C3-W1

AN

01 March 2023

Required Packages

Part 1 - Basic Exploration of US Data

The New York Times (the Times) has aggregated reported COVID-19 data from state and local governments and health departments since 2020 and provides public access through a repository on GitHub. One of the data sets provided by the Times is county-level data for cumulative cases and deaths each day. This will be your primary data set for the first two parts of your analysis.

County-level COVID data from 2020, 2021, and 2022 has been imported below. Each row of data reports the cumulative number of cases and deaths for a specific county each day. A FIPS code, a standard geographic identifier, is also provided which you will use in Part 2 to construct a map visualization at the county level for a state.

Additionally, county-level population estimates reported by the US Census Bureau has been imported as well. You will use these estimates to caluclate statistics per 100,000 people.

```
# Import New York Times COVID-19 data
# Import Population Estimates from US Census Bureau

us_counties_2020 <- read_csv("https://raw.githubusercontent.com/nytimes/covid-19
-data/master/us-counties-2020.csv")
```

```
## Rows: 884737 Columns: 6
## ── Column specification ────────────────────────────────────────────────────
───
## Delimiter: ","
## chr  (3): county, state, fips
## dbl  (2): cases, deaths
## date (1): date
##
## 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 mess
age.
```

```
us_counties_2021 <- read_csv("https://raw.githubusercontent.com/nytimes/covid-19
-data/master/us-counties-2021.csv")
```

```
## Rows: 1185373 Columns: 6
## ── Column specification ────────────────────────────────────────────────────
───
## Delimiter: ","
## chr  (3): county, state, fips
## dbl  (2): cases, deaths
## date (1): date
##
## 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 mess
age.
```

```
us_counties_2022 <- read_csv("https://raw.githubusercontent.com/nytimes/covid-19
-data/master/us-counties-2022.csv")
```

```
## Rows: 1188042 Columns: 6
## ── Column specification ────────────────────────────────────────────────────
───
## Delimiter: ","
## chr  (3): county, state, fips
## dbl  (2): cases, deaths
## date (1): date
##
## 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 mess
age.
```

```
us_population_estimates <- read_csv("fips_population_estimates.csv")
```

```
## Rows: 6286 Columns: 7
## ── Column specification ────────────────────────────────────────────────
───
## Delimiter: ","
## chr (2): STNAME, CTYNAME
## dbl (5): fips, STATE, COUNTY, Year, Estimate
##
## 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 mess
age.
```

Question 1

Your first task is to combine and tidy the 2020, 2021, and 2022 COVID data sets and find the total deaths and cases for each day since March 15, 2020 (2020-03-15). The data sets provided from the NY Times also includes statistics from Puerto Rico, a US territory. You may remove these observations from the data as they will not be needed for your analysis. Once you have tidied the data, find the total COVID-19 cases and deaths since March 15, 2020. Write a sentence or two after the code block communicating your results. Use inline code to include the `max_date`, `us_total_cases`, and `us_total_deaths` variables. To write inline code use `r`.

