COVID-19 Analysis

Qua Vallery

2022-08-13

Description of Data

The COVID-19 Data Repository by the Center for Systems Science and Engineering (CSSE) at John Hopkins University includes daily COVID-19 reports for the US and globally. All data is read in from the daily case reports to generate time series summary tables including confirmed cases and deaths for the US and globally. For this analysis I will be focusing on US confirmed cases and deaths only. The two CSVs from the John Hopkins COVID-19 Data Repository used in the analysis are named time_series_covid19_confirmed_US.csv and time_series_covid19_deaths_US.csv. I developed a third CSV for this analysis named state_codes.csv. Below is a description of all columns for each dataset:

time_series_covid19_confirmed_US.csv

Each record represents total confirmed COVID cases by county, state between January 22, 2020, and August 5, 2022.

- UID: Unique Identifier for each row entry.
- Iso2: Officially assigned two-letter country code identifiers.
- Iso3: Officially assigned three-letter country code identifiers.
- Code3: Officially assigned numeric country code identifiers.
- FIPS: US only. Federal Information Processing Standards code that uniquely identifies counties within the USA.
- Admin2: County name. US only.
- Province_State: Province, state or dependency name.
- Country_Region: Country, region or sovereignty name. The names of locations included on the Website correspond with the official designations used by the U.S. Department of State.
- Lat: Latitude.
- Long_: Longitude.
- Combined_Key: Combination of Admin2, Province_State, Country_Region.

time_series_COVID19_deaths_US.csv

- UID: Unique Identifier for each row entry.
- Iso2: Officially assigned two-letter country code identifiers.
- Iso3: Officially assigned three-letter country code identifiers.
- Code3: Officially assigned numeric country code identifiers.
- FIPS: US only. Federal Information Processing Standards code that uniquely identifies counties within the USA.
- Admin2: County name. US only.
- Province_State: Province, state or dependency name.

- Country_Region: Country, region or sovereignty name. The names of locations included on the Website correspond with the official designations used by the U.S. Department of State.
- Lat: Latitude.
- Long_: Longitude.
- Combined_Key: Combination of Admin2, Province_State, Country_Region.
- Population: Population of the Admin2 (the county).

state_codes.csv

- State: Name of US state.
- State Initial: US State two-letter abbreviation.

Objectives

For this analysis I answered the following questions:

1. What are the total confirmed COVID-19 cases and deaths by state?
2. What does California's COVID-19 deaths look like over time?
3. Which states had the smallest and largest percent increase in deaths compared to 2020?
4. What is the COVID-19 deaths impact look like across different geographic areas of the US?
5. Is there a relationship between the number of COVID-19 cases and deaths?

Import libraries and Data

I started by importing the necessary libraries and data.

```
# load packages
library(lubridate)
```

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

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

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1
```

```
## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x lubridate::as.difftime() masks base::as.difftime()
## x lubridate::date()        masks base::date()
## x dplyr::filter()          masks stats::filter()
## x lubridate::intersect()   masks base::intersect()
## x dplyr::lag()             masks stats::lag()
## x lubridate::setdiff()     masks base::setdiff()
## x lubridate::union()       masks base::union()
```

```
library(tidyquant)
```

```
## Loading required package: PerformanceAnalytics

## Loading required package: xts

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

##
## Attaching package: 'xts'

## The following objects are masked from 'package:dplyr':
##
##     first, last

##
## Attaching package: 'PerformanceAnalytics'

## The following object is masked from 'package:graphics':
##
##     legend

## Loading required package: quantmod

## Loading required package: TTR

## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo

## == Need to Learn tidyquant? ======================================================
## Business Science offers a 1-hour course - Learning Lab #9: Performance Analysis & Portfolio Optimiza
## </> Learn more at: https://university.business-science.io/p/learning-labs-pro </>
```

```
install.packages("usmap", repos = "http://cran.us.r-project.org")
```

```
## Installing package into 'C:/Users/qvall/Documents/R/win-library/4.0'
## (as 'lib' is unspecified)
```

```
library(usmap)

# import files
US_confirmed_tbl <- read_csv("time_series_covid19_confirmed_US.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   .default = col_double(),
##   iso2 = col_character(),
##   iso3 = col_character(),
##   Admin2 = col_character(),
##   Province_State = col_character(),
##   Country_Region = col_character(),
##   Combined_Key = col_character()
## )
## i Use `spec()` for the full column specifications.
```

```
US_deaths_tbl <- read_csv("time_series_covid19_deaths_US.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   .default = col_double(),
##   iso2 = col_character(),
##   iso3 = col_character(),
##   Admin2 = col_character(),
##   Province_State = col_character(),
##   Country_Region = col_character(),
##   Combined_Key = col_character()
## )
## i Use `spec()` for the full column specifications.
```

```
state_codes_tbl <- readr::read_delim("state_codes.csv", delim = "|")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   State = col_character(),
##   state = col_character()
## )
```

Examine Data

I examined the data for any import issues, classification errors, and missing values. There were no import or classification errors. However, there were two columns, FIPS and Admin2, with missing values. I further

examined the columns with missing values in each data set and discovered that the same observations were missing values in both the cases and deaths data sets for the same columns. Ultimately, I decided not to remove or impute the missing values because I wasn't relying on those columns for my analysis.

```
# check for import errors
errors_confirmed_tbl <- readr::problems(US_confirmed_tbl)
errors_deaths_tbl <- readr::problems(US_deaths_tbl)
errors_codes_tbl <- readr::problems(state_codes_tbl)

# check for correct classification of fields

US_confirmed_tbl
```

```
## # A tibble: 3,342 x 938
##         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 929 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>, ...
```

```
US_deaths_tbl
```

```
## # A tibble: 3,342 x 939
##         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 930 more variables: Long_ <dbl>,
## #   Combined_Key <chr>, Population <dbl>, 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>, ...
```

```
# NA check
findNAs_confirmed_tbl <- colSums(is.na(US_confirmed_tbl))

findNAs_deaths_tbl <- colSums(is.na(US_deaths_tbl))
```

```
# Extract rows with NAs

    NAs_Confirmed_FIPS_tbl <- US_confirmed_tbl[is.na(US_confirmed_tbl$FIPS),]
    NAs_Confirmed_Admin2_tbl <- US_confirmed_tbl[is.na(US_confirmed_tbl$Admin2),]

    NAs_Deaths_FIPS_tbl <- US_deaths_tbl[is.na(US_deaths_tbl$FIPS),]
    NAs_Deaths_Admin2_tbl <- US_deaths_tbl[is.na(US_deaths_tbl$Admin2),]

