Lets Talk About COVID

```
tinytex::tlmgr_install("pdfcrop")
```

```
## tlmgr update --all --self
```

```
## tlmgr install pdfcrop
```

Will need to run ‘’tinytex::tlmgr_install(“pdfcrop”)” for the knitting of this document if it isn’t already installed

Introduction

Four best friends hanging out in a living room. Three boxes of pizza, freshly delivered from Papa Johns. Two heated debates on who gets to play the infamous Mario. And 1 Nintendo Switch loaded with one of our favorite games: Mario Party. We hit our die and finish our first minigame (I won of course). All of the sudden, our cellphones begin to ring. Texts, news articles, and app feeds drown our phones with something we never heard before: COVID. Once we realized it wasn’t insomnia alerting us about our cookies, we put down our phones and continued playing until one of us (sadly not me) became victorious. At that time, none of us knew it; that would be our last time together for months.
COVID interrupted most of our lives for the past year and a half. Schools, jobs, family events, and even concerts shifted to an online presence. Governments worked frantically to develop the best policies. Scientists worked day and night to find a cure. And for some reason, toilet paper became a luxury item in the United States. Regardless, COVID has changed lives on the global scale. Throughout this data, we will review how COVID is trending and where we should focus so we can go back to a ‘normal’ life.

Packages

First, we will need the **tidverse** and **lubridate** packages to carry out our analysis.

```
library(tidyverse)
```

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

## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.1.0     v dplyr   1.0.5
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1

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

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

Data

We will be using four csv files from the The New York Times Company* Github. The New York Times Company is an American mass media company that produces a daily newspaper (*The New York Times*), located in New York City. This newspaper circulates both domestically and internationally. The goal of the company is to deliver as much unbiased news information as possible. Using their data, we will load the following datasets:

- time_series_covid19_confirmed_global.csv
  - Total amount of Covid Cases globally
- time_series_covid19_deaths_global.csv
  - Total amount of COVID-related deaths
- time_series_covid19_confirmed_US.csv
  - Total amount of COVID cases in the U.S.
- time_series_covid19_deaths_US.csv
  - Total amount of COVID-related deaths in the U.S.

These datasets contains the number of daily cases from January 22, 2020 until current date (In this case August 10, 2021).

```
url_in <- paste0("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/",
"master/csse_covid_19_data/csse_covid_19_time_series/")

filenames <- 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, filenames)
```

```
global_cases <- read_csv(urls[1])
global_deaths <- read_csv(urls[2])
us_cases <- read_csv(urls[3])
us_deaths <- read_csv(urls[4])
```

Data Cleaning

Lets take a quick look at the global_cases and global_deaths datasets.

```
global_cases
```

```
## # A tibble: 279 x 573
##    ‘Province/State‘  ‘Country/Region‘   Lat   Long ‘1/22/20‘ ‘1/23/20‘ ‘1/24/20‘
##    <chr>             <chr>            <dbl>  <dbl>     <dbl>     <dbl>     <dbl>
##  1 <NA>              Afghanistan       33.9  67.7          0         0         0
##  2 <NA>              Albania           41.2  20.2          0         0         0
##  3 <NA>              Algeria           28.0   1.66         0         0         0
##  4 <NA>              Andorra           42.5   1.52         0         0         0
##  5 <NA>              Angola           -11.2  17.9          0         0         0
##  6 <NA>              Antigua and Bar~  17.1 -61.8          0         0         0
##  7 <NA>              Argentina        -38.4 -63.6          0         0         0
##  8 <NA>              Armenia           40.1  45.0          0         0         0
##  9 Australian Capit~ Australia        -35.5 149.           0         0         0
## 10 New South Wales   Australia        -33.9 151.           0         0         0
## # ... with 269 more rows, and 566 more variables: 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>,
## #   2/11/20 <dbl>, 2/12/20 <dbl>, 2/13/20 <dbl>, 2/14/20 <dbl>, 2/15/20 <dbl>,
## #   2/16/20 <dbl>, 2/17/20 <dbl>, 2/18/20 <dbl>, 2/19/20 <dbl>, 2/20/20 <dbl>,
## #   2/21/20 <dbl>, 2/22/20 <dbl>, 2/23/20 <dbl>, 2/24/20 <dbl>, 2/25/20 <dbl>,
## #   2/26/20 <dbl>, 2/27/20 <dbl>, 2/28/20 <dbl>, 2/29/20 <dbl>, 3/1/20 <dbl>,
## #   3/2/20 <dbl>, 3/3/20 <dbl>, 3/4/20 <dbl>, 3/5/20 <dbl>, 3/6/20 <dbl>,
## #   3/7/20 <dbl>, 3/8/20 <dbl>, 3/9/20 <dbl>, 3/10/20 <dbl>, 3/11/20 <dbl>,
## #   3/12/20 <dbl>, 3/13/20 <dbl>, 3/14/20 <dbl>, 3/15/20 <dbl>, 3/16/20 <dbl>,
## #   3/17/20 <dbl>, 3/18/20 <dbl>, 3/19/20 <dbl>, 3/20/20 <dbl>, 3/21/20 <dbl>,
## #   3/22/20 <dbl>, 3/23/20 <dbl>, 3/24/20 <dbl>, 3/25/20 <dbl>, 3/26/20 <dbl>,
## #   3/27/20 <dbl>, 3/28/20 <dbl>, 3/29/20 <dbl>, 3/30/20 <dbl>, 3/31/20 <dbl>,
## #   4/1/20 <dbl>, 4/2/20 <dbl>, 4/3/20 <dbl>, 4/4/20 <dbl>, 4/5/20 <dbl>,
## #   4/6/20 <dbl>, 4/7/20 <dbl>, 4/8/20 <dbl>, 4/9/20 <dbl>, 4/10/20 <dbl>,
## #   4/11/20 <dbl>, 4/12/20 <dbl>, 4/13/20 <dbl>, 4/14/20 <dbl>, 4/15/20 <dbl>,
## #   4/16/20 <dbl>, 4/17/20 <dbl>, 4/18/20 <dbl>, 4/19/20 <dbl>, 4/20/20 <dbl>,
## #   4/21/20 <dbl>, 4/22/20 <dbl>, 4/23/20 <dbl>, 4/24/20 <dbl>, 4/25/20 <dbl>,
## #   4/26/20 <dbl>, 4/27/20 <dbl>, 4/28/20 <dbl>, 4/29/20 <dbl>, 4/30/20 <dbl>,
## #   5/1/20 <dbl>, 5/2/20 <dbl>, 5/3/20 <dbl>, ...
```

```
global_deaths
```

```
## # A tibble: 279 x 573
##    ‘Province/State‘  ‘Country/Region‘   Lat   Long ‘1/22/20‘ ‘1/23/20‘ ‘1/24/20‘
##    <chr>             <chr>            <dbl>  <dbl>     <dbl>     <dbl>     <dbl>
##  1 <NA>              Afghanistan       33.9  67.7          0         0         0
##  2 <NA>              Albania           41.2  20.2          0         0         0
##  3 <NA>              Algeria           28.0   1.66         0         0         0
##  4 <NA>              Andorra           42.5   1.52         0         0         0
##  5 <NA>              Angola           -11.2  17.9          0         0         0
##  6 <NA>              Antigua and Bar~  17.1 -61.8          0         0         0
##  7 <NA>              Argentina        -38.4 -63.6          0         0         0
##  8 <NA>              Armenia           40.1  45.0          0         0         0
##  9 Australian Capit~ Australia        -35.5 149.           0         0         0
## 10 New South Wales   Australia        -33.9 151.           0         0         0
## # ... with 269 more rows, and 566 more variables: 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>,
## #   2/11/20 <dbl>, 2/12/20 <dbl>, 2/13/20 <dbl>, 2/14/20 <dbl>, 2/15/20 <dbl>,
## #   2/16/20 <dbl>, 2/17/20 <dbl>, 2/18/20 <dbl>, 2/19/20 <dbl>, 2/20/20 <dbl>,
## #   2/21/20 <dbl>, 2/22/20 <dbl>, 2/23/20 <dbl>, 2/24/20 <dbl>, 2/25/20 <dbl>,
## #   2/26/20 <dbl>, 2/27/20 <dbl>, 2/28/20 <dbl>, 2/29/20 <dbl>, 3/1/20 <dbl>,
## #   3/2/20 <dbl>, 3/3/20 <dbl>, 3/4/20 <dbl>, 3/5/20 <dbl>, 3/6/20 <dbl>,
## #   3/7/20 <dbl>, 3/8/20 <dbl>, 3/9/20 <dbl>, 3/10/20 <dbl>, 3/11/20 <dbl>,
## #   3/12/20 <dbl>, 3/13/20 <dbl>, 3/14/20 <dbl>, 3/15/20 <dbl>, 3/16/20 <dbl>,
## #   3/17/20 <dbl>, 3/18/20 <dbl>, 3/19/20 <dbl>, 3/20/20 <dbl>, 3/21/20 <dbl>,
## #   3/22/20 <dbl>, 3/23/20 <dbl>, 3/24/20 <dbl>, 3/25/20 <dbl>, 3/26/20 <dbl>,
## #   3/27/20 <dbl>, 3/28/20 <dbl>, 3/29/20 <dbl>, 3/30/20 <dbl>, 3/31/20 <dbl>,
## #   4/1/20 <dbl>, 4/2/20 <dbl>, 4/3/20 <dbl>, 4/4/20 <dbl>, 4/5/20 <dbl>,
## #   4/6/20 <dbl>, 4/7/20 <dbl>, 4/8/20 <dbl>, 4/9/20 <dbl>, 4/10/20 <dbl>,
## #   4/11/20 <dbl>, 4/12/20 <dbl>, 4/13/20 <dbl>, 4/14/20 <dbl>, 4/15/20 <dbl>,
## #   4/16/20 <dbl>, 4/17/20 <dbl>, 4/18/20 <dbl>, 4/19/20 <dbl>, 4/20/20 <dbl>,
## #   4/21/20 <dbl>, 4/22/20 <dbl>, 4/23/20 <dbl>, 4/24/20 <dbl>, 4/25/20 <dbl>,
## #   4/26/20 <dbl>, 4/27/20 <dbl>, 4/28/20 <dbl>, 4/29/20 <dbl>, 4/30/20 <dbl>,
## #   5/1/20 <dbl>, 5/2/20 <dbl>, 5/3/20 <dbl>, ...
```

Looking at the datasets, we need to do some cleaning. First, we will put each date in one column called 'date'. This way, each date per country will represent an observation. Also, we do not need the Latitude and Longitude for our analysis, so we will drop them.