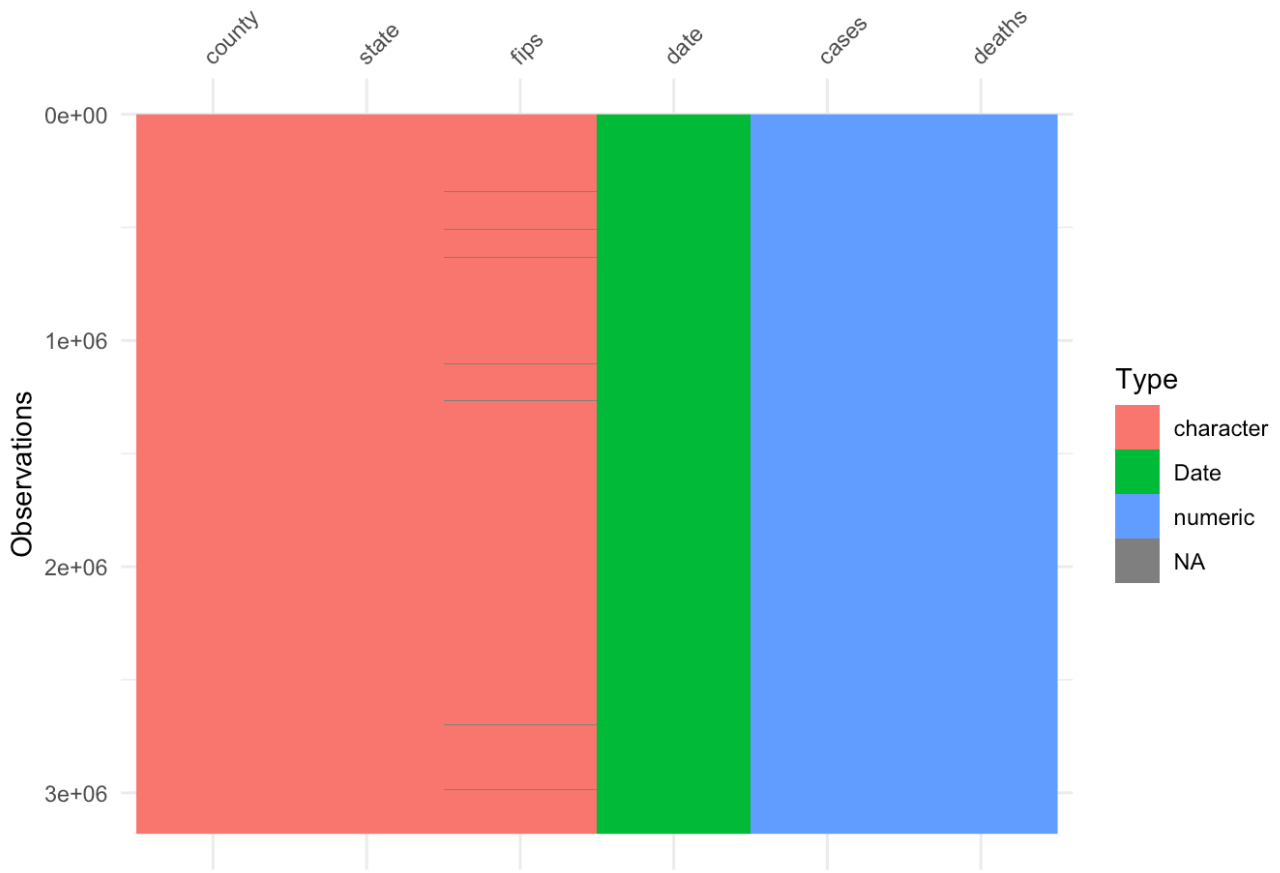
```
# Combine and tidy the 2020, 2021, and 2022 COVID data sets.
# Hint: Review the rbind() documentation to combine the three data sets.
#
## YOUR CODE HERE ##

#combine 2020 and 2021
df<- rbind(us_counties_2020,us_counties_2021)
#add 2022
df_all <- rbind(df, us_counties_2022)
dim(df_all)
```

```
## [1] 3258152       6
```

```
# remove Puerto Rico
df_all <- df_all %>% filter(state != "Puerto Rico")
dim(df_all)
```

```
## [1] 3181427       6
```

```
#check for NAs
vis_dat(df_all, warn_large_data = FALSE)
```




```
# start with March, 15, 2020
df_all <-  df_all %>% filter(date >= "2020-03-15")
dim(df_all)
```

```
## [1] 3179120       6
```

```
# find the most recent date in the data set
max_date <- max(df_all$date)
max_date
```

```
## [1] "2022-12-31"
```

```
# create table q1 with cumulative total cases and total deaths
q1 <- df_all %>% group_by(date) %>% summarise(
  total_deaths = sum(deaths, na.rm=TRUE),
  total_cases = sum(cases, na.rm=TRUE))

q1
```

```
## # A tibble: 1,022 × 3
##    date       total_deaths total_cases
##    <date>            <dbl>       <dbl>
##  1 2020-03-15           68        3595
##  2 2020-03-16           91        4502
##  3 2020-03-17          117        5901
##  4 2020-03-18          162        8345
##  5 2020-03-19          212       12387
##  6 2020-03-20          277       17998
##  7 2020-03-21          359       24507
##  8 2020-03-22          457       33050
##  9 2020-03-23          577       43474
## 10 2020-03-24          783       53899
## # … with 1,012 more rows
```

```
summary(q1)
```

```
##       date             total_deaths      total_cases
##  Min.   :2020-03-15   Min.   :     68   Min.   :    3595
##  1st Qu.:2020-11-25   1st Qu.: 261399   1st Qu.:12794583
##  Median :2021-08-07   Median : 613922   Median :35610854
##  Mean   :2021-08-07   Mean   : 621402   Mean   :45362659
##  3rd Qu.:2022-04-19   3rd Qu.: 984270   3rd Qu.:80160605
##  Max.   :2022-12-31   Max.   :1094296   Max.   :99374764
```

```
# find the maximums
us_total_cases <-max(q1$total_cases)
us_total_deaths <- max(q1$total_deaths)

us_total_cases
```

```
## [1] 99374764
```

```
us_total_deaths
```

```
## [1] 1094296
```