    NAs_Combined_FIPS_tbl <- NAs_Confirmed_FIPS_tbl %>%
        left_join(y = NAs_Deaths_FIPS_tbl, by = "UID")

    NAs_Combined_Admin2_tbl <- NAs_Confirmed_Admin2_tbl %>%
        left_join(y = NAs_Deaths_Admin2_tbl, by = "UID")

    NAs_Combined_FIPS_tbl
```

```
## # A tibble: 10 x 1,876
##         UID iso2.x iso3.x code3.x FIPS.x Admin2.x              Province_State.x
##       <dbl> <chr>  <chr>    <dbl>  <dbl> <chr>                  <chr>
##  1 84070002 US     USA        840     NA Dukes and Nantucket    Massachusetts
##  2 84070005 US     USA        840     NA Federal Correctional ~ Michigan
##  3 84070004 US     USA        840     NA Michigan Department o~ Michigan
##  4 84070003 US     USA        840     NA Kansas City            Missouri
##  5 84070015 US     USA        840     NA Bear River             Utah
##  6 84070016 US     USA        840     NA Central Utah           Utah
##  7 84070017 US     USA        840     NA Southeast Utah         Utah
##  8 84070018 US     USA        840     NA Southwest Utah         Utah
##  9 84070019 US     USA        840     NA TriCounty              Utah
## 10 84070020 US     USA        840     NA Weber-Morgan           Utah
## # ... with 1,869 more variables: Country_Region.x <chr>, Lat.x <dbl>,
## #   Long_.x <dbl>, Combined_Key.x <chr>, 1/22/20.x <dbl>, 1/23/20.x <dbl>,
## #   1/24/20.x <dbl>, 1/25/20.x <dbl>, 1/26/20.x <dbl>, 1/27/20.x <dbl>,
## #   1/28/20.x <dbl>, 1/29/20.x <dbl>, 1/30/20.x <dbl>, 1/31/20.x <dbl>,
## #   2/1/20.x <dbl>, 2/2/20.x <dbl>, 2/3/20.x <dbl>, 2/4/20.x <dbl>,
## #   2/5/20.x <dbl>, 2/6/20.x <dbl>, 2/7/20.x <dbl>, 2/8/20.x <dbl>,
## #   2/9/20.x <dbl>, 2/10/20.x <dbl>, 2/11/20.x <dbl>, 2/12/20.x <dbl>, ...
```

```
    NAs_Combined_Admin2_tbl
```

```
## # A tibble: 6 x 1,876
##      UID iso2.x iso3.x code3.x FIPS.x Admin2.x Province_State.x Country_Region.x
##    <dbl> <chr>  <chr>    <dbl>  <dbl> <chr>    <chr>            <chr>
## 1 1.6 e1 AS     ASM         16     60 <NA>     American Samoa   US
## 2 8.41e7 US     USA        840  88888 <NA>     Diamond Princess US
```

```
## 3 8.41e7 US     USA        840  99999 <NA>     Grand Princess   US
## 4 3.16e2 GU     GUM        316     66 <NA>     Guam             US
## 5 5.8 e2 MP     MNP        580     69 <NA>     Northern Marian~ US
## 6 8.5 e2 VI     VIR        850     78 <NA>     Virgin Islands   US
## # ... with 1,868 more variables: Lat.x <dbl>, Long_.x <dbl>,
## #   Combined_Key.x <chr>, 1/22/20.x <dbl>, 1/23/20.x <dbl>, 1/24/20.x <dbl>,
## #   1/25/20.x <dbl>, 1/26/20.x <dbl>, 1/27/20.x <dbl>, 1/28/20.x <dbl>,
## #   1/29/20.x <dbl>, 1/30/20.x <dbl>, 1/31/20.x <dbl>, 2/1/20.x <dbl>,
## #   2/2/20.x <dbl>, 2/3/20.x <dbl>, 2/4/20.x <dbl>, 2/5/20.x <dbl>,
## #   2/6/20.x <dbl>, 2/7/20.x <dbl>, 2/8/20.x <dbl>, 2/9/20.x <dbl>,
## #   2/10/20.x <dbl>, 2/11/20.x <dbl>, 2/12/20.x <dbl>, 2/13/20.x <dbl>, ...
```

Tidy and Transform

```
# select columns needed for analysis

    US_confirmed_stripped_tbl <- US_confirmed_tbl %>%
        select(-iso2, -iso3, -code3, -FIPS)
    US_deaths_stripped_tbl <- US_deaths_tbl %>%
        select(-iso2, -iso3, -code3, -FIPS)

# pivot longer

    US_confirmed_longer_tbl <- US_confirmed_stripped_tbl %>%
        pivot_longer(
            cols = `1/22/20`: `8/5/22`,
            names_to = "Date",
            values_to = "Total Confirmed"
        )

    US_confirmed_longer_tbl %>% glimpse()
```

```
## Rows: 3,098,034
## Columns: 9
## $ UID               <dbl> 84001001, 84001001, 84001001, 84001001, 84001001, 84~
## $ Admin2            <chr> "Autauga", "Autauga", "Autauga", "Autauga", "Autauga~
## $ Province_State    <chr> "Alabama", "Alabama", "Alabama", "Alabama", "Alabama~
## $ Country_Region    <chr> "US", "US", "US", "US", "US", "US", "US", "US", "US"~
## $ Lat               <dbl> 32.53953, 32.53953, 32.53953, 32.53953, 32.53953, 32~
## $ Long_             <dbl> -86.64408, -86.64408, -86.64408, -86.64408, -86.6440~
## $ Combined_Key      <chr> "Autauga, Alabama, US", "Autauga, Alabama, US", "Aut~
## $ Date              <chr> "1/22/20", "1/23/20", "1/24/20", "1/25/20", "1/26/20~
## $ `Total Confirmed` <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
```

```
    US_deaths_longer_tbl <- US_deaths_stripped_tbl %>%
        pivot_longer(
            cols = `1/22/20`: `8/5/22`,
            names_to = "Date",
            values_to = "Total Deaths"
        )
```