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

Next, we need to combine global_deaths and global_cases so the cases and deaths are in one dataset. Also, we see our date is currently a character data type. We will need to use the *lubridate* package to convert the date column to a date object.

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

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

```
summary(global)
```

```
##  Province_State     Country_Region          date                cases         
##  Length:158751      Length:158751      Min.   :2020-01-22   Min.   :       0  
##  Class :character   Class :character   1st Qu.:2020-06-12   1st Qu.:     121  
##  Mode  :character   Mode  :character   Median :2020-11-01   Median :    1924  
##                                        Mean   :2020-11-01   Mean   :  253230  
##                                        3rd Qu.:2021-03-23   3rd Qu.:   42662  
##                                        Max.   :2021-08-12   Max.   :36306724  
##      deaths      
##  Min.   :     0  
##  1st Qu.:     1  
##  Median :    29  
##  Mean   :  5962  
##  3rd Qu.:   736  
##  Max.   :619093  
```

Looking at the data, we can see the cases are heavily, positively skewed; ths could mean there are a lot of rows with 0. So, we want to filter those out. After the filter, our minimum case is 1. Now lets view the see if the maximum number of cases listed is an outlier.

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

```
##  Province_State     Country_Region          date                cases         
##  Length:142753      Length:142753      Min.   :2020-01-22   Min.   :       1  
##  Class :character   Class :character   1st Qu.:2020-07-13   1st Qu.:     323  
##  Mode  :character   Mode  :character   Median :2020-11-25   Median :    3636  
##                                        Mean   :2020-11-22   Mean   :  281609  
##                                        3rd Qu.:2021-04-05   3rd Qu.:   61204  
##                                        Max.   :2021-08-12   Max.   :36306724  
##      deaths      
##  Min.   :     0  
##  1st Qu.:     3  
##  Median :    58  
##  Mean   :  6630  
##  3rd Qu.:  1027  
##  Max.   :619093
```

As we can see, there are multiple cases close to 36055002. So, we don't have to worry about this data point being an outlier.

```
global %>% filter(cases > 35000000 )
```

```
## # A tibble: 12 x 5
##    Province_State Country_Region date          cases deaths
##    <chr>          <chr>          <date>        <dbl>  <dbl>
##  1 <NA>           US             2021-08-01 35003417 613292
##  2 <NA>           US             2021-08-02 35131393 613743
##  3 <NA>           US             2021-08-03 35237950 614321
##  4 <NA>           US             2021-08-04 35330664 614811
##  5 <NA>           US             2021-08-05 35440488 615346
##  6 <NA>           US             2021-08-06 35695469 616493
##  7 <NA>           US             2021-08-07 35739551 616718
##  8 <NA>           US             2021-08-08 35763785 616829
##  9 <NA>           US             2021-08-09 35948131 617321
## 10 <NA>           US             2021-08-10 36055002 618137
## 11 <NA>           US             2021-08-11 36190179 618479
## 12 <NA>           US             2021-08-12 36306724 619093
```

Now we want to take a look at the US cases. When looking at the dataset we can see some weird codes and data types. We need to pivot the dates, while keeping Admin2, Province/State, Country/Region and Lat/Long. We also need to convert *date* to a date object. We can do the same for us_deaths as it follows a similar format as us_cases.

```
us_cases
```

```
## # A tibble: 3,342 x 580
##         UID iso2  iso3  code3  FIPS Admin2   Province_State Country_Region   Lat
##       <dbl> <chr> <chr> <dbl> <dbl> <chr>    <chr>          <chr>          <dbl>
##  1 84001001 US    USA     840  1001 Autauga  Alabama        US              32.5
##  2 84001003 US    USA     840  1003 Baldwin  Alabama        US              30.7
##  3 84001005 US    USA     840  1005 Barbour  Alabama        US              31.9
##  4 84001007 US    USA     840  1007 Bibb     Alabama        US              33.0
##  5 84001009 US    USA     840  1009 Blount   Alabama        US              34.0
##  6 84001011 US    USA     840  1011 Bullock  Alabama        US              32.1
##  7 84001013 US    USA     840  1013 Butler   Alabama        US              31.8
##  8 84001015 US    USA     840  1015 Calhoun  Alabama        US              33.8
##  9 84001017 US    USA     840  1017 Chambers Alabama        US              32.9
## 10 84001019 US    USA     840  1019 Cherokee Alabama        US              34.2
## # ... with 3,332 more rows, and 571 more variables: Long_ <dbl>,
## #   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>, 2/11/20 <dbl>, 2/12/20 <dbl>, 2/13/20 <dbl>,
## #   2/14/20 <dbl>, 2/15/20 <dbl>, 2/16/20 <dbl>, 2/17/20 <dbl>, 2/18/20 <dbl>,
## #   2/19/20 <dbl>, 2/20/20 <dbl>, 2/21/20 <dbl>, 2/22/20 <dbl>, 2/23/20 <dbl>,
## #   2/24/20 <dbl>, 2/25/20 <dbl>, 2/26/20 <dbl>, 2/27/20 <dbl>, 2/28/20 <dbl>,
## #   2/29/20 <dbl>, 3/1/20 <dbl>, 3/2/20 <dbl>, 3/3/20 <dbl>, 3/4/20 <dbl>,
## #   3/5/20 <dbl>, 3/6/20 <dbl>, 3/7/20 <dbl>, 3/8/20 <dbl>, 3/9/20 <dbl>,
## #   3/10/20 <dbl>, 3/11/20 <dbl>, 3/12/20 <dbl>, 3/13/20 <dbl>, 3/14/20 <dbl>,
## #   3/15/20 <dbl>, 3/16/20 <dbl>, 3/17/20 <dbl>, 3/18/20 <dbl>, 3/19/20 <dbl>,
## #   3/20/20 <dbl>, 3/21/20 <dbl>, 3/22/20 <dbl>, 3/23/20 <dbl>, 3/24/20 <dbl>,
## #   3/25/20 <dbl>, 3/26/20 <dbl>, 3/27/20 <dbl>, 3/28/20 <dbl>, 3/29/20 <dbl>,
## #   3/30/20 <dbl>, 3/31/20 <dbl>, 4/1/20 <dbl>, 4/2/20 <dbl>, 4/3/20 <dbl>,
## #   4/4/20 <dbl>, 4/5/20 <dbl>, 4/6/20 <dbl>, 4/7/20 <dbl>, 4/8/20 <dbl>,
## #   4/9/20 <dbl>, 4/10/20 <dbl>, 4/11/20 <dbl>, 4/12/20 <dbl>, 4/13/20 <dbl>,
## #   4/14/20 <dbl>, 4/15/20 <dbl>, 4/16/20 <dbl>, 4/17/20 <dbl>, 4/18/20 <dbl>,
## #   4/19/20 <dbl>, 4/20/20 <dbl>, 4/21/20 <dbl>, 4/22/20 <dbl>, 4/23/20 <dbl>,
## #   4/24/20 <dbl>, 4/25/20 <dbl>, 4/26/20 <dbl>, 4/27/20 <dbl>, 4/28/20 <dbl>,
## #   ...
```

```r
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_))
```

```r
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_))
```

We will join the two US datasets.

```r
US <- us_cases %>%
full_join(us_deaths)
```

```
## Joining, by = c("Admin2", "Province_State", "Country_Region", "Combined_Key", "date")
```

```
US
```

```
## # A tibble: 1,901,598 x 8
##    Admin2 Province_State Country_Region Combined_Key date       cases Population
##    <chr>  <chr>          <chr>          <chr>        <date>     <dbl>      <dbl>
##  1 Autau~ Alabama        US             Autauga, Al~ 2020-01-22     0      55869
##  2 Autau~ Alabama        US             Autauga, Al~ 2020-01-23     0      55869
##  3 Autau~ Alabama        US             Autauga, Al~ 2020-01-24     0      55869
##  4 Autau~ Alabama        US             Autauga, Al~ 2020-01-25     0      55869
##  5 Autau~ Alabama        US             Autauga, Al~ 2020-01-26     0      55869
##  6 Autau~ Alabama        US             Autauga, Al~ 2020-01-27     0      55869
##  7 Autau~ Alabama        US             Autauga, Al~ 2020-01-28     0      55869
##  8 Autau~ Alabama        US             Autauga, Al~ 2020-01-29     0      55869
##  9 Autau~ Alabama        US             Autauga, Al~ 2020-01-30     0      55869
## 10 Autau~ Alabama        US             Autauga, Al~ 2020-01-31     0      55869
## # ... with 1,901,588 more rows, and 1 more variable: deaths <dbl>
```

We need to combine the state and country_region variables of the global_dataset to create a key. This will allow some comparative analysis of the different countries. Also, we need to add the population of these countries to the final dataset.