– Communicate your methodology, results, and interpretation here –

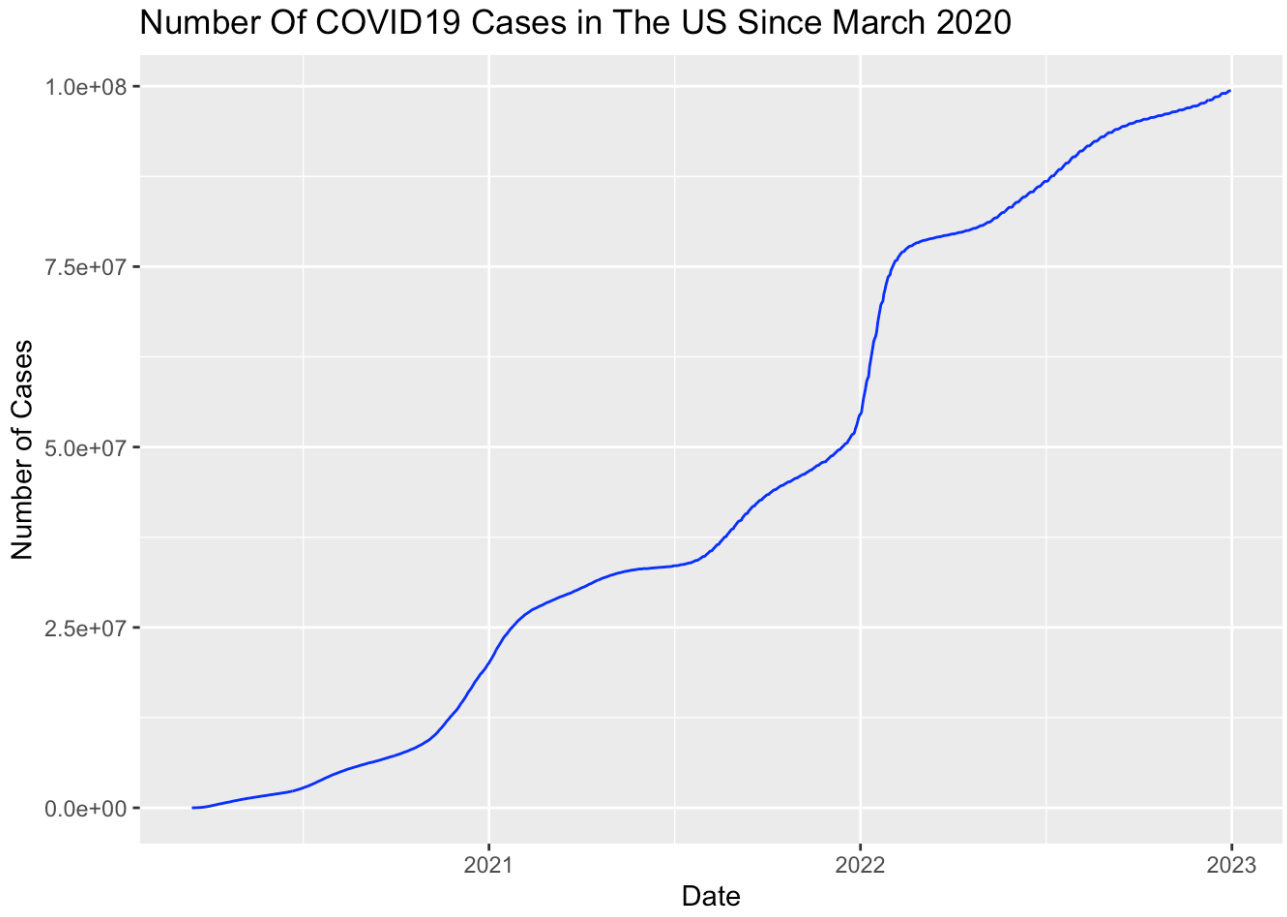
As of December 31, 2022, the total cases in the USA is 9.9374764^{7}, and the total death is 1.094296^{6}

Question 2

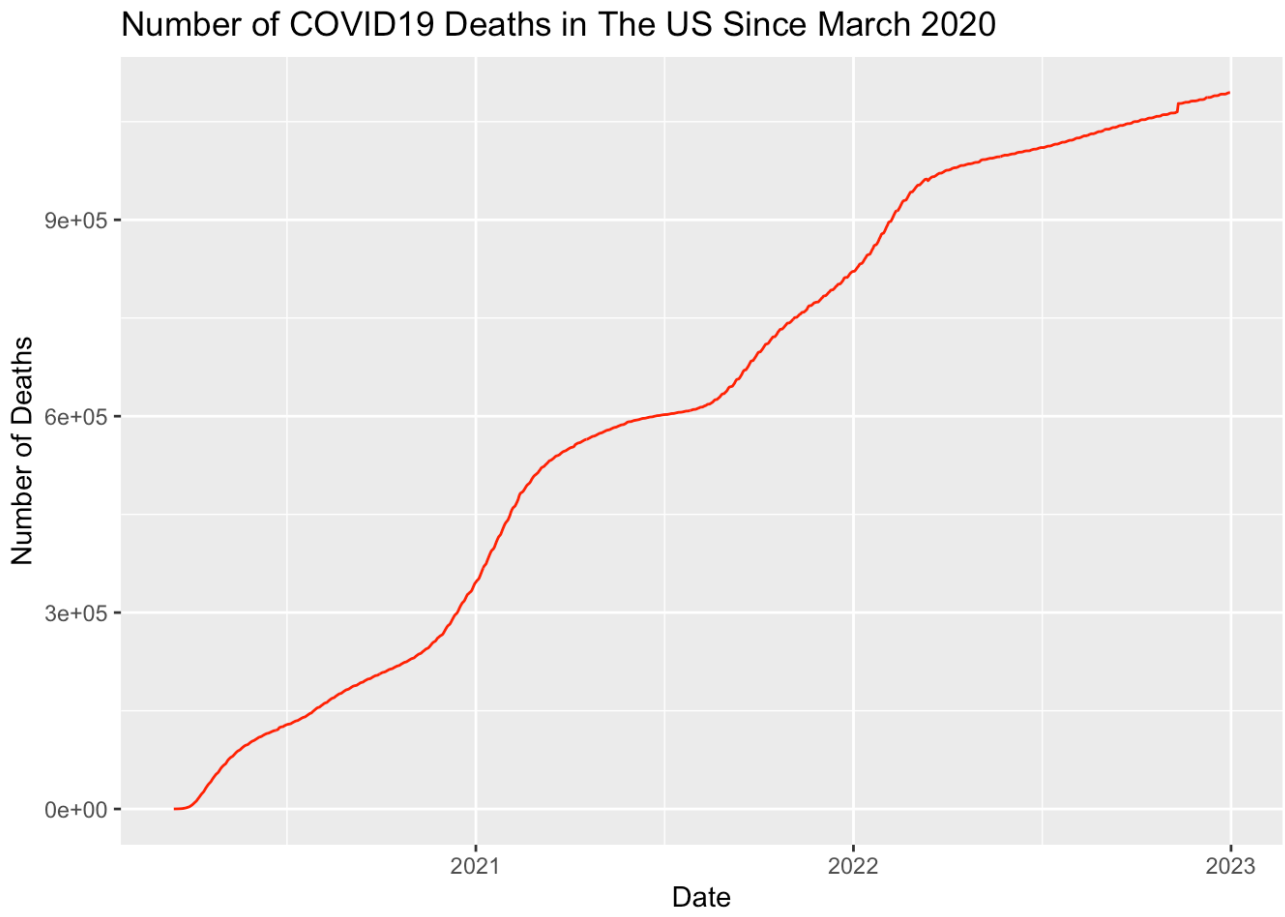
Create a visualization for the total number of deaths and cases in the US since March 15, 2020. Before you create your visualization, review the types of plots you can create using the ggplot2 library and