```
    US_deaths_longer_tbl %>% glimpse()
```

```
## Rows: 3,098,034
## Columns: 10
## $ UID            <dbl> 84001001, 84001001, 84001001, 84001001, 84001001, 84001~
## $ Admin2         <chr> "Autauga", "Autauga", "Autauga", "Autauga", "Autauga", ~
## $ Province_State <chr> "Alabama", "Alabama", "Alabama", "Alabama", "Alabama", ~
## $ Country_Region <chr> "US", "US", "US", "US", "US", "US", "US", "US", "US", "~
## $ Lat            <dbl> 32.53953, 32.53953, 32.53953, 32.53953, 32.53953, 32.53~
## $ Long_          <dbl> -86.64408, -86.64408, -86.64408, -86.64408, -86.64408, ~
## $ Combined_Key   <chr> "Autauga, Alabama, US", "Autauga, Alabama, US", "Autaug~
## $ Population     <dbl> 55869, 55869, 55869, 55869, 55869, 55869, 55869, 55869,~
## $ Date           <chr> "1/22/20", "1/23/20", "1/24/20", "1/25/20", "1/26/20", ~
## $ `Total Deaths` <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
```

```
# lubridate

    US_confirmed_longer_tbl$Date <- mdy(US_confirmed_longer_tbl$Date)
    US_confirmed_longer_tbl %>% glimpse()
```

```
## Rows: 3,098,034
## Columns: 9
## $ UID              <dbl> 84001001, 84001001, 84001001, 84001001, 84001001, 84~
## $ Admin2           <chr> "Autauga", "Autauga", "Autauga", "Autauga", "Autauga~
## $ Province_State   <chr> "Alabama", "Alabama", "Alabama", "Alabama", "Alabama~
## $ Country_Region   <chr> "US", "US", "US", "US", "US", "US", "US", "US", "US"~
## $ Lat              <dbl> 32.53953, 32.53953, 32.53953, 32.53953, 32.53953, 32~
## $ Long_            <dbl> -86.64408, -86.64408, -86.64408, -86.64408, -86.6440~
## $ Combined_Key     <chr> "Autauga, Alabama, US", "Autauga, Alabama, US", "Aut~
## $ Date             <date> 2020-01-22, 2020-01-23, 2020-01-24, 2020-01-25, 202~
## $ `Total Confirmed` <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
```

```
    US_deaths_longer_tbl$Date <- mdy(US_deaths_longer_tbl$Date)
    US_deaths_longer_tbl %>% glimpse()
```

```
## Rows: 3,098,034
## Columns: 10
## $ UID            <dbl> 84001001, 84001001, 84001001, 84001001, 84001001, 84001~
## $ Admin2         <chr> "Autauga", "Autauga", "Autauga", "Autauga", "Autauga", ~
## $ Province_State <chr> "Alabama", "Alabama", "Alabama", "Alabama", "Alabama", ~
## $ Country_Region <chr> "US", "US", "US", "US", "US", "US", "US", "US", "US", "~
## $ Lat            <dbl> 32.53953, 32.53953, 32.53953, 32.53953, 32.53953, 32.53~
## $ Long_          <dbl> -86.64408, -86.64408, -86.64408, -86.64408, -86.64408, ~
## $ Combined_Key   <chr> "Autauga, Alabama, US", "Autauga, Alabama, US", "Autaug~
## $ Population     <dbl> 55869, 55869, 55869, 55869, 55869, 55869, 55869, 55869,~
## $ Date           <date> 2020-01-22, 2020-01-23, 2020-01-24, 2020-01-25, 2020-0~
## $ `Total Deaths` <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
```

```
# groupby and summarize

    US_confirmed_y_tbl <- US_confirmed_longer_tbl %>%
```

```
        mutate(Year = year(Date)) %>%
        mutate(Date = as.character(Date)) %>%
        filter(Date == "2020-12-31" | Date == "2021-12-31" | Date == "2022-08-05") %>%

        group_by(UID, Admin2, Province_State, Year) %>%

        summarize(Total_Confirmed = sum(`Total Confirmed`)) %>%
        ungroup()
```

```
## `summarise()` has grouped output by 'UID', 'Admin2', 'Province_State'. You can override using the `.g
```

```
    US_deaths_y_tbl <- US_deaths_longer_tbl%>%
        mutate(Year = year(Date)) %>%
        mutate(Date = as.character(Date)) %>%
        filter(Date == "2020-12-31" | Date == "2021-12-31" | Date == "2022-08-05") %>%

        group_by(UID, Admin2, Province_State, Population, Year) %>%
        summarize(`Total Deaths` = sum(`Total Deaths`)) %>%
        ungroup()
```

```
## `summarise()` has grouped output by 'UID', 'Admin2', 'Province_State', 'Population'. You can overrid
```

```
        US_deaths_y_m_tbl <- US_deaths_longer_tbl %>%
        mutate(Year = year(Date)) %>%
        mutate(Month = month(Date, label = TRUE)) %>%
        mutate(Year_Month = ceiling_date(x = Date, unit = "month" ) - 1) %>%
        filter(Date == Year_Month) %>%


        group_by(Province_State,Year_Month) %>%
        summarize(`Total Deaths` = sum(`Total Deaths`)) %>%
        ungroup()
```

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

```
# summarize total confirmed and deaths by state and year
    US_confirmed_y_tbl <- US_confirmed_y_tbl %>%
        group_by(Province_State, Year) %>%
        summarize(Total_Confirmed = sum(Total_Confirmed)) %>%
        ungroup()
```

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

```
    US_deaths_y_tbl <- US_deaths_y_tbl %>%
        group_by(Province_State, Year) %>%
        summarize(`Total Deaths` = sum(`Total Deaths`)) %>%
        ungroup()
```

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

```r
# Calculate cases and deaths per year
    US_confirmed_y_tbl <- US_confirmed_y_tbl %>%
        group_by(Province_State) %>%
        mutate(Total_Confirmed_Lag = lag(Total_Confirmed, n = 1)) %>%
        mutate(Total_Confirmed_Lag = case_when(
            is.na(Total_Confirmed_Lag) ~ 0,
            TRUE ~ Total_Confirmed_Lag)) %>%
        mutate(diff = Total_Confirmed - Total_Confirmed_Lag) %>%
        ungroup()

    US_deaths_y_tbl <- US_deaths_y_tbl %>%
        group_by(Province_State) %>%
        mutate(Total_Deaths_Lag = lag(`Total Deaths`, n = 1)) %>%
        mutate(Total_Deaths_Lag = case_when(
            is.na(Total_Deaths_Lag) ~ 0,
            TRUE ~ Total_Deaths_Lag)) %>%
        mutate(diff = `Total Deaths` - Total_Deaths_Lag) %>%
        ungroup()

 # add unique identifier

    US_confirmed_y_tbl <- tibble::rowid_to_column(US_confirmed_y_tbl, "ID")