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

We will use this csv to get the population for the different countries.

```
uid_lookup_url <- paste0("https://raw.githubusercontent.com/CSSEGISandData/",
"COVID-19/master/csse_covid_19_data/UID_ISO_FIPS_LookUp_Table.csv")

uid <- read_csv(uid_lookup_url) %>%
  select(-c(Lat, Long_, Combined_Key, code3, iso2, iso3, Admin2))
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   UID = col_double(),
##   iso2 = col_character(),
##   iso3 = col_character(),
##   code3 = col_double(),
##   FIPS = col_character(),
##   Admin2 = col_character(),
##   Province_State = col_character(),
##   Country_Region = col_character(),
##   Lat = col_double(),
##   Long_ = col_double(),
##   Combined_Key = col_character(),
##   Population = col_double()
## )
```

Here we'll add the uid csv to global_dataset to add the population as a column.

```
global <- global %>%
  left_join(uid, by = c("Province_State", "Country_Region")) %>%
  select(-c(UID, FIPS)) %>%
  select(Province_State, Country_Region, date, cases, deaths, Population,
         Combined_Key)
global
```

```
## # A tibble: 142,753 x 7
##    Province_State Country_Region date       cases deaths Population Combined_Key
##    <chr>          <chr>          <date>     <dbl>  <dbl>      <dbl> <chr>
##  1 <NA>           Afghanistan    2020-02-24     1      0   38928341 Afghanistan
##  2 <NA>           Afghanistan    2020-02-25     1      0   38928341 Afghanistan
##  3 <NA>           Afghanistan    2020-02-26     1      0   38928341 Afghanistan
##  4 <NA>           Afghanistan    2020-02-27     1      0   38928341 Afghanistan
##  5 <NA>           Afghanistan    2020-02-28     1      0   38928341 Afghanistan
##  6 <NA>           Afghanistan    2020-02-29     1      0   38928341 Afghanistan
##  7 <NA>           Afghanistan    2020-03-01     1      0   38928341 Afghanistan
##  8 <NA>           Afghanistan    2020-03-02     1      0   38928341 Afghanistan
##  9 <NA>           Afghanistan    2020-03-03     2      0   38928341 Afghanistan
## 10 <NA>           Afghanistan    2020-03-04     4      0   38928341 Afghanistan
## # ... with 142,743 more rows
```

Visualization and Analysis

First, we'll create a data set that will have the number of cases and deaths by state. Also, we will create the **deaths_per_mil** variable to use for comparative analysis.

```
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) %>%
    select(Province_State, Country_Region, date, cases, deaths, deaths_per_mill,
    Population) %>%
    ungroup()
```

```
## ‘summarise()‘ has grouped output by ’Province_State’, ’Country_Region’. You can override using the ‘
```

```
us_by_state
```

```
## # A tibble: 33,002 x 7
##    Province_State Country_Region date       cases deaths deaths_per_mill
##    <chr>          <chr>          <date>     <dbl>  <dbl>           <dbl>
##  1 Alabama        US             2020-01-22     0      0               0
##  2 Alabama        US             2020-01-23     0      0               0
##  3 Alabama        US             2020-01-24     0      0               0
##  4 Alabama        US             2020-01-25     0      0               0
##  5 Alabama        US             2020-01-26     0      0               0
##  6 Alabama        US             2020-01-27     0      0               0
##  7 Alabama        US             2020-01-28     0      0               0
##  8 Alabama        US             2020-01-29     0      0               0
##  9 Alabama        US             2020-01-30     0      0               0
## 10 Alabama        US             2020-01-31     0      0               0
## # ... with 32,992 more rows, and 1 more variable: Population <dbl>
```

Next we can view the total amount of cases and deaths for the US. We can see at the beginning of 2020 the first cases of COVID came to the US.

```
us_totals <- us_by_state %>%
  group_by(Country_Region, date) %>%
  summarize(cases = sum(cases), deaths = sum(deaths),
            Population = sum(Population)) %>%
  mutate(deaths_per_mill = deaths * 1000000 / Population) %>%
  select(Country_Region, date, cases, deaths, deaths_per_mill, Population) %>%
  ungroup()
```

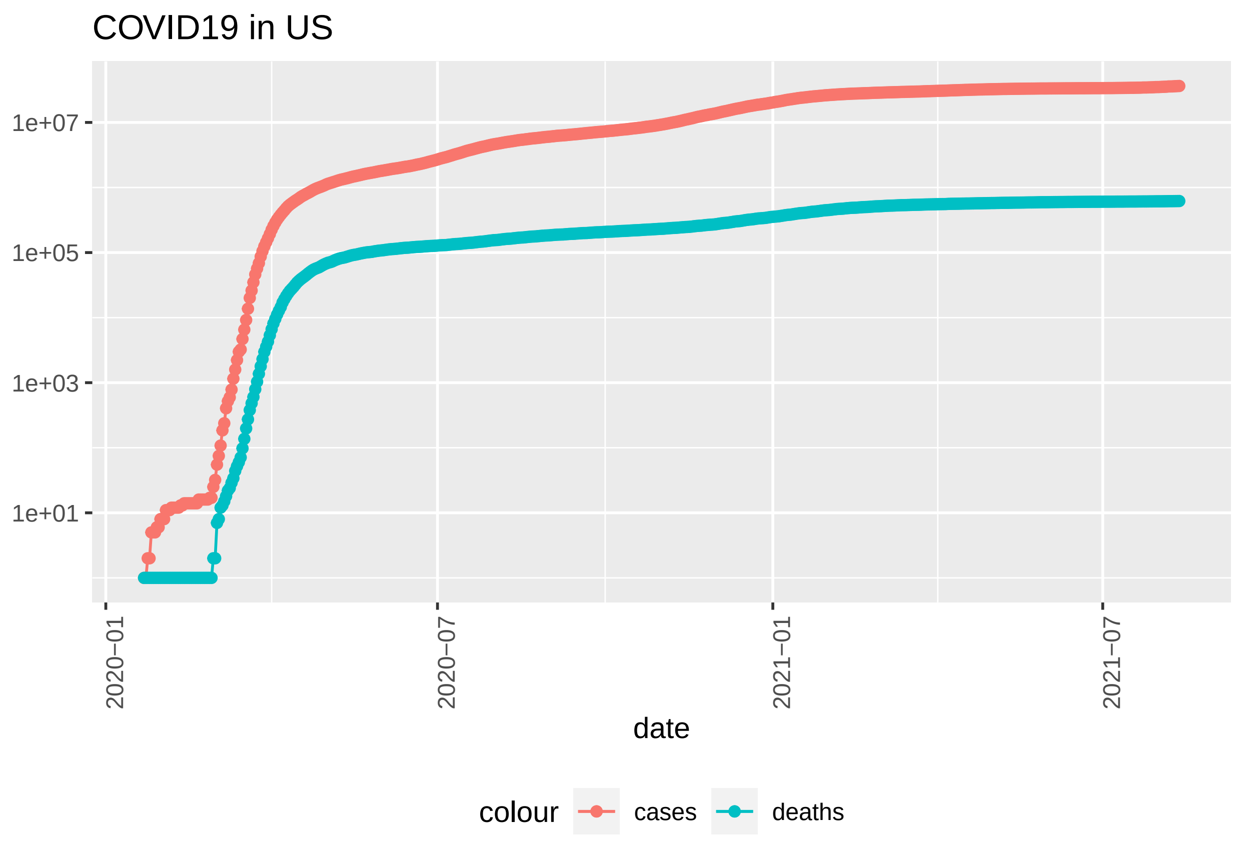
```
## `summarise()` has grouped output by 'Country_Region'. You can override using the `.groups` argument.
```

```
us_totals
```

```
## # A tibble: 569 x 6
##    Country_Region date       cases deaths deaths_per_mill Population
##    <chr>          <date>     <dbl>  <dbl>           <dbl>      <dbl>
##  1 US             2020-01-22     1      1         0.00300  332875137
##  2 US             2020-01-23     1      1         0.00300  332875137
##  3 US             2020-01-24     2      1         0.00300  332875137
##  4 US             2020-01-25     2      1         0.00300  332875137
##  5 US             2020-01-26     5      1         0.00300  332875137
##  6 US             2020-01-27     5      1         0.00300  332875137
##  7 US             2020-01-28     5      1         0.00300  332875137
##  8 US             2020-01-29     6      1         0.00300  332875137
##  9 US             2020-01-30     6      1         0.00300  332875137
## 10 US             2020-01-31     8      1         0.00300  332875137
## # ... with 559 more rows
```

Next, lets visualize the total amount of cases and deaths in the United States since the beginning of COVID. We can see the number of cases increases, substantially, from the beginning to end of 2021. Since, there was a slight increase until March; then the amount of cases have been stagnant. During this time the US lifted their restrictions and the cases didn't drastically increase. This could be due to the effects of the vaccine. More people were returning back to normal lives without causing much effect on the population.

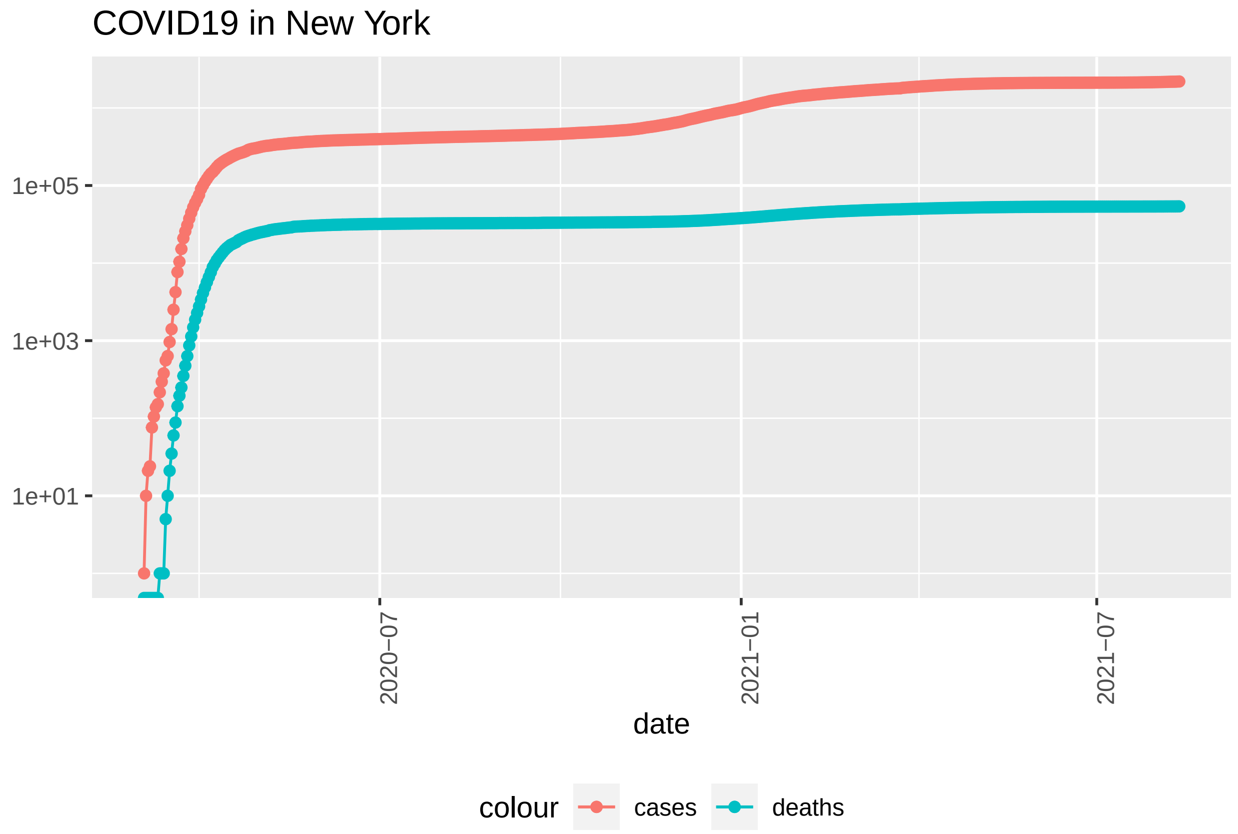
```
us_totals %>%
  filter(cases > 0) %>%
  ggplot(aes(x = date, y = cases)) +
  geom_line(aes(color = "cases")) +
  geom_point(aes(color = "cases")) +
  geom_line(aes(y = deaths, color = "deaths")) +
  geom_point(aes(y = deaths, color = "deaths")) +
  scale_y_log10() +
  theme(legend.position = "bottom",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



Next, lets look at New York. This will help us get a glimpse of what is happening at the state level. At first glance, it seems that the number of COVID cases have leveled off. Is this true?