think about which plots would be effective in communicating your results. After you have created your visualization, write a few sentences describing your visualization. How could the plot be interpreted? Could it be misleading?

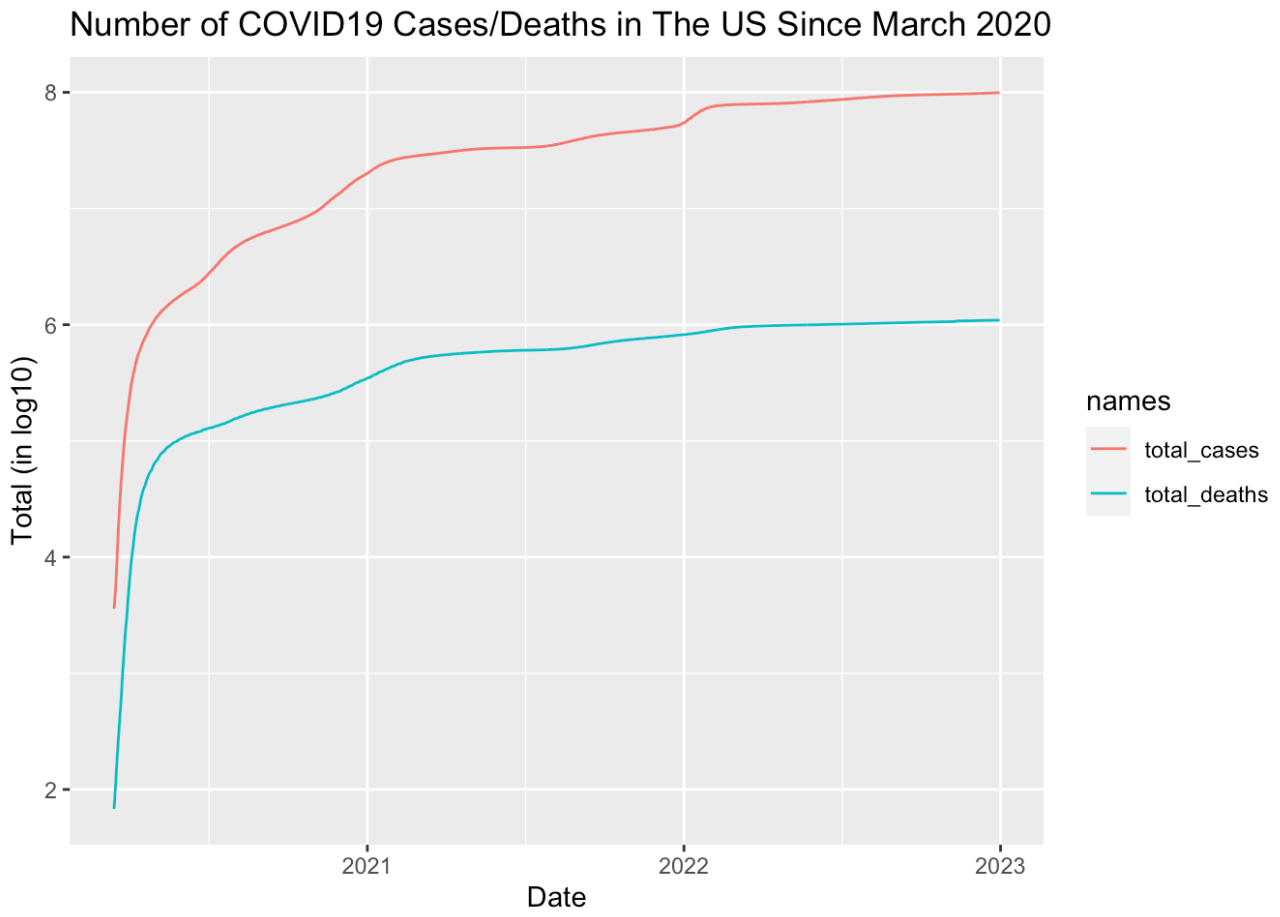
```
# Create a visualization for the total number of US cases and deaths since March
15, 2020.
#
## YOUR CODE HERE ##
# viz for total cases, use time line
ggplot(q1, aes(date, total_cases)) + geom_line(color = "blue") +
  labs(x= "Date",
       y= "Number of Cases") +
  ggtitle("Number Of COVID19 Cases in The US Since March 2020")
```




```
# viz for total deaths, use time line
ggplot(q1, aes(date, total_deaths)) + geom_line(color = "red") +
  labs(x= "Date",
       y= "Number of Deaths") +
  ggtitle("Number of COVID19 Deaths in The US Since March 2020")
```




```
#overlay both variables on the same graph.
#First, use pivot_longer for long form

q2 <- pivot_longer(q1, cols = c("total_deaths", "total_cases") ,
                     names_to = "names",
                     values_to = "total")

# graph, use log10()
ggplot(q2, aes(date, log10(total), color = names)) + geom_line()+
  labs( x = "Date",
        y= "Total (in log10)") +
  ggtitle("Number of COVID19 Cases/Deaths in The US Since March 2020")
```