    US_confirmed_y_tbl %>% glimpse()
```

```
## Rows: 174
## Columns: 6
## $ ID                  <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15,~
## $ Province_State      <chr> "Alabama", "Alabama", "Alabama", "Alaska", "Alaska~
## $ Year                <dbl> 2020, 2021, 2022, 2020, 2021, 2022, 2020, 2021, 20~
## $ Total_Confirmed     <dbl> 361226, 896614, 1436458, 47014, 157169, 286803, 0,~
## $ Total_Confirmed_Lag <dbl> 0, 361226, 896614, 0, 47014, 157169, 0, 0, 11, 0, ~
## $ diff                <dbl> 361226, 535388, 539844, 47014, 110155, 129634, 0, ~
```

```r
    US_deaths_y_tbl <- tibble::rowid_to_column(US_deaths_y_tbl, "ID")

    US_deaths_y_tbl %>% glimpse()
```

```
## Rows: 174
## Columns: 6
## $ ID               <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16~
## $ Province_State   <chr> "Alabama", "Alabama", "Alabama", "Alaska", "Alaska", ~
## $ Year             <dbl> 2020, 2021, 2022, 2020, 2021, 2022, 2020, 2021, 2022,~
## $ `Total Deaths`   <dbl> 4827, 16455, 19974, 206, 978, 1309, 0, 0, 33, 8864, 2~
## $ Total_Deaths_Lag <dbl> 0, 4827, 16455, 0, 206, 978, 0, 0, 0, 0, 8864, 24229,~
## $ diff             <dbl> 4827, 11628, 3519, 206, 772, 331, 0, 0, 33, 8864, 153~
```

```r
    US_combined_y_tbl <- US_confirmed_y_tbl %>%
        left_join(y = US_deaths_y_tbl, by = "ID")

# remove and rename columns
```

```
    US_combined_y_tbl <- US_combined_y_tbl %>%
        select(-Province_State.y, -Year.y) %>%
        rename(Year = Year.x, Cases_Per_Year = diff.x, Deaths_Per_Year = diff.y, Province_State = Provi
```

Analysis and Visualization

What are the total confirmed COVID-19 cases and deaths by state?

The top 6 states with the most COVID-19 cases are: 1. California 2. Texas 3. Florida 4. New York 5. Illinois 6. Pennsylvania

The top 6 states with the most COVID-19 deaths are: 1. California 2. Texas 3. Florida 4. New York 5. Pennsylvania 6. Georgia

Five out six states are on both lists, which suggests that states a with high number of COVID-19 cases also have a high number of deaths. Illinois didn't show up in the top 6 states with the most COVID-19 deaths, which suggests the state's COVID-19 death rate is lower than it's case rate. Unlike Illinois,Georgia is on the top six deaths list, which suggests COVID-19 death rate is higher than it's case rate.

```
confirmed_by_state <- US_combined_y_tbl %>%
        select(Province_State, Cases_Per_Year) %>%
        filter(Province_State != "Diamond Princess", Province_State != "Grand Princess") %>%
        group_by(Province_State) %>%
        summarize(Total_Cases = sum(Cases_Per_Year)) %>%
        arrange(desc(Total_Cases)) %>%
        ungroup()

    confirmed_by_state
```

```
## # A tibble: 56 x 2
##    Province_State Total_Cases
##    <chr>                <dbl>
##  1 California        10810291
##  2 Texas              7611709
##  3 Florida            6855234
##  4 New York           5861208
##  5 Illinois           3594415
##  6 Pennsylvania       3105341
##  7 North Carolina     3016395
##  8 Ohio               2976027
##  9 Georgia            2778580
## 10 Michigan           2692485
## # ... with 46 more rows
```

```
    deaths_by_state <- US_combined_y_tbl %>%
        select(Province_State, Deaths_Per_Year) %>%
        filter(Province_State != "Diamond Princess", Province_State != "Grand Princess") %>% #remove sh
        group_by(Province_State) %>%
        summarize(`Total Deaths`= sum(Deaths_Per_Year)) %>%
        arrange(desc(`Total Deaths`)) %>%
        ungroup()
```

```
    deaths_by_state
```

```
## # A tibble: 56 x 2
##    Province_State `Total Deaths`
##    <chr>                   <dbl>
##  1 California              93816
##  2 Texas                   89463
##  3 Florida                 78047
##  4 New York                70490
##  5 Pennsylvania            46261
##  6 Georgia                 39173
##  7 Ohio                    39133
##  8 Illinois                38966
##  9 Michigan                37534
## 10 New Jersey              34326
## # ... with 46 more rows
```

```
confirmed_by_state %>% head()
```

```
## # A tibble: 6 x 2
##   Province_State Total_Cases
##   <chr>                <dbl>
## 1 California        10810291
## 2 Texas              7611709
## 3 Florida            6855234
## 4 New York           5861208
## 5 Illinois           3594415
## 6 Pennsylvania       3105341
```