```
us_by_state %>%
  filter(Province_State == "New York") %>%
  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")) +
  geom_point(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)
```

```
## Warning: Transformation introduced infinite values in continuous y-axis

## Warning: Transformation introduced infinite values in continuous y-axis
```



Analyzing the Data

As the numbers get larger, the total amount of cases loses its meaning. At first, fifty new COVID cases seemed like a lot. However, once we reached 500,000 cases, it's hard to tell the change on a daily bass. Lets add variables to represent the daily, new amount of cases and deaths.

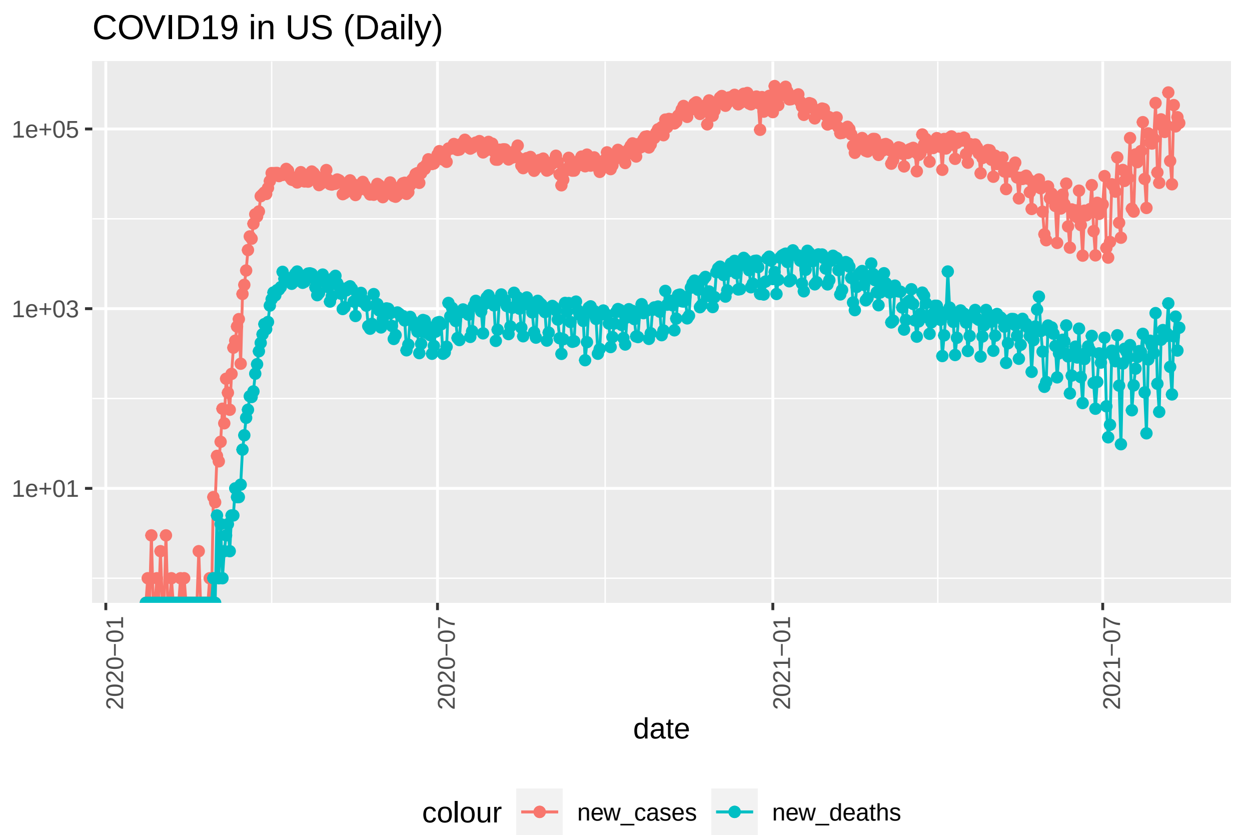
```
us_by_state <- us_by_state %>%
  mutate(new_cases = cases - lag(cases), new_deaths = deaths - lag(deaths))
us_totals <- us_totals %>%
  mutate(new_cases = cases - lag(cases),
         new_deaths = deaths - lag(deaths))

us_by_state
```

```
## # A tibble: 33,002 x 9
##    Province_State Country_Region date       cases deaths deaths_per_mill
##    <chr>          <chr>          <date>     <dbl>  <dbl>           <dbl>
##  1 Alabama        US             2020-01-22     0      0               0
##  2 Alabama        US             2020-01-23     0      0               0
##  3 Alabama        US             2020-01-24     0      0               0
##  4 Alabama        US             2020-01-25     0      0               0
##  5 Alabama        US             2020-01-26     0      0               0
##  6 Alabama        US             2020-01-27     0      0               0
##  7 Alabama        US             2020-01-28     0      0               0
##  8 Alabama        US             2020-01-29     0      0               0
##  9 Alabama        US             2020-01-30     0      0               0
## 10 Alabama        US             2020-01-31     0      0               0
## # ... with 32,992 more rows, and 3 more variables: Population <dbl>,
## #   new_cases <dbl>, new_deaths <dbl>
```

Now, lets visualize the amount of new cases per day. We can see that there was a decrease after March. Again, this is when many American citizens began receiving the COVID vaccine. However, with the new Delta variant, we can see a rise in July 2021. In fact, the amount is close to the highest daily amount since January 2021 - the peak of COVID cases.

```
us_totals %>%
  filter(cases > 0) %>%
  ggplot(aes(x = date, y = new_cases)) +
  geom_line(aes(color = "new_cases")) +
  geom_point(aes(color = "new_cases")) +
  geom_line(aes(y = new_deaths, color = "new_deaths")) +
  geom_point(aes(y = new_deaths, color = "new_deaths")) +
  scale_y_log10() +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90)) +
  labs(title = "COVID19 in US (Daily)", y = NULL)
```

```
## Warning: Transformation introduced infinite values in continuous y-axis

## Warning: Transformation introduced infinite values in continuous y-axis

## Warning: Transformation introduced infinite values in continuous y-axis

## Warning: Transformation introduced infinite values in continuous y-axis

## Warning: Removed 1 row(s) containing missing values (geom_path).

## Warning: Removed 1 rows containing missing values (geom_point).

## Warning: Removed 1 row(s) containing missing values (geom_path).

## Warning: Removed 1 rows containing missing values (geom_point).
```



Which states are handling COVID the best? Which are handling it the worst? We'll see the 10 states with smallest/largest deaths per thousand. Looking at the first table, we can see the states with the least deaths per thousand. Most of these states are areas that are rural or tourist attractions. Are Hawaii and the Virgin Islands doing better due to the drop in tourist activity? On the other hand, Alaska and Utah are large states with a smaller population; citizens are more spread out and may not come in contact as often.

```
us_state_totals <- us_by_state %>%
  group_by(Province_State) %>%
  summarize(deaths = max(deaths), cases = max(cases),
            population = max(Population),
            cases_per_thou = 1000* cases / population,
            deaths_per_thou = 1000 * deaths / population) %>%
  filter(cases > 0, population > 0)

us_state_totals %>%
  slice_min(deaths_per_thou, n=10) %>%
  select(deaths_per_thou, cases_per_thou, everything())
```

```
## # A tibble: 10 x 6
##    deaths_per_thou cases_per_thou Province_State        deaths  cases population
##              <dbl>          <dbl> <chr>                  <dbl>  <dbl>      <dbl>
##  1          0.0363           3.32 Northern Mariana Isl~      2    183      55144
##  2          0.382           47.5  Virgin Islands            41   5093     107268
##  3          0.386           34.2  Hawaii                   546  48397    1415872
##  4          0.423           41.5  Vermont                  264  25883     623989
##  5          0.544          107.   Alaska                   403  79485     740995
##  6          0.672           53.7  Maine                    903  72119    1344212
##  7          0.694           56.1  Oregon                  2928 236698    4217737
##  8          0.706           41.5  Puerto Rico             2651 155709    3754939
##  9          0.786          138.   Utah                    2521 443488    3205958
## 10          0.816           66.2  Washington              6215 504132    7614893
```

Compared to the states with the lower amounts of deaths per thousands, the states with the higher amount of cases have a larger population. Congested areas may have more occurrences of contact with others; this may cause more cases/deaths. When comparing these states, we need to question how these cases/deaths recorded? Also, we need to consider factors that may cause a difference that isn't recorded. One factor could be health care quality or citizens' access to health care.

```
us_state_totals %>%
  slice_max(deaths_per_thou, n=10) %>%
  select(deaths_per_thou, cases_per_thou, everything())
```

```
## # A tibble: 10 x 6
##    deaths_per_thou cases_per_thou Province_State deaths   cases population
##              <dbl>          <dbl> <chr>                <dbl>   <dbl>      <dbl>
##  1            3.00           119. New Jersey       26672 1055252    8882190
##  2            2.77           113. New York         53828 2192224   19453561
##  3            2.63           106. Massachusetts    18131  733188    6892503
##  4            2.60           126. Mississippi       7730  376124    2976149
##  5            2.59           148. Rhode Island      2744  157188    1059361
##  6            2.53           131. Arizona          18412  955767    7278717
##  7            2.47           131. Louisiana        11462  607228    4648794
##  8            2.39           127. Alabama          11724  623919    4903185
##  9            2.33           101. Connecticut       8307  361294    3565287
## 10            2.32           143. South Dakota      2053  126611     884659
```

COVID by Political Party

Although COVID has taken another recent surge, the United States political parties are still divided on what actions we should take. Republicans believe the country should operate as normal and the disease will either go away, or become something we adapt to. On the other hand, Democrats believe we should take as much precaution as possible; also, Democrats have a stronger push for national vaccination requirements. Lets take a look and see if the polarizing views affect the political parties differently. First, we'll create vectors consisting of members of the political parties. We're defining each state's political party as who they voted for in the 2020 presidential election. We discovered this from:

https://www.archives.gov/electoral-college/2020. Guam, Puerto Rico, The Virgin Islands nor The Northern Mariana Islands voted in the U.S. election; therefore, we will remove their cases. and not include them in this part of the analysis.