– Communicate your methodology, results, and interpretation here – From the overlaid graph, it can be seen that there is a correspondence between the number f cases and number of deaths

Question 3

While it is important to know the total deaths and cases throughout the COVID-19 pandemic, it is also important for local and state health officials to know the the number of new cases and deaths each day to understand how rapidly the virus is spreading. Using the table you created in Question 1, calculate the number of new deaths and cases each day and a seven-day average of new deaths and cases. Once you have organized your data, find the days that saw the largest number of new cases and deaths. Write a sentence or two after the code block communicating your results.

```
# Create a new table, based on the table from Question 1, and calculate the numb
er of new deaths and cases each day and a seven day average of new deaths and ca
ses.
#
# Hint: Look at the documentation for lag() when computing the number of new dea
ths and cases and the seven-day averages.
#
#
## YOUR CODE HERE ##

# number of new deaths each day, use lag()
q3 <- q1 %>%
  mutate(delta_deaths_1 = ifelse(lag(total_deaths) < total_deaths,
                                 -(lag(total_deaths) - total_deaths), total_deat
hs))

# number of new cases each day, use lag()
q3 <- q3 %>%
  mutate(delta_cases_1 = ifelse(lag(total_cases) < total_cases,
                                 -(lag(total_cases) - total_cases), total_case
s))
# replace NA with 0
q3 <- replace(q3, is.na(q3), 0)
q3
```

```
## # A tibble: 1,022 × 5
##    date       total_deaths total_cases delta_deaths_1 delta_cases_1
##    <date>            <dbl>       <dbl>          <dbl>         <dbl>
##  1 2020-03-15           68        3595              0             0
##  2 2020-03-16           91        4502             23           907
##  3 2020-03-17          117        5901             26          1399
##  4 2020-03-18          162        8345             45          2444
##  5 2020-03-19          212       12387             50          4042
##  6 2020-03-20          277       17998             65          5611
##  7 2020-03-21          359       24507             82          6509
##  8 2020-03-22          457       33050             98          8543
##  9 2020-03-23          577       43474            120         10424
## 10 2020-03-24          783       53899            206         10425
## # … with 1,012 more rows
```

```
# seven-day averages
# Convert the data frame to a time series (ts) object with zoo package. d = deat
h, c = case
tsd <- zoo(q3$delta_deaths_1, q3$date)
tsc <- zoo(q3$delta_cases_1, q3$date)

# Calculate the 7-day moving average (ma) using rollmean() and lag(),
mad <- rollmean(tsd, k = 7, na.pad = TRUE, align = "right")
mac <- rollmean(tsc, k = 7, na.pad = TRUE, align = "right")

# r = rounded
madr <- as.numeric(round(mad, 2))
macr <- as.numeric(round(mac, 2))
q3b<- q3%>% mutate(delta_deaths_7 = madr, delta_cases_7 = macr)
q3b
```

```
## # A tibble: 1,022 × 7
##    date       total_deaths total_cases delta_deaths_1 delta_ca…¹ delta…² delt
a…³
##    <date>            <dbl>       <dbl>          <dbl>      <dbl>   <dbl>   <d
bl>
##  1 2020-03-15           68        3595              0          0    NA
NA
##  2 2020-03-16           91        4502             23        907    NA
NA
##  3 2020-03-17          117        5901             26       1399    NA
NA
##  4 2020-03-18          162        8345             45       2444    NA
NA
##  5 2020-03-19          212       12387             50       4042    NA
NA
##  6 2020-03-20          277       17998             65       5611    NA
NA
##  7 2020-03-21          359       24507             82       6509    41.6   29
87.
##  8 2020-03-22          457       33050             98       8543    55.6   42
08.
##  9 2020-03-23          577       43474            120      10424    69.4   55
67.
## 10 2020-03-24          783       53899            206      10425    95.1   68
57.
## # … with 1,012 more rows, and abbreviated variable names ¹delta_cases_1,
## #   ²delta_deaths_7, ³delta_cases_7
```

```
# Calculate the total number of new deaths for each day
totals_d <- aggregate(delta_deaths_1 ~ date, q3b, sum)

# Find the day(s) with the highest number of new deaths
max_new_deaths_date<- totals_d$date[which.max(totals_d$delta_deaths_1)]
max_new_deaths_date
```

```
## [1] "2022-12-24"
```

```
# Calculate the total number of new cases for each day
totals_c <- aggregate(delta_cases_1 ~ date, q3b, sum)

# Find the day(s) with the highest number of new deaths
max_new_cases_date<- totals_c$date[which.max(totals_c$delta_cases_1)]
max_new_cases_date
```

```
## [1] "2022-10-08"
```

– Communicate your methodology, results, and interpretation here – The methodology relies on using the function lag() from deplyr The number of daily new cases and deaths increases and decreases with time. The 7-day average for new cases and deaths increases and decreases with time.