```
deaths_by_state %>% head()
```

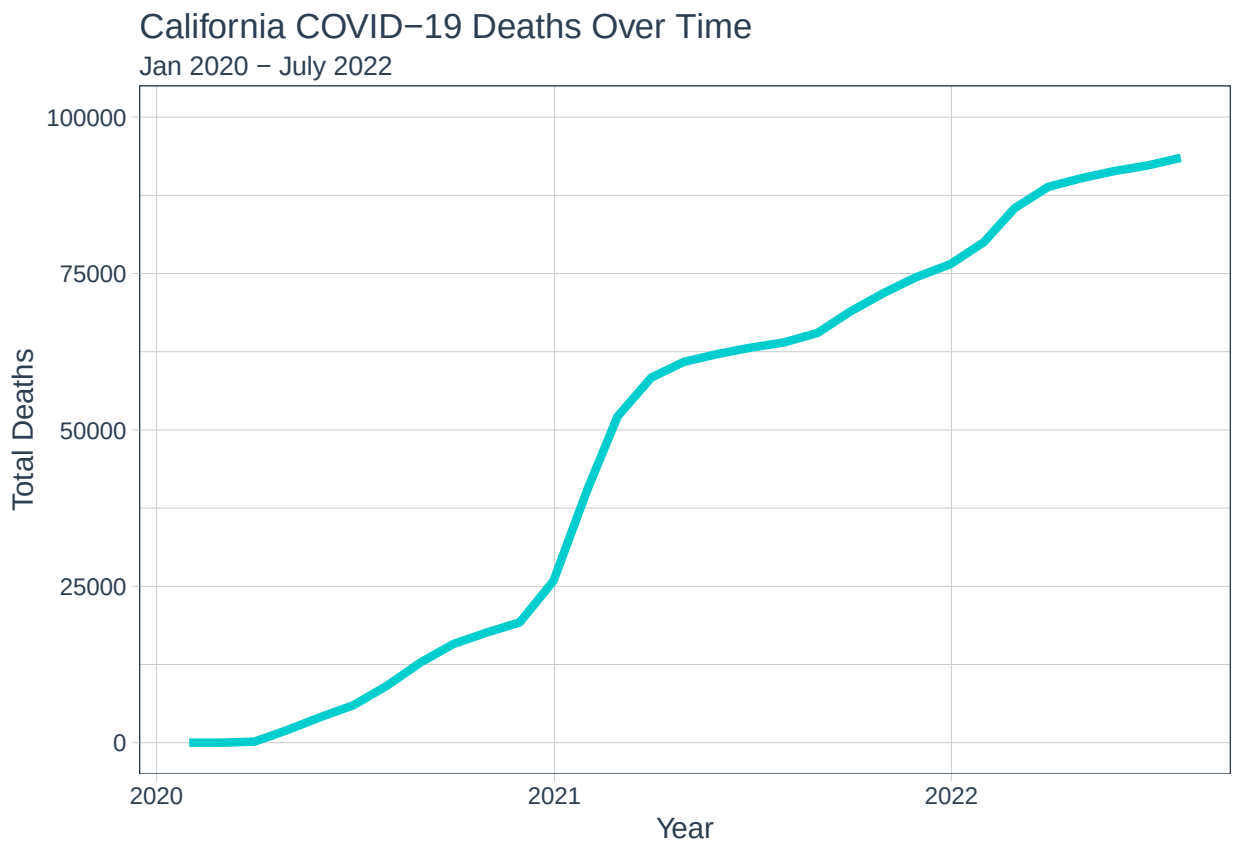
```
## # A tibble: 6 x 2
##   Province_State `Total Deaths`
##   <chr>                   <dbl>
## 1 California              93816
## 2 Texas                   89463
## 3 Florida                 78047
## 4 New York                70490
## 5 Pennsylvania            46261
## 6 Georgia                 39173
```

2. What does California's COVID-19 deaths look like over time?

Because I live in California, I decided to graph California's COVID-19 deaths over time. COVID-19 deaths are only through July 2022 due to not having an entire month's worth of August COVID-19 data at the time of this analysis.

By the end of 2020, California had about 25 thousand COVID-19 deaths. However, that number quickly doubled. By the end of the first quarter of 2021, California had surpassed 50 thousand COVID-19 deaths. Deaths slowed down in the summer of 2021 and reached 75 thousand by the end of 2021. As of July 2022, California has had approximately 93 thousand COVID-19 deaths.

```r
# Plot California deaths over time


    CA_deaths_plot <- US_deaths_y_m_tbl %>%

        filter(Province_State == "California") %>%
        select(Year_Month, `Total Deaths`) %>%

        ggplot(aes(x = Year_Month, y = `Total Deaths`)) +
        geom_line(color = "cyan3", size = 1.5) +
        labs(
            title = "California COVID-19 Deaths Over Time",
            subtitle = "Jan 2020 - July 2022",
            x = "Year",
        ) +
        ylim(0, 100000) +
        theme_tq()

    CA_deaths_plot
```



3. Which states had the smallest and largest percent increase in COVID-19 deaths compared to 2020?

The top 6 states with the largest percent increase in COVID-19 deaths compared to 2020 are: 1. Northern Mariana Islands 2. Maine 3. Oklahoma 4. Alaska 5. Kentucky 6. Oregon

The top 6 states with the smallest percent increase in COVID-19 deaths compared to 2020 are: 1. Rhode Island 2. Connecticut 3. New York 4. New Jersey 5. North Dakota 6. District of Columbia

It's worth noting that the percent changes are relative. Relative changes on small numbers can appear to be more significant than they are. For example, Northern Mariana Islands had 2 COVID-19 deaths reported in 2020 and 38 reported in 2022, resulting in an 1,800 percent increase. However, the absolute number of COVID-19 deaths in Northern Mariana Islands is extremely small.

```
# Calculate percent difference
    state_pct_diff_tbl <- US_combined_y_tbl %>%
        select(Province_State, Year, `Total Deaths`) %>%
        group_by(Province_State) %>%
        mutate(year_2020 = first(`Total Deaths`)) %>%
        mutate(diff = `Total Deaths` - year_2020) %>%
        mutate(pct_diff = diff / year_2020) %>% #View()
        mutate(pct_diff = case_when(
            pct_diff == "NaN" ~ 0,
            pct_diff == "Inf" ~ 1,
            TRUE ~ pct_diff)) %>%
        mutate(pct_diff = round(pct_diff, 2)) %>%
        mutate(pct_diff_chr = scales::percent(pct_diff)) %>% #I need to remove this field
        ungroup()

    state_pct_diff_tbl %>% glimpse()
```

```
## Rows: 174
## Columns: 7
## $ Province_State <chr> "Alabama", "Alabama", "Alabama", "Alaska", "Alaska", "A~
## $ Year           <dbl> 2020, 2021, 2022, 2020, 2021, 2022, 2020, 2021, 2022, 2~
## $ `Total Deaths` <dbl> 4827, 16455, 19974, 206, 978, 1309, 0, 0, 33, 8864, 242~
## $ year_2020      <dbl> 4827, 4827, 4827, 206, 206, 206, 0, 0, 0, 8864, 8864, 8~
## $ diff           <dbl> 0, 11628, 15147, 0, 772, 1103, 0, 0, 33, 0, 15365, 2197~
## $ pct_diff       <dbl> 0.00, 2.41, 3.14, 0.00, 3.75, 5.35, 0.00, 0.00, 1.00, 0~
## $ pct_diff_chr   <chr> "0%", "241%", "314%", "0%", "375%", "535%", "0%", "0%",~
```

```
# Show percent difference in deaths compared to 2020
    state_max_pct_increase <- state_pct_diff_tbl %>%
        filter(Year == "2022") %>%
        filter(Province_State != "Diamond Princess", Province_State != "Grand Princess") %>%
        group_by(Province_State) %>%
        summarize(max_increase = max(pct_diff)) %>%
        arrange(desc(max_increase)) %>% glimpse() %>%
        mutate(max_increase = scales::percent(max_increase)) %>%
        ungroup()
```

```
## Rows: 56
## Columns: 2
## $ Province_State <chr> "Northern Mariana Islands", "Maine", "Oklahoma", "Alask~
## $ max_increase   <dbl> 18.00, 6.18, 5.56, 5.35, 5.27, 4.56, 4.53, 4.37, 4.26, ~
```