```
republican <-c("Alabama", "Alaska", "Arkansas", "Florida", "Idaho", "Indiana",
               "Iowa", "Kansas", "Kentucky", "Lousiana", "Mississippi",
               "Missouri", "Montana", "Nebraska", "North Carolina",
               "North Dakota", "Ohio", "Oklahoma", "South Carolina",
               "South Dakota", "Tennessee", "Texas", "Utah", "West Virginia",
               "Wyoming")
democrat <- c("Arizona", "California", "Colorado", "Connecticut", "Deleware",
              "District of Columbia", "Georgia", "Hawaii",
              "Illinois", "Maine", "Maryland", "Massachusetts", "Michigan",
              "Minnesota",  "Nevada", "New Hampshire", "New Jersey",
              "New Mexico", "New York", "Oregon", "Pennsylvania",
              "Rhode Island", "Vermont", "Virginia", "Wisonsin", "Washington")
```

Next, we need to create the 'party' column so we can label each state's political party with how they voted in the 2020 presidential election.

```
us_by_party <- us_by_state %>%
    filter(us_by_state$Province_State != "Guam" &
           us_by_state$Province_State != "Puerto Rico" &
           us_by_state$Province_State != "Virgin Islands" &
           us_by_state$Province_State != "Northern Mariana Islands") %>%
  group_by(Province_State) %>%
  mutate(party = ifelse(Province_State %in% republican,'R','D')) %>%
  group_by(party, date) %>%
  summarise(cases = sum(cases),
            deaths = sum(deaths),
            population = sum(Population),
            cases_per_thou = 1000 * cases / population,
            deaths_per_thou = 1000 * deaths / population) %>%
  mutate(new_cases = cases - lag(cases), new_deaths = deaths - lag(deaths))
```

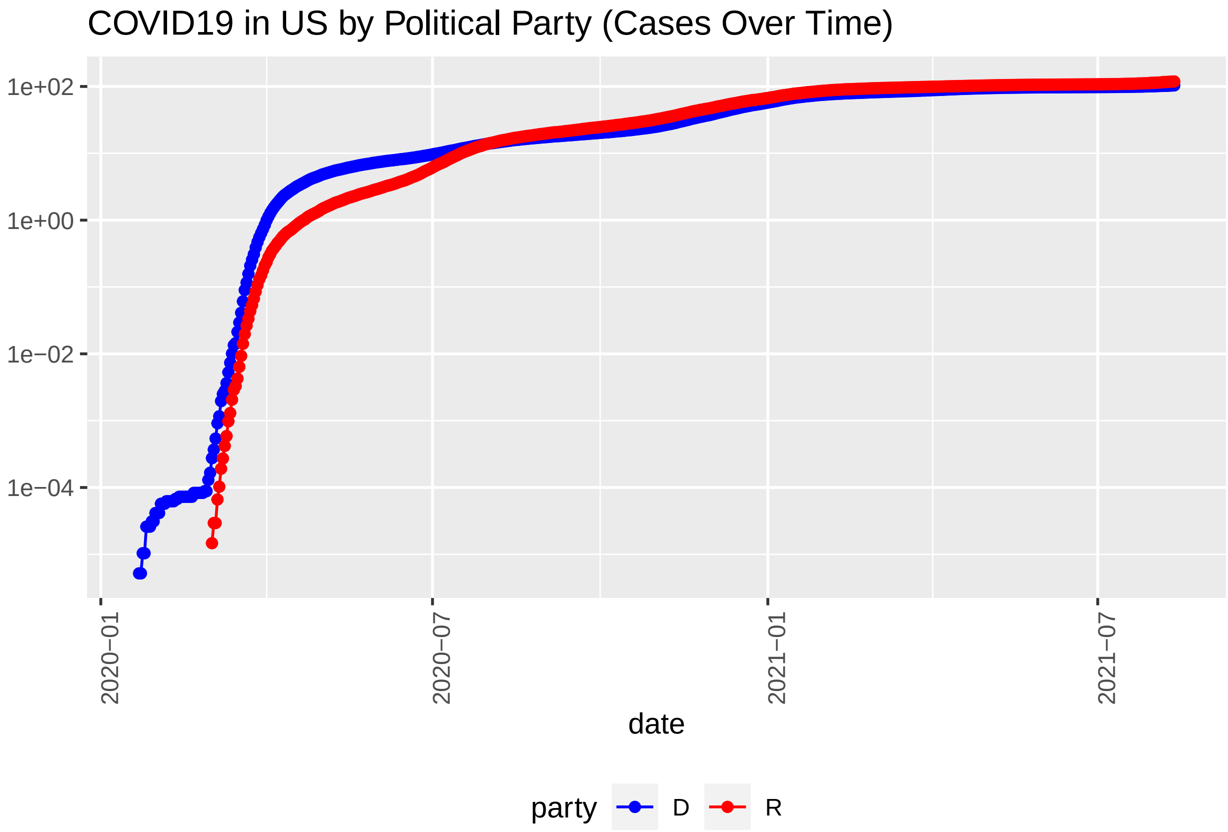
```
## `summarise()` has grouped output by 'party'. You can override using the `.groups` argument.
```

```
us_by_party
```

```
## # A tibble: 1,138 x 9
## # Groups:   party [2]
##    party date       cases deaths population cases_per_thou deaths_per_thou
##    <chr> <date>     <dbl>  <dbl>      <dbl>          <dbl>           <dbl>
##  1 D     2020-01-22     1      0  192661280     0.00000519               0
##  2 D     2020-01-23     1      0  192661280     0.00000519               0
##  3 D     2020-01-24     2      0  192661280     0.0000104                0
##  4 D     2020-01-25     2      0  192661280     0.0000104                0
##  5 D     2020-01-26     5      0  192661280     0.0000260                0
##  6 D     2020-01-27     5      0  192661280     0.0000260                0
##  7 D     2020-01-28     5      0  192661280     0.0000260                0
##  8 D     2020-01-29     6      0  192661280     0.0000311                0
##  9 D     2020-01-30     6      0  192661280     0.0000311                0
## 10 D     2020-01-31     8      0  192661280     0.0000415                0
## # ... with 1,128 more rows, and 2 more variables: new_cases <dbl>,
## #   new_deaths <dbl>
```

First, let's view how COVID surged in each party. We can see the democratic party suffered from COVID a bit earlier than the republican party; however, it seems both became stagnant around January 2021.

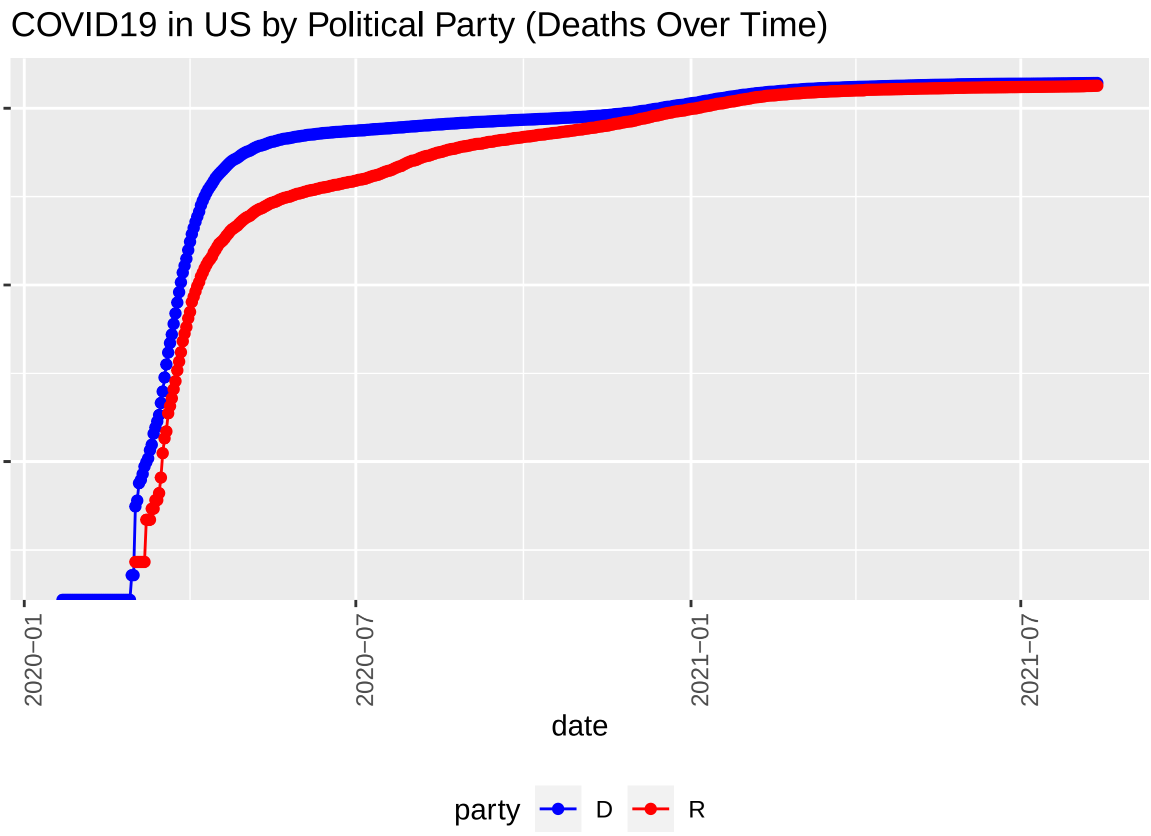
```
us_by_party %>%
  filter(cases > 0) %>%
  ggplot(aes(x = date, y = cases_per_thou)) +
  geom_line(aes(color = party)) +
  geom_point(aes(color = party)) +
  scale_y_log10() +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90)) +
  labs(title = "COVID19 in US by Political Party (Cases Over Time)", y = NULL) +
  scale_color_manual(breaks = c("D", "R"), values=c("blue", "red"))
```



Similar to the number of cases, states in the democratic party suffered from more deaths than the republican party early on; however, the number of cases seemed to even out towards the end.