Question 4

```
# Create a new table, based on the table from Question 3, and calculate the numb
er of new deaths and cases per 100,000 people each day and a seven day average o
f new deaths and cases per 100,000 people.

## YOUR CODE HERE ##

# remove 2022 from the dada set
q3c <- q3b %>% filter( !date >= "2022-01-01" & date <= "2022-12-31" )


# extract 2020 from us_population_estimates
demographics_2020 <- us_population_estimates %>% filter( Year == 2020)
#compute total US population in 2020
pop_2020 <- sum(demographics_2020$Estimate)
pop_2020
```

```
## [1] 331501080
```

```
# extract 2021 from us_population_estimates
demographics_2021 <- us_population_estimates %>% filter( Year == 2021)
#compute total US population in 2021
pop_2021 <- sum(demographics_2021$Estimate)
pop_2021
```

```
## [1] 331893745
```

```
# create new table q4 from q3 that has new deaths and cases per 100,000 people
# Use case_when to differentiate between 2020 and 2021

q4 <- q3c %>% mutate(date = date,
      total_deaths= case_when(
                  date >= "2020-03-15" & date <= "2020-12-31" ~ total_deaths*10
0000/pop_2020,
                  date >= "2021-01-01" & date <= "2021-12-31" ~ total_deaths*10
0000/pop_2021),

      total_cases= case_when(
                  date >= "2020-03-15" & date <= "2020-12-31" ~ total_cases*100
000/pop_2020,
                  date >= "2021-01-01" & date <= "2021-12-31" ~ total_cases*100
000/pop_2021),

      delta_deaths_1= case_when(
                   date >= "2020-03-15" & date <= "2020-12-31" ~ delta_deaths_1
*100000/pop_2020,
                   date >= "2021-01-01" & date <= "2021-12-31" ~    delta_death
s_1*100000/pop_2021),

      delta_cases_1= case_when(
                  date >= "2020-03-15" & date <= "2020-12-31" ~ delta_cases_1*1
00000/pop_2020,
                  date >= "2021-01-01" & date <= "2021-12-31" ~ delta_cases_1*1
00000/pop_2021),

      delta_deaths_7= case_when(
                   date >= "2020-03-15" & date <= "2020-12-31" ~ delta_deaths_7
*100000/pop_2020,
                   date >= "2021-01-01" & date <= "2021-12-31" ~ delta_deaths_7
*100000/pop_2021),

     delta_cases_7= case_when(
                  date >= "2020-03-15" & date <= "2020-12-31" ~ delta_cases_7*1
00000/pop_2020,
                  date >= "2021-01-01" & date <= "2021-12-31" ~ delta_cases_7*1
00000/pop_2021) )

q4
```

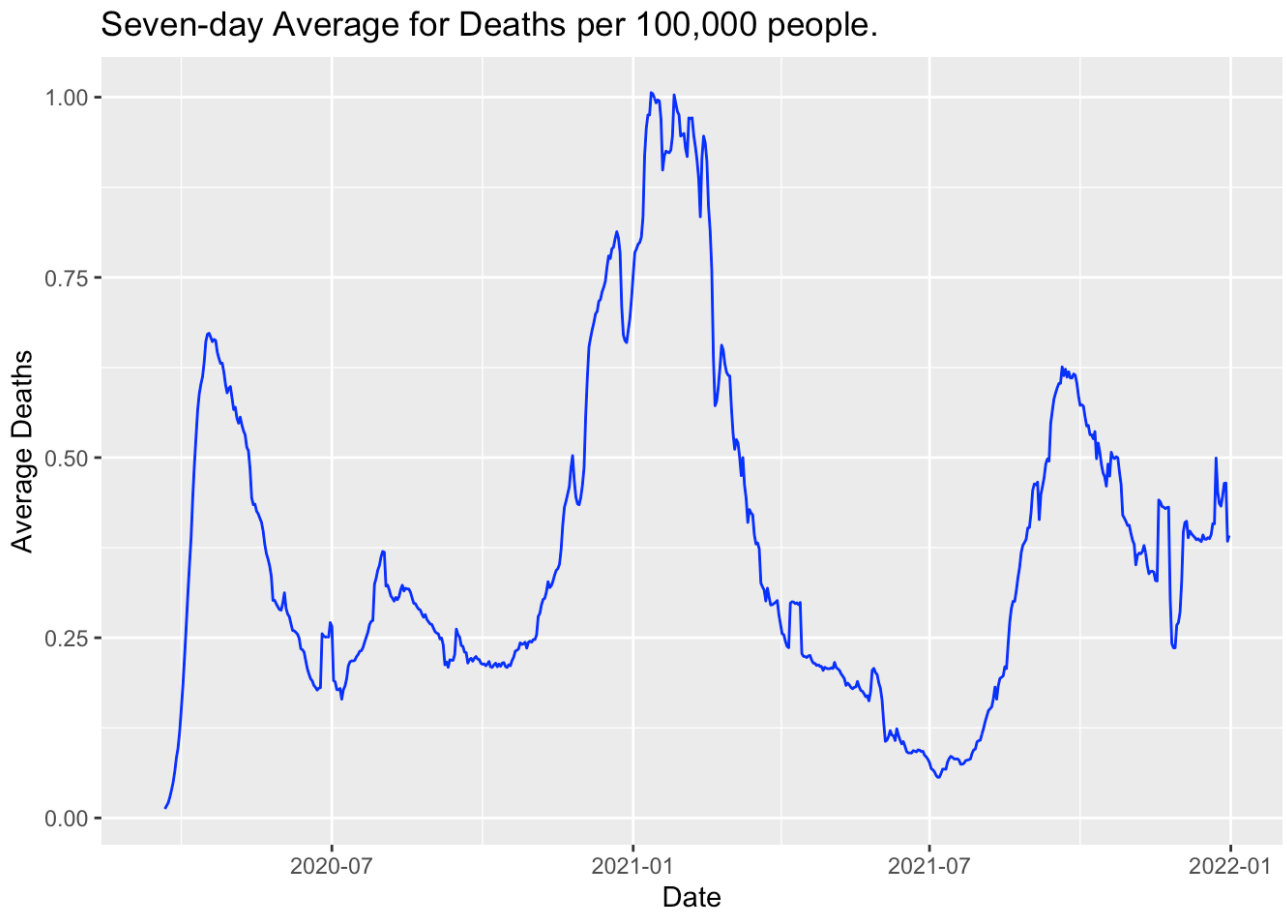
```
## # A tibble: 657 × 7
##    date       total_deaths total_cases delta_deaths_1 delta_ca…¹ delta…² delt
a…³
##    <date>            <dbl>       <dbl>          <dbl>      <dbl>   <dbl>   <d
bl>
##  1 2020-03-15       0.0205        1.08        0            0     NA       NA
##  2 2020-03-16       0.0275        1.36        0.00694      0.274 NA       NA
##  3 2020-03-17       0.0353        1.78        0.00784      0.422 NA       NA
##  4 2020-03-18       0.0489        2.52        0.0136       0.737 NA       NA
##  5 2020-03-19       0.0640        3.74        0.0151       1.22  NA       NA
##  6 2020-03-20       0.0836        5.43        0.0196       1.69  NA       NA
##  7 2020-03-21       0.108         7.39        0.0247       1.96   0.0125   0.
901
##  8 2020-03-22       0.138         9.97        0.0296       2.58   0.0168   1.
27
##  9 2020-03-23       0.174        13.1         0.0362       3.14   0.0209   1.
68
## 10 2020-03-24       0.236        16.3         0.0621       3.14   0.0287   2.
07
## # … with 647 more rows, and abbreviated variable names ¹delta_cases_1,
## #   ²delta_deaths_7, ³delta_cases_7
```