```
    state_max_pct_increase
```

```
## # A tibble: 56 x 2
##    Province_State           max_increase
##    <chr>                    <chr>
##  1 Northern Mariana Islands 1 800.00%
##  2 Maine                    618.00%
##  3 Oklahoma                 556.00%
##  4 Alaska                   535.00%
##  5 Kentucky                 527.00%
##  6 Oregon                   456.00%
##  7 Hawaii                   453.00%
##  8 West Virginia            437.00%
##  9 Virgin Islands           426.00%
## 10 Vermont                  410.00%
## # ... with 46 more rows
```

```
    largest_pct_increase <- state_max_pct_increase %>% head()
    smallest_pct_increase <- state_max_pct_increase %>% tail()

    largest_pct_increase
```

```
## # A tibble: 6 x 2
##   Province_State           max_increase
##   <chr>                    <chr>
## 1 Northern Mariana Islands 1 800.00%
## 2 Maine                    618.00%
## 3 Oklahoma                 556.00%
## 4 Alaska                   535.00%
## 5 Kentucky                 527.00%
## 6 Oregon                   456.00%
```

```
    smallest_pct_increase
```

```
## # A tibble: 6 x 2
##   Province_State       max_increase
##   <chr>                <chr>
## 1 Rhode Island         88.00%
## 2 Connecticut          86.00%
## 3 New York             86.00%
## 4 New Jersey           80.00%
## 5 North Dakota         79.00%
## 6 District of Columbia 74.00%
```

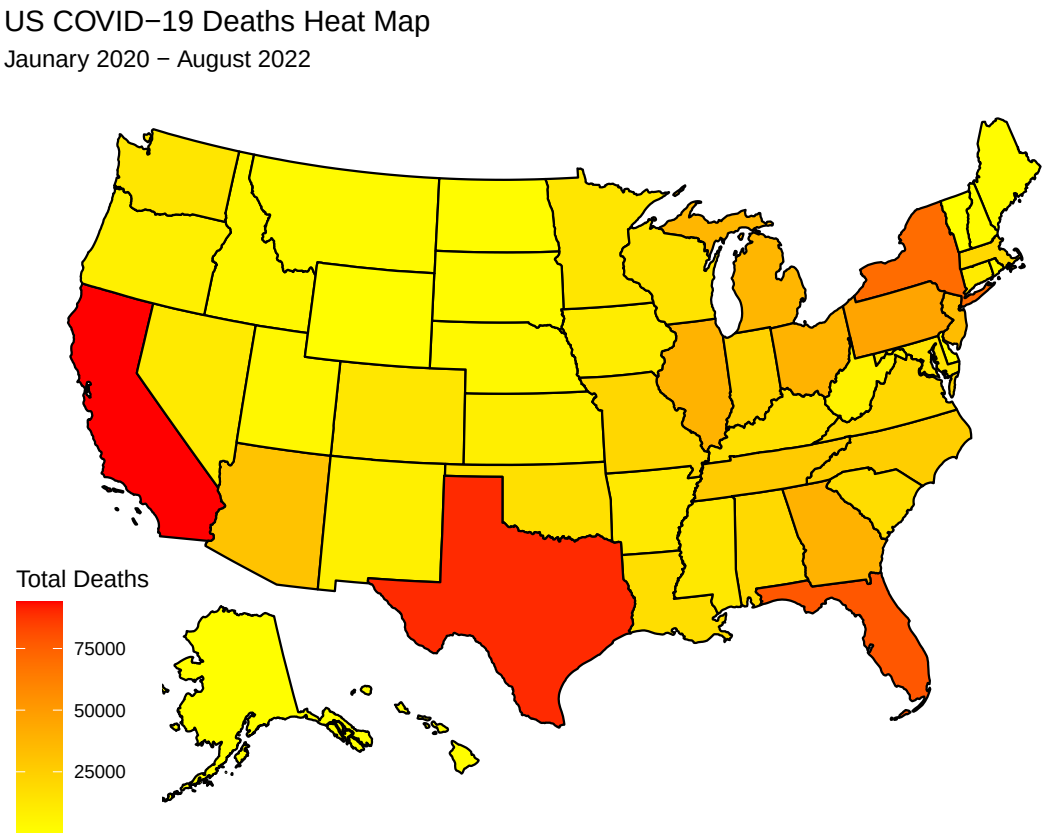
4. What is the COVID-19 deaths impact look like across different geographic areas of the US?

There about four states, California, Texas, New York, Florida in the highest threshold with around 75 thousand deaths or more. While I know there are a lot more people in California and New York, it is still interesting to see the two states in the highest band because I do know they have some of the highest vaccination rates. It would be worth exploring when most of the deaths in California and New York happened. For example, were there were a lot of deaths early on before vaccines were available.

```r
#5.4 Create a map

    # reorder tibble
    deaths_by_state <- deaths_by_state %>%
        arrange(Province_State)

    # create state codes tbl
    state_codes_tbl <- readr::read_delim("state_codes.csv", delim = "|") # I created this CSV file
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   State = col_character(),
##   state = col_character()
## )
```

```r
    state_codes_tbl <- state_codes_tbl %>%
        rename(Province_State = State)

     # join tables
    deaths_by_state <- deaths_by_state %>%
        left_join(y = state_codes_tbl, by ="Province_State" ) %>%
        rename(values = `Total Deaths`)

    deaths_by_state_map_tbl <- deaths_by_state %>%
        select(-Province_State)

    density_map_tbl <- map_with_data(data = deaths_by_state_map_tbl, values = "values")
    density_map_tbl <- density_map_tbl %>%
        select(state, values)

