = 90)) + labs(title="COVID19 IN US", y=NULL)



state <- “New York”

US_by_State %>% filter(Province_State ==state) %>% filter(cases > 0) %>% ggplot(aes(x = date, y = cases)) + geom_line(aes(color = “cases”)) + geom_point(aes(color = “cases”)) + geom_line(aes(y = deaths, color = “deaths”)) + scale_y_log10() + theme(legend.position=“bottom”, axis.text.x = element_text(angle = 90)) + labs(title=“COVID19 IN New York”, y=NULL)



max(US_totals$date)

```
\section(table)
[1] "2023-01-30"
```

max(US_totals$deaths)

```
\section(table)
[1] 1107855
```