```
us_by_party %>%
  filter(cases > 0) %>%
  ggplot(aes(x = date, y = deaths_per_thou)) +
  geom_line(aes(color = party)) +
  geom_point(aes(color = party)) +
  scale_y_log10() +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90)) +
  labs(title = "COVID19 in US by Political Party (Deaths Over Time)",
       y = NULL) +
  scale_color_manual(breaks = c("D", "R"), values=c("blue", "red"))
```

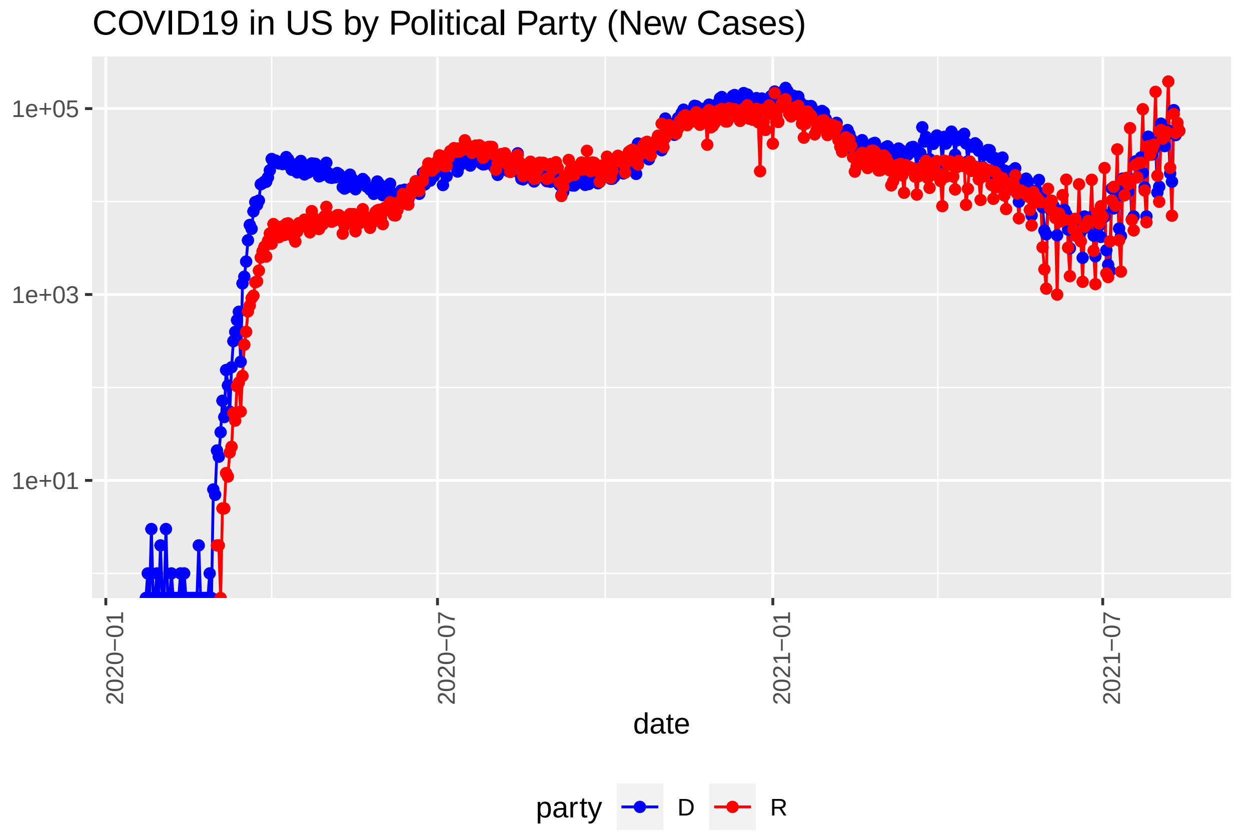
```
## Warning: Transformation introduced infinite values in continuous y-axis

## Warning: Transformation introduced infinite values in continuous y-axis
```



As we know, a look at the overall amount of cases is not a good measurement of how COVID is affecting the country. Next we're looking at the number of new cases each day. As we can see, after July of 2020, the amount of new COVID cases seem to be the same.

```
us_by_party %>%
  filter(cases > 0) %>%
  ggplot(aes(x = date, y = new_cases)) +
  geom_line(aes(color = party)) +
  geom_point(aes(color = party)) +
    scale_y_log10() +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90)) +
  labs(title = "COVID19 in US by Political Party (New Cases)", y = NULL) +
  scale_color_manual(breaks = c("D", "R"), values=c("blue", "red"))
```

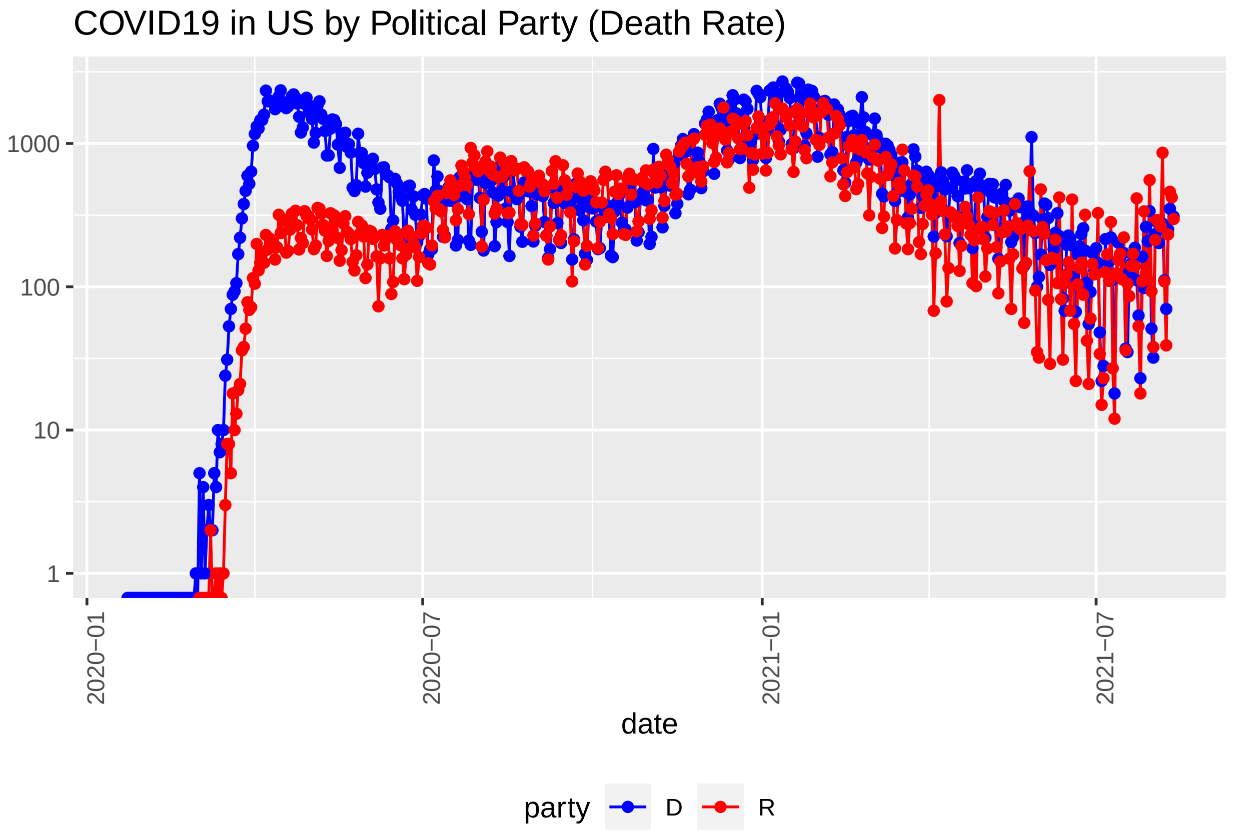
```
## Warning: Transformation introduced infinite values in continuous y-axis

## Warning: Transformation introduced infinite values in continuous y-axis

## Warning: Removed 1 row(s) containing missing values (geom_path).

## Warning: Removed 1 rows containing missing values (geom_point).
```



Similar to the number of new cases, the number of daily new deaths has been identical between political parties (other than between March 2020 - July 2020).

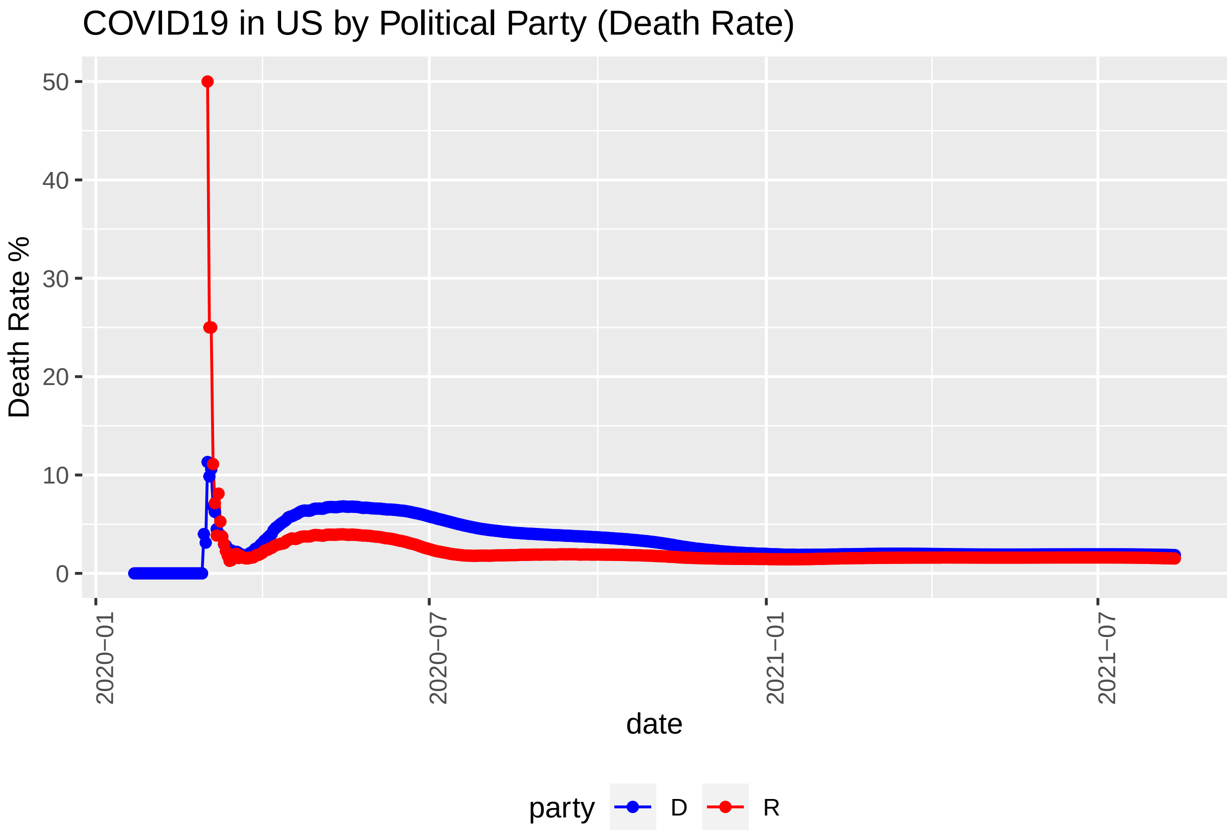
```
us_by_party %>%
  filter(cases > 0) %>%
  ggplot(aes(x = date, y = new_deaths)) +
  geom_line(aes(color = party)) +
  geom_point(aes(color = party)) +
    scale_y_log10() +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90)) +
  labs(title = "COVID19 in US by Political Party (Death Rate)", y = NULL) +
  scale_color_manual(breaks = c("D", "R"), values=c("blue", "red"))
```