    # Plot map
    plot_usmap(regions = "states", data = density_map_tbl, values = "values") +
    labs(title = "US COVID-19 Deaths Heat Map",
        subtitle = "Jaunary 2020 - August 2022") +
    theme(panel.background = element_blank()) +
    scale_fill_continuous(low = "yellow", high = "red", name = "Total Deaths")
```



5. Is there a relationship between the number of COVID-19 cases and deaths?

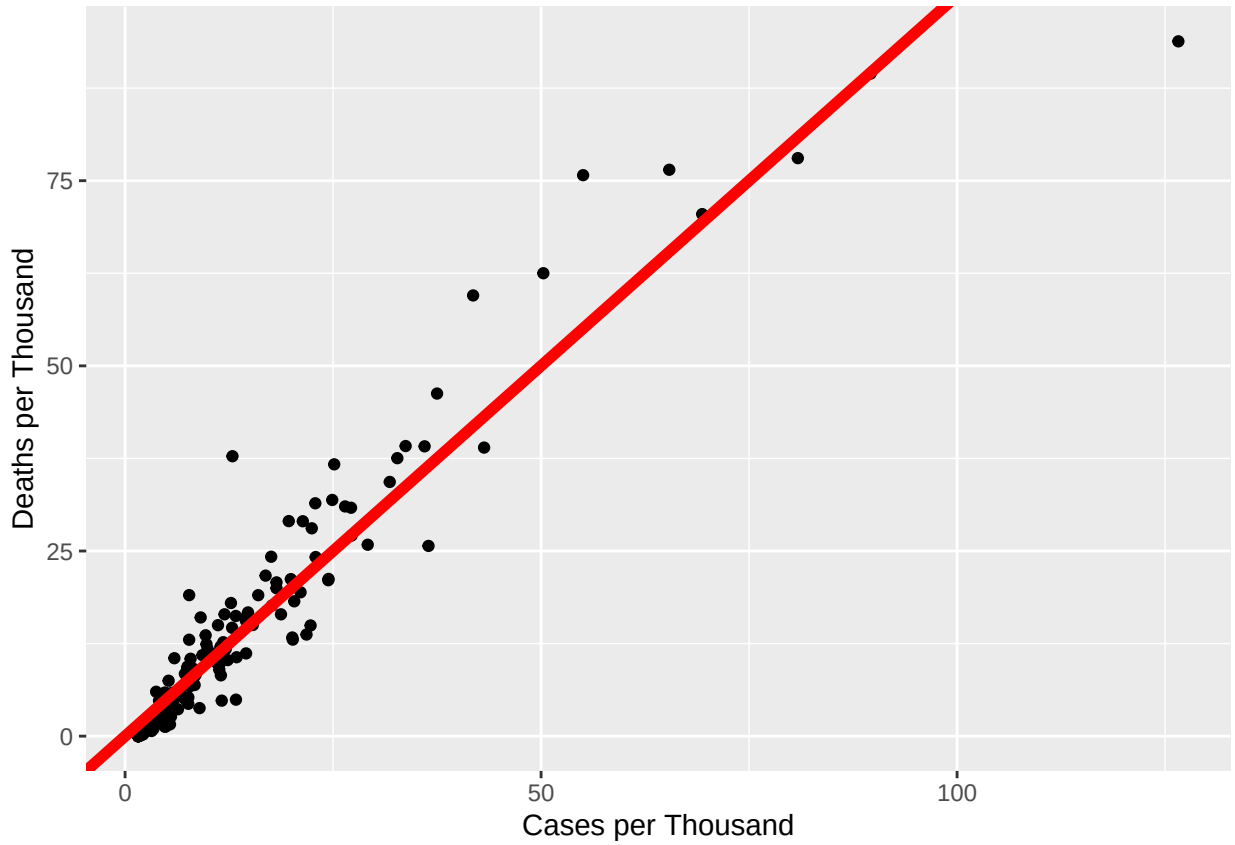
It appears that there is a strong relationship between the number of COVID-19 cases and deaths. Additionally, the goodness of fit measures for linear regression model, r squared, has a value of 0.91.Meaning that the linear regression model explains the observed data well. Lastly, the p-value is significantly small at 2.2e-16, which suggests that the predictor variable, number of COVID-19 cases are associated with the response variable, number of COVID-19 deaths

```
# update combined tbl with cases per thousand and deaths per thousand columns

   US_combined_y_tbl <-  US_combined_y_tbl %>%
       mutate(cases_per_thou = Total_Confirmed / 1000) %>%
       mutate(deaths_per_thou = `Total Deaths` / 1000)


   # estimate linear regression
   my_mod <- lm(deaths_per_thou ~ cases_per_thou, US_combined_y_tbl)
   summary(my_mod)
```

```
##
## Call:
## lm(formula = deaths_per_thou ~ cases_per_thou, data = US_combined_y_tbl)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -32.782  -1.797  -1.191   1.278  24.894
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.5861049  0.4731979   3.352 0.000987 ***
## cases_per_thou 0.0115641  0.0002742  42.174  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.186 on 172 degrees of freedom
## Multiple R-squared:  0.9118, Adjusted R-squared:  0.9113
## F-statistic:  1779 on 1 and 172 DF,  p-value: < 2.2e-16
```

```
    # create data for ggplot
    data_mod_tbl <- tibble(predicted = predict(my_mod),
                           observed = US_combined_y_tbl$deaths_per_thou)

    # create plot
    ggplot(data_mod_tbl,
           aes(x = predicted,
               y = observed)) +
        geom_point() +
        geom_abline(intercept = 0,
                    slope = 1,
                    color = "red",
                    size = 2) +
        labs(
            x = "Cases per Thousand",
            y = "Deaths per Thousand"
        )
```




```
        theme_tq()
```

```
## List of 93
##  $ line                        :List of 6
##   ..$ colour       : chr "#2c3e50"
##   ..$ size         : num 0.5
##   ..$ linetype     : num 1
##   ..$ lineend      : chr "butt"
##   ..$ arrow        : logi FALSE
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_line" "element"
##  $ rect                        :List of 5
##   ..$ fill         : chr "#FFFFFF"
##   ..$ colour       : chr "#2c3e50"
##   ..$ size         : num 0.5
##   ..$ linetype     : num 1
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ text                        :List of 11
##   ..$ family       : chr ""
##   ..$ face         : chr "plain"
##   ..$ colour       : chr "#2c3e50"
##   ..$ size         : num 11
##   ..$ hjust        : num 0.5
##   ..$ vjust        : num 0.5
```

```
##   ..$ angle        : num 0
##   ..$ lineheight   : num 0.9
##   ..$ margin       : 'margin' num [1:4] 0points 0points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : logi FALSE
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ title                    : NULL
##  $ aspect.ratio             : NULL
##  $ axis.title               :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : 'rel' num 1
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.title.x             :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 1
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 2.75points 0points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.title.x.top         :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 0
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 2.75points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.title.x.bottom      : NULL
##  $ axis.title.y             :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
```

```
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 1
##   ..$ angle        : num 90
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 2.75points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.title.y.left         : NULL
##  $ axis.title.y.right        :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 0
##   ..$ angle        : num -90
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 0points 2.75points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text                 :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : 'rel' num 0.8
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.x               :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 1
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 2.2points 0points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.x.top           :List of 11
```

```
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 0
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 2.2points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.x.bottom        : NULL
##  $ axis.text.y               :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : num 1
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 2.2points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.y.left          : NULL
##  $ axis.text.y.right         :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : num 0
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 0points 2.2points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.ticks                :List of 6
##   ..$ colour       : chr "grey80"
##   ..$ size         : 'rel' num 0.333
##   ..$ linetype     : NULL
##   ..$ lineend      : NULL
##   ..$ arrow        : logi FALSE
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_line" "element"
##  $ axis.ticks.x              : NULL
##  $ axis.ticks.x.top          : NULL
##  $ axis.ticks.x.bottom       : NULL
```