– Communicate your methodology, results, and interpretation here – Adjusted for US population as per 100,000, the 7 day average of new cases and deaths increases and decreases with time.

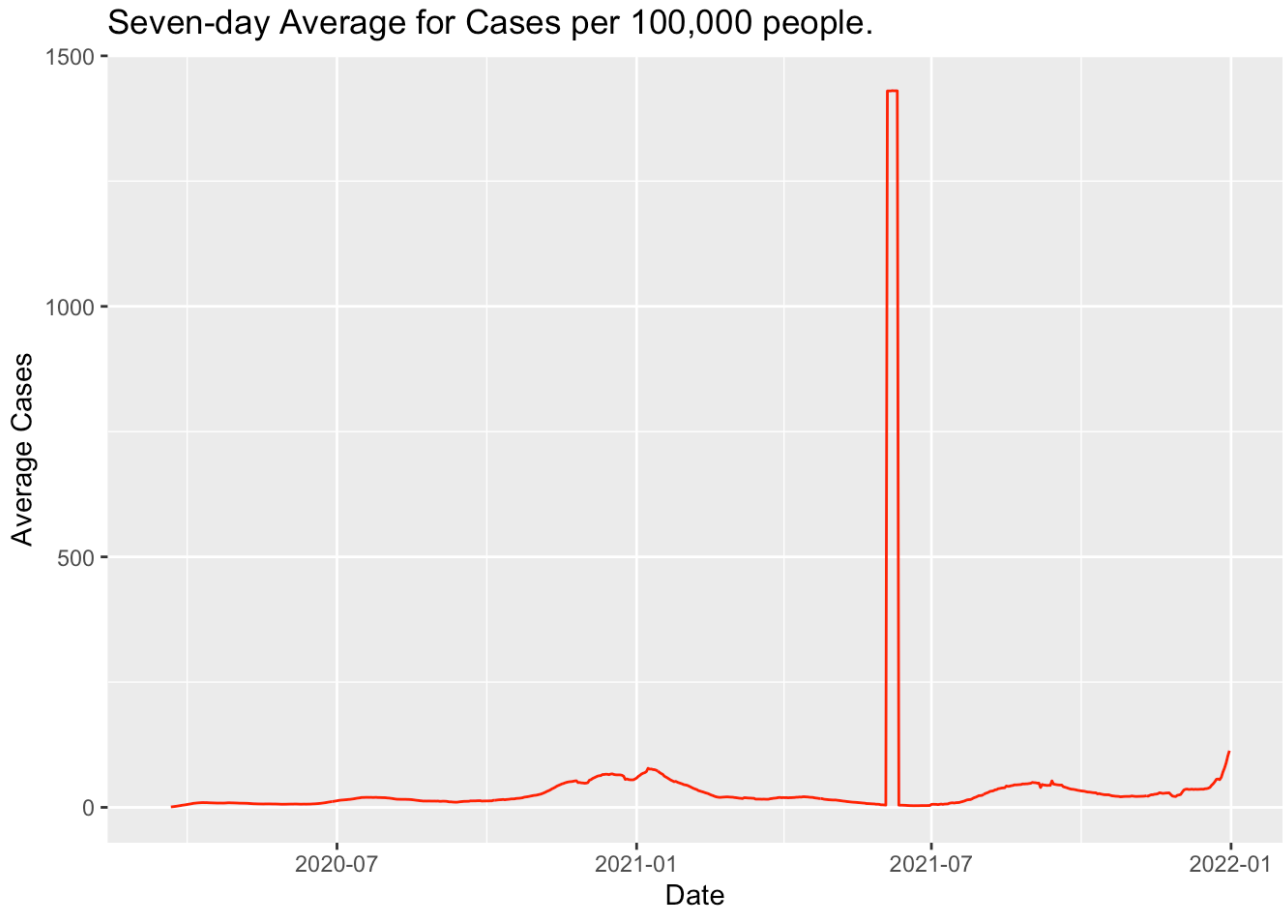
The methodology relies on using the function case_when() from deplyr to separate data from the two years.