```
## Warning in self$trans$transform(x): NaNs produced

## Warning: Transformation introduced infinite values in continuous y-axis

## Warning in self$trans$transform(x): NaNs produced

## Warning: Transformation introduced infinite values in continuous y-axis

## Warning: Removed 1 row(s) containing missing values (geom_path).

## Warning: Removed 2 rows containing missing values (geom_point).
```



How many of the COVID cases resulted in deaths? Does the state's political party make a difference? First, lets create a variable called "Death Rate" which will be the number of deaths divided by the number of cases. As we can see, there seems to be an outlier around March 2020. Let's take a closer look at these cases.

```r
us_by_party <- us_by_party %>%
  mutate(death_rate = 100* (deaths / cases))
us_by_party_rate_plot <- us_by_party %>%
  filter(cases > 0 & death_rate != Inf) %>%
  ggplot(aes(x = date, y = death_rate)) +
  geom_line(aes(color = party)) +
  geom_point(aes(color = party)) +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90)) +
  labs(title = "COVID19 in US by Political Party (Death Rate)",
       y="Death Rate %") +
  scale_color_manual(breaks = c("D", "R"), values=c("blue", "red"))
us_by_party_rate_plot
```



These dates are from the beginning of COVID when there weren't many cases. Let's recreate the plot while ignoring these outliers.

```
us_by_party %>% filter(death_rate > 10 & death_rate != Inf) %>%
  select(c("party", "date", "cases", "deaths", "death_rate"))
```

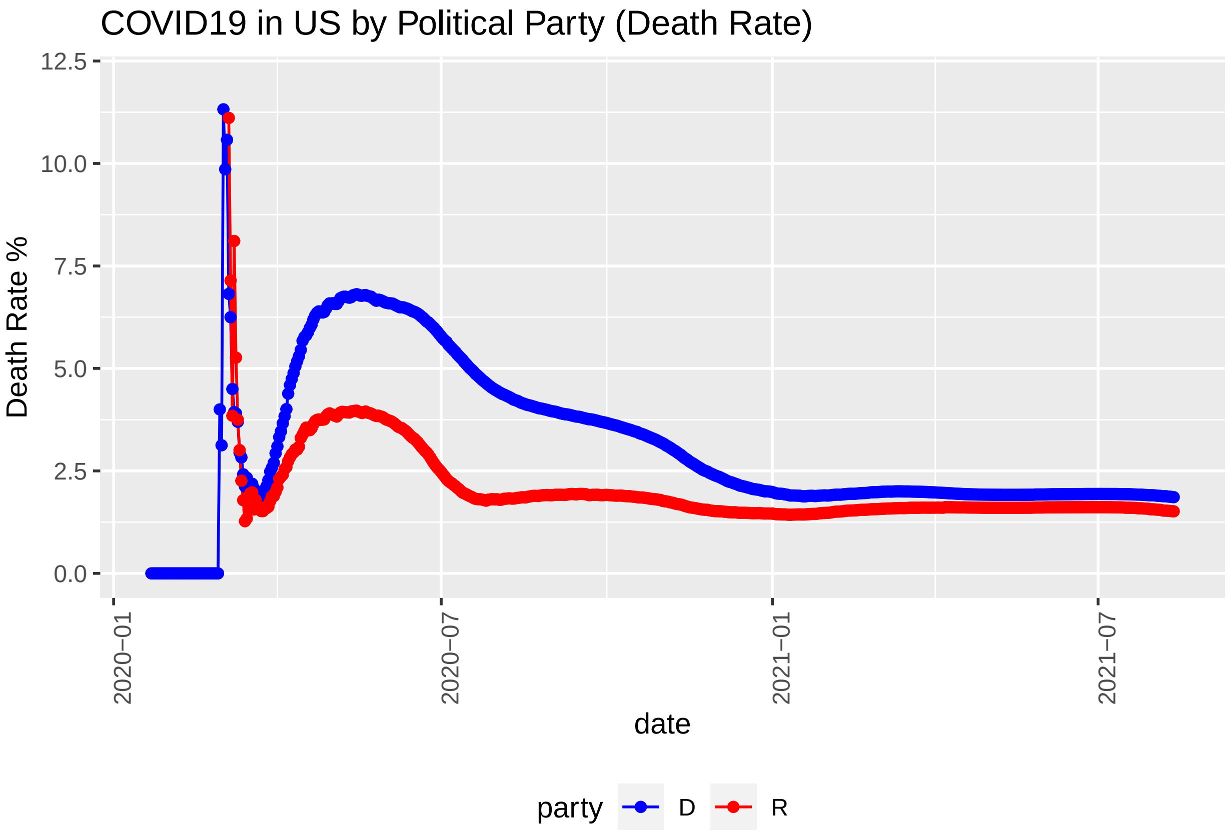
```
## # A tibble: 6 x 5
## # Groups:   party [2]
##   party date       cases deaths death_rate
##   <chr> <date>     <dbl>  <dbl>      <dbl>
## 1 D     2020-03-02    53      6       11.3
## 2 D     2020-03-04   104     11       10.6
## 3 R     2020-03-02     2      1       50
## 4 R     2020-03-03     4      1       25
## 5 R     2020-03-04     4      1       25
## 6 R     2020-03-05     9      1       11.1
```

Looking at the graph again, it seems the death rate has become rather consistent. Since there's a large amount of total cases, maybe it would help to define death rate as the number of new_deaths divided by the number of new_cases.

```
us_by_party_rate_plot + ylim(0, 12)
```

```
## Warning: Removed 3 row(s) containing missing values (geom_path).
```

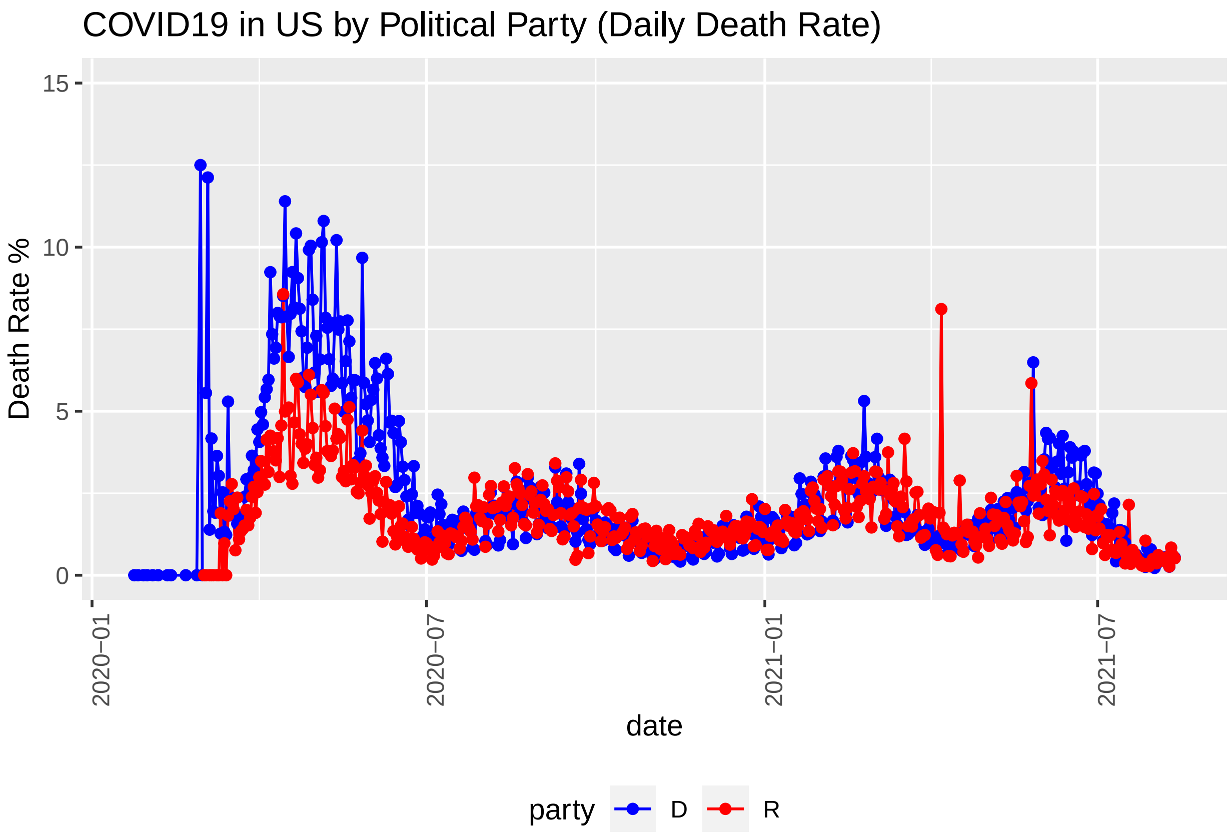
```
## Warning: Removed 3 rows containing missing values (geom_point).
```



Disregarding the three outliers in March of 2020, we can see the death rate of new cases has fluctuated between 0 and 5%, for the most part. In fact, this past month (July 2021) has had a period of lowest death rates since COVID began. Similar to the other cases between COVID data, there isn't much of a difference between political parties. Maybe the United States push for vaccines have helped?