```
##  $ axis.ticks.y              : NULL
##  $ axis.ticks.y.left         : NULL
##  $ axis.ticks.y.right        : NULL
##  $ axis.ticks.length         : 'simpleUnit' num 2.75points
##   ..- attr(*, "unit")= int 8
##  $ axis.ticks.length.x       : NULL
##  $ axis.ticks.length.x.top   : NULL
##  $ axis.ticks.length.x.bottom: NULL
##  $ axis.ticks.length.y       : NULL
##  $ axis.ticks.length.y.left  : NULL
##  $ axis.ticks.length.y.right : NULL
##  $ axis.line                 : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
##  $ axis.line.x               : NULL
##  $ axis.line.x.top           : NULL
##  $ axis.line.x.bottom        : NULL
##  $ axis.line.y               : NULL
##  $ axis.line.y.left          : NULL
##  $ axis.line.y.right         : NULL
##  $ legend.background         :List of 5
##   ..$ fill         : NULL
##   ..$ colour       : logi NA
##   ..$ size         : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ legend.margin             : 'margin' num [1:4] 5.5points 5.5points 5.5points 5.5points
##   ..- attr(*, "unit")= int 8
##  $ legend.spacing            : 'simpleUnit' num 11points
##   ..- attr(*, "unit")= int 8
##  $ legend.spacing.x          : NULL
##  $ legend.spacing.y          : NULL
##  $ legend.key                :List of 5
##   ..$ fill         : chr "#FFFFFF"
##   ..$ colour       : logi NA
##   ..$ size         : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ legend.key.size           : 'simpleUnit' num 1.2lines
##   ..- attr(*, "unit")= int 3
##  $ legend.key.height         : NULL
##  $ legend.key.width          : NULL
##  $ legend.text               :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : 'rel' num 0.8
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
```

```
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ legend.text.align         : NULL
##  $ legend.title              :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : num 0
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ legend.title.align        : NULL
##  $ legend.position           : chr "bottom"
##  $ legend.direction          : NULL
##  $ legend.justification      : chr "center"
##  $ legend.box                : NULL
##  $ legend.box.just           : NULL
##  $ legend.box.margin         : 'margin' num [1:4] 0cm 0cm 0cm 0cm
##   ..- attr(*, "unit")= int 1
##  $ legend.box.background     : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
##  $ legend.box.spacing        : 'simpleUnit' num 11points
##   ..- attr(*, "unit")= int 8
##  $ panel.background          :List of 5
##   ..$ fill         : chr "#FFFFFF"
##   ..$ colour       : logi NA
##   ..$ size         : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ panel.border              :List of 5
##   ..$ fill         : logi NA
##   ..$ colour       : chr "#2c3e50"
##   ..$ size         : 'rel' num 0.5
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ panel.spacing             : 'simpleUnit' num 0.75cm
##   ..- attr(*, "unit")= int 1
##  $ panel.spacing.x           : NULL
##  $ panel.spacing.y           : NULL
##  $ panel.grid                :List of 6
##   ..$ colour       : chr "white"
##   ..$ size         : NULL
##   ..$ linetype     : NULL
##   ..$ lineend      : NULL
##   ..$ arrow        : logi FALSE
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_line" "element"
```

```
##  $ panel.grid.major          :List of 6
##   ..$ colour       : chr "grey80"
##   ..$ size         : 'rel' num 0.333
##   ..$ linetype     : NULL
##   ..$ lineend      : NULL
##   ..$ arrow        : logi FALSE
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_line" "element"
##  $ panel.grid.minor          :List of 6
##   ..$ colour       : chr "grey80"
##   ..$ size         : 'rel' num 0.333
##   ..$ linetype     : NULL
##   ..$ lineend      : NULL
##   ..$ arrow        : logi FALSE
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_line" "element"
##  $ panel.grid.major.x        : NULL
##  $ panel.grid.major.y        : NULL
##  $ panel.grid.minor.x        : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
##  $ panel.grid.minor.y        : NULL
##  $ panel.ontop               : logi FALSE
##  $ plot.background           :List of 5
##   ..$ fill         : NULL
##   ..$ colour       : chr "white"
##   ..$ size         : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ plot.title                :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : 'rel' num 1.2
##   ..$ hjust        : num 0
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 4points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ plot.title.position       : chr "panel"
##  $ plot.subtitle             :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : 'rel' num 0.9
##   ..$ hjust        : num 0
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 3points 0points
```

```
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ plot.caption              :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : 'rel' num 0.8
##   ..$ hjust        : num 1
##   ..$ vjust        : num 1
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 5.5points 0points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ plot.caption.position     : chr "panel"
##  $ plot.tag                  :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : 'rel' num 1.2
##   ..$ hjust        : num 0.5
##   ..$ vjust        : num 0.5
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ plot.tag.position         : chr "topleft"
##  $ plot.margin               : 'margin' num [1:4] 5.5points 5.5points 5.5points 5.5points
##   ..- attr(*, "unit")= int 8
##  $ strip.background          :List of 5
##   ..$ fill         : chr "#2c3e50"
##   ..$ colour       : chr "#2c3e50"
##   ..$ size         : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ strip.background.x        : NULL
##  $ strip.background.y        : NULL
##  $ strip.placement           : chr "inside"
##  $ strip.text                :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : chr "#FFFFFF"
##   ..$ size         : 'rel' num 0.8
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
```

```
##   ..$ margin       : 'margin' num [1:4] 5points 0points 5points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ strip.text.x              : NULL
##  $ strip.text.y              :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : num -90
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ strip.switch.pad.grid     : 'simpleUnit' num 2.75points
##   ..- attr(*, "unit")= int 8
##  $ strip.switch.pad.wrap     : 'simpleUnit' num 2.75points
##   ..- attr(*, "unit")= int 8
##  $ strip.text.y.left         :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : num 90
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  - attr(*, "class")= chr [1:2] "theme" "gg"
##  - attr(*, "complete")= logi TRUE
##  - attr(*, "validate")= logi TRUE
```

Conclusion and Bias Identification

It's interesting to see the number of COVID-19 cases and deaths and relationships in the US by state as well as the death trend in California. Additional analysis could be done to look at more granular locational impacts, COVID-19 cases and deaths compared to state populations, and case and deaths rates over time.

A bias that came to mine as I was completing this project was vaccination rates. However, I didn't merge COVID-19 vaccination data to explore this biased thought.