Question 5

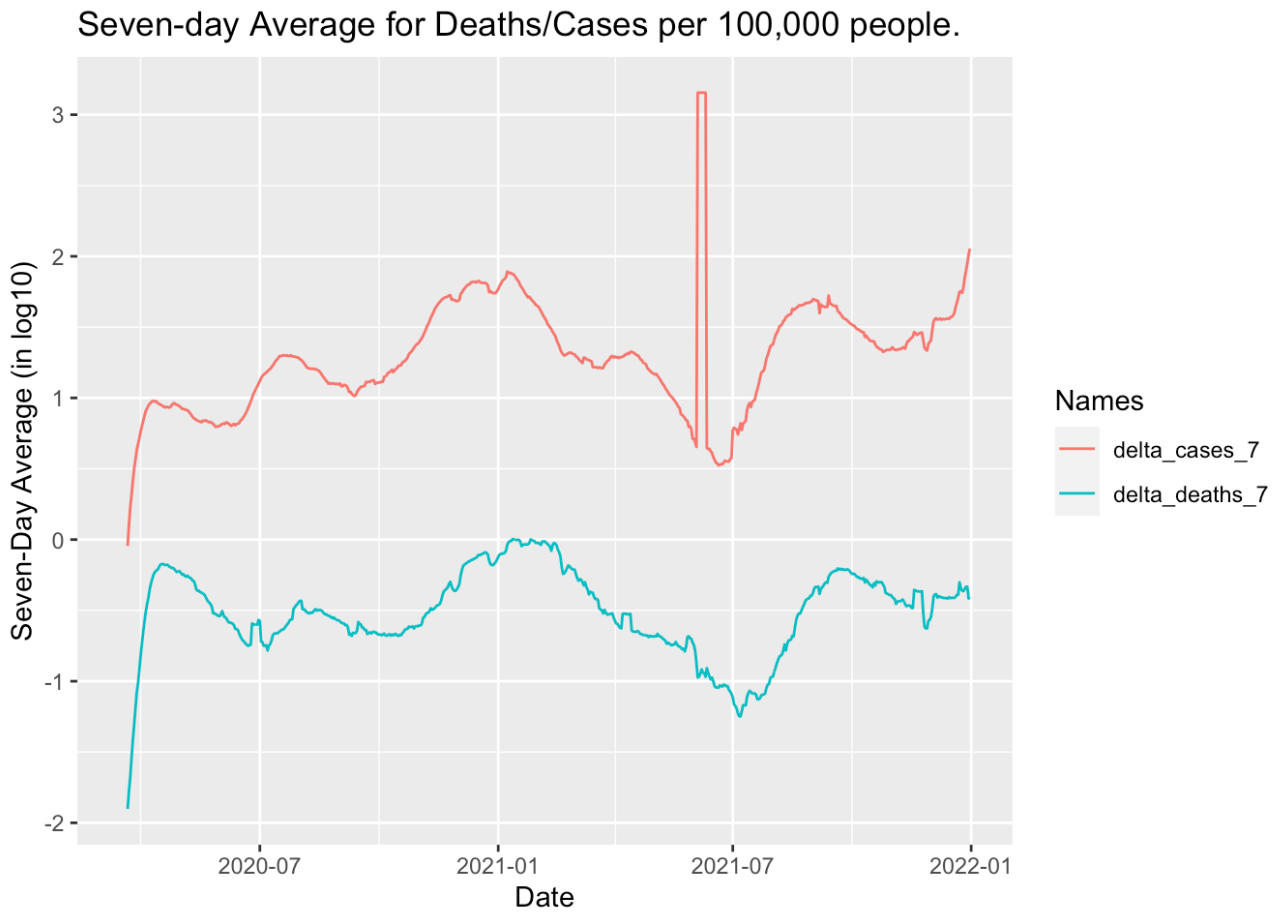
```
# Create a visualization to compare the seven-day average cases and deaths per 1
00,000 people.

# graph for 7-day average for deaths
ggplot(q4, aes(date, delta_deaths_7)) + geom_line(color = "blue") +
  labs(x= "Date",
       y= "Average Deaths") +
  ggtitle("Seven-day Average for Deaths per 100,000 people.")
```




```
# graph for 7-day Cases average for Cases
ggplot(q4, aes(date, delta_cases_7)) + geom_line(color = "red") +
  labs(x= "Date",
       y= "Average Cases") +
  ggtitle("Seven-day Average for Cases per 100,000 people.")
```




```
#overlay both variables on the same graph.
#First, use pivot_longer for long form
q4b <- pivot_longer(q4, cols = c("delta_deaths_7", "delta_cases_7") ,
                    names_to = "Names",
                    values_to = "Seven_day_average")

# Graph
ggplot(q4b, aes(date, log10(Seven_day_average), color = Names)) + geom_line()+
  labs( x = "Date",
        y= "Seven-Day Average (in log10)") +
  ggtitle("Seven-day Average for Deaths/Cases per 100,000 people.")
```



– Communicate your methodology, results, and interpretation here – There is a correspondence between 7-day average new cases and 7-day average new deaths.