```
us_by_party <- us_by_party %>%
  mutate(death_rate = 100* (new_deaths / new_cases))
us_by_party_rate_plot <- us_by_party %>%
  filter(cases > 0 & death_rate != Inf) %>%
  ggplot(aes(x = date, y = death_rate)) +
  geom_line(aes(color = party)) +
  geom_point(aes(color = party)) +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90)) +
  labs(title = "COVID19 in US by Political Party (Daily Death Rate)",
       y="Death Rate %") +
  scale_color_manual(breaks = c("D", "R"), values=c("blue", "red"))

us_by_party_rate_plot + ylim(0,15)
```

```
## Warning: Removed 3 rows containing missing values (geom_point).
```



Modeling Data

After analyzing the data, we would like to build linear model to help predict the number of cases in the future. In other words, we would like to predict the number of deaths per thousand, given the number of cases. Looking at the model, it is telling us that we can get the number of deaths per thousand if we subtract .018 from .016 * cases per thousand.

```
  mod <- lm(deaths_per_thou ~ cases_per_thou, data = us_state_totals)
summary(mod)
```

```
##
## Call:
## lm(formula = deaths_per_thou ~ cases_per_thou, data = us_state_totals)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.42858 -0.22114  0.00205  0.21564  1.10333
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -0.01938    0.21890  -0.089     0.93
## cases_per_thou  0.01615    0.00203   7.958 1.31e-10 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.4701 on 53 degrees of freedom
## Multiple R-squared:  0.5444, Adjusted R-squared:  0.5358
## F-statistic: 63.33 on 1 and 53 DF,  p-value: 1.306e-10
```

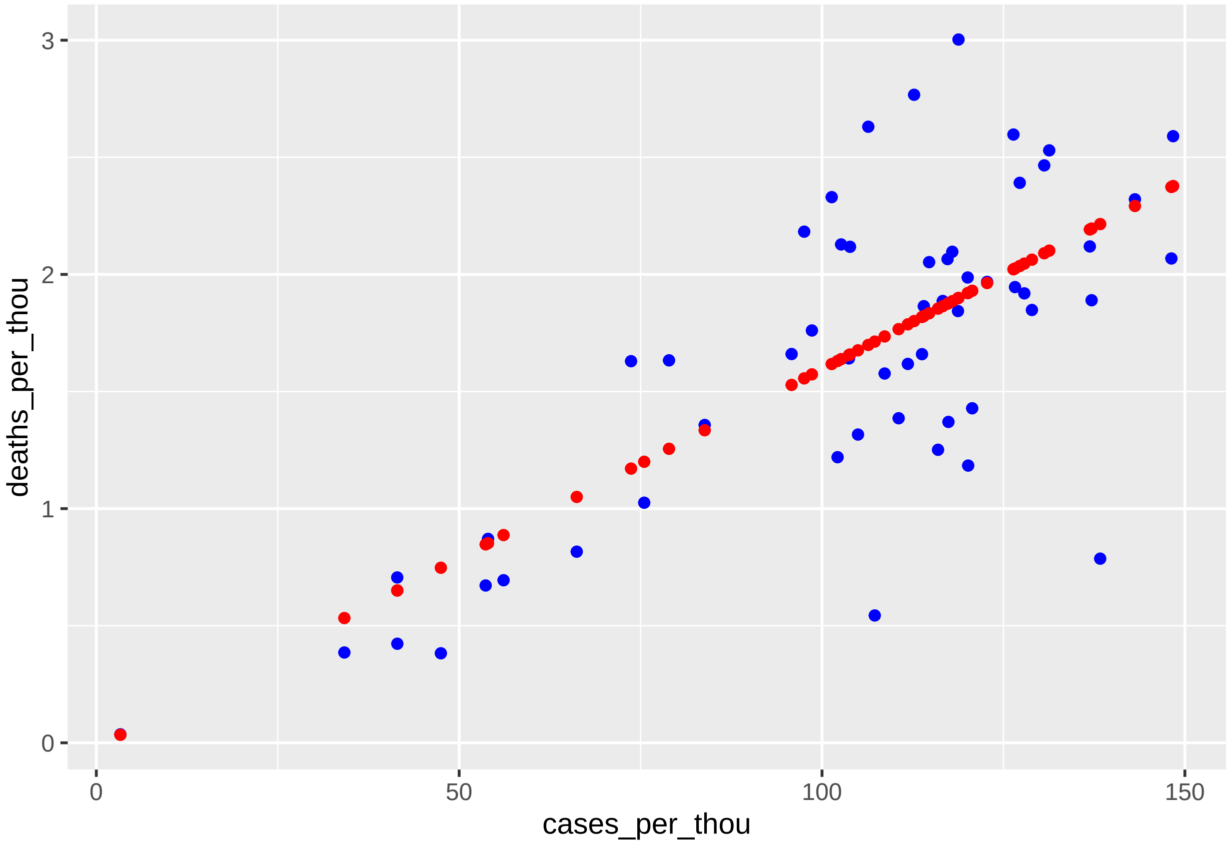
Here we can compare the actual predictions of deaths compared to the actual values. We can see, other than Alaska, Connecticut, and the District of Columbia, our predictions are close.

```
us_tot_w_pred <- us_state_totals %>% mutate(pred = predict(mod))
us_tot_w_pred %>% select(c('Province_State', 'cases_per_thou',
                       'deaths_per_thou', 'pred')) %>% head(10)
```

```
## # A tibble: 10 x 4
##    Province_State       cases_per_thou deaths_per_thou  pred
##    <chr>                         <dbl>           <dbl> <dbl>
##  1 Alabama                       127.            2.39   2.04
##  2 Alaska                        107.            0.544  1.71
##  3 Arizona                       131.            2.53   2.10
##  4 Arkansas                      137.            2.12   2.19
##  5 California                    104.            1.64   1.66
##  6 Colorado                      102.            1.22   1.63
##  7 Connecticut                   101.            2.33   1.62
##  8 Delaware                      117.            1.89   1.86
##  9 District of Columbia           73.7           1.63   1.17
## 10 Florida                       129.            1.85   2.06
```

Let's plot these predictions with our real data. We can see our prediction (in red) follows the same trend as the real COVID data (in blue). The model makes an exact prediction for some and it's largely off for some. It would be great to look further to see which factors are causing this issues.

```
us_tot_w_pred %>% ggplot() + geom_point(aes(x=cases_per_thou, y=deaths_per_thou),
                                         color = "blue")+
  geom_point(aes(x = cases_per_thou, y = pred), color = "red")
```



Conclusion

When viewing this data, it is important to consider potential bias. First, using different variables could lead to different results. For example, implementing population density could have made a difference in our predictions. The closer in contact people are with COVID, the more likely they are to develop symptoms. Also, I chose not include the outliers in the Death Rate plots. Removing them helped us view the difference in the smaller data points. If I wouldn't have removed the outliers, the view of the graph would have made it seem as if the death_rates didn't change. Third, I have my own opinions bout COVID; therefore, I made choices to review different factors than others might have. For example, I chose to look at the differences between political parties when I could have viewed the differences between regions; or, I could have studied the data on a global scale.

Overall, it seems COVID is not disappearing any time soon. In fact, it seems that we're having another surge. However, just because we are still having COVID cases, doesn't mean we are experiencing deaths as much as we have in the past. This could be because of government policy, citizen interactions, vaccine development or other potential factors. Although the U.S. political parties have opposing viewpoints, these opinions don't create different experiences with COVID. Regardless, we should keep developing strategies so we can all go back to our normal lives, while being safe. After all, I still haven't gotten my rematch of Mario Party!

```
utils::sessionInfo()
```

```
## R version 4.0.4 (2021-02-15)
## Platform: x86_64-apple-darwin17.0 (64-bit)
## Running under: macOS Big Sur 10.16
##
## Matrix products: default
## BLAS:   /Library/Frameworks/R.framework/Versions/4.0/Resources/lib/libRblas.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/4.0/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] lubridate_1.7.10 forcats_0.5.1    stringr_1.4.0    dplyr_1.0.5
##  [5] purrr_0.3.4      readr_1.4.0      tidyr_1.1.3      tibble_3.1.0
##  [9] ggplot2_3.3.3    tidyverse_1.3.0
##
## loaded via a namespace (and not attached):
##  [1] tinytex_0.32      tidyselect_1.1.0  xfun_0.24         haven_2.3.1
##  [5] colorspace_2.0-0  vctrs_0.3.6       generics_0.1.0    htmltools_0.5.1.1
##  [9] yaml_2.2.1        utf8_1.2.1        rlang_0.4.10      pillar_1.5.1
## [13] glue_1.4.2        withr_2.4.1       DBI_1.1.1         dbplyr_2.1.0
## [17] modelr_0.1.8      readxl_1.3.1      lifecycle_1.0.0   munsell_0.5.0
## [21] gtable_0.3.0      cellranger_1.1.0  rvest_1.0.0       evaluate_0.14
## [25] labeling_0.4.2    knitr_1.31        curl_4.3          fansi_0.4.2
## [29] highr_0.8         broom_0.7.5       Rcpp_1.0.6        scales_1.1.1
## [33] backports_1.2.1   jsonlite_1.7.2    farver_2.1.0      fs_1.5.0
## [37] hms_1.0.0         digest_0.6.27     stringi_1.5.3     grid_4.0.4
## [41] cli_2.3.1         tools_4.0.4       magrittr_2.0.1    crayon_1.4.1
## [45] pkgconfig_2.0.3   ellipsis_0.3.1    xml2_1.3.2        reprex_1.0.0
## [49] assertthat_0.2.1  rmarkdown_2.7     httr_1.4.2        rstudioapi_0.13
## [53] R6_2.5.0          compiler_4.0.4
```