Final_project

2022-04-12

Information

This project contains various data visualizations and statistical observations of Covid and Unemployment during the years 2019, 2020, 2021. It will yield some of the important results about Covid trends.

Libraries

I have installed various different libraries that helped me in making data visualizations

```
library(DBI)
library(tidyverse)
```

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

## v ggplot2 3.3.6     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.8
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1

## Warning: package 'tidyr' was built under R version 4.0.5

## Warning: package 'readr' was built under R version 4.0.5

## Warning: package 'dplyr' was built under R version 4.0.5

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

```
library(here)
```

```
## here() starts at /Users/dwishamehta/Downloads
```

```
library(janitor)
```

```
##
## Attaching package: 'janitor'

## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

```
library(rvest)
```

```
##
## Attaching package: 'rvest'

## The following object is masked from 'package:readr':
##
##     guess_encoding
```

```
library(plotly)
```

```
##
## Attaching package: 'plotly'

## The following object is masked from 'package:ggplot2':
##
##     last_plot

## The following object is masked from 'package:stats':
##
##     filter

## The following object is masked from 'package:graphics':
##
##     layout
```

```
library(readxl)
```

```
## Warning: package 'readxl' was built under R version 4.0.5
```

```
library(lubridate)
```

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

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

```
library(dplyr)
library(ggbeeswarm)
library(RSelenium)
library(jsonlite)
```

```
## Warning: package 'jsonlite' was built under R version 4.0.5

##
## Attaching package: 'jsonlite'
```

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

```
library(ggthemes)
library(readr)
library(RSocrata)
library(RSelenium)
library(robotstxt)
library(readr)
library(base)
library(ggplot2)
library(usmap)
```

```
## Warning: package 'usmap' was built under R version 4.0.5
```

```
library(reshape2)
```

```
##
## Attaching package: 'reshape2'
```

```
## The following object is masked from 'package:tidyr':
##
##     smiths
```

```
library(scales)
```

```
##
## Attaching package: 'scales'
```

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

```
## The following object is masked from 'package:readr':
##
##     col_factor
```

```
library(zoo)
```

```
##
## Attaching package: 'zoo'
```

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

```
library(plotrix)
```

```
##
## Attaching package: 'plotrix'

## The following object is masked from 'package:scales':
##
##     rescale
```

```
library(Rcpp)
```

```
## Warning: package 'Rcpp' was built under R version 4.0.5
```

Data

I have used four datasets for making analysis in this project

Covid data: This dataset contains information about new cases, new cases, total cases for each state in the years from 2019 to 2022. This dataset is obtained from CDC (Center of Disease Control and Prevention)- https://data.cdc.gov/Case-Surveillance/United-States-COVID-19-Cases-and-Deaths-by-State-o/9mfq-cb36/data Read this data directly from the csv file after importing this dataset in this project.

Vaccine data: This dataset contains information about total vaccines distributed and the type of vaccine distributed for each state in the years from 2019 to 2022. This dataset is obtained from CDC (Center of Disease Control and Prevention)- https://data.cdc.gov/Vaccinations/COVID-19-Vaccinations-in-the-United-States-Jurisdi/unsk-b7fc/data Read this data directly from the csv file after importing this dataset in this project.

Unemployment2019_2020 data: This dataset contains information about unemployment rate for the year 2019 and 2020 for each state and also provides information about the change over the year. This dataset is obtained from US Bureau of Labor Statistics- https://www.bls.gov/lau/lastch20.htm Read this data using webscrapping method with the use of selector gadget function.

Unemployment2020_2021 data: This dataset contains information about unemployment rate for the year 2020 and 2021 for each state and also provides information about the change over the year. This dataset is obtained from US Bureau of Labor Statistics- https://www.bls.gov/lau/lastch21.htm Read this data using webscrapping method with the use of selector gadget function.

```
#Read data
covid_data <-read_csv("United_States_COVID-19_Cases_and_Deaths_by_State_over_Time.csv")
```

```
## Rows: 49680 Columns: 15-- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (5): submission_date, state, created_at, consent_cases, consent_deaths
## dbl (10): tot_cases, conf_cases, prob_cases, new_case, pnew_case, tot_death,...
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
vaccine_data <- read_csv("COVID-19_Vaccinations_in_the_United_States_Jurisdiction.csv")
```

```
## Rows: 32408 Columns: 82-- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (2): Date, Location
## dbl (80): MMWR_week, Distributed, Distributed_Janssen, Distributed_Moderna, ...
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
url <- "https://www.bls.gov/lau/lastch20.htm"
robotstxt::paths_allowed(url)
```

```
##  www.bls.gov
```

```
## [1] TRUE
```

```
unemployment2019_2020 <- read_html(url) %>% html_elements("#lastch20") %>% .[[1]]  %>%
  html_table()

url <- "https://www.bls.gov/lau/lastch21.htm"
robotstxt::paths_allowed(url)
```

```
##  www.bls.gov
```

```
## [1] TRUE
```

```
unemployment2020_2021 <- read_html(url) %>% html_elements("#lastch21") %>% .[[1]]  %>%
  html_table()
```

From the covid dataset, useful and interesting variables has been selected for making further analysis. Also, renamed the variable date and changed it to mdy format.

```
#clean covid dataset
covid_data <- covid_data %>%
  select(state, submission_date, tot_cases, new_case, tot_death, new_death) %>%
  rename(date = submission_date) %>%
  mutate(date = mdy(date))
covid_data
```

```
## # A tibble: 49,680 x 6
##    state date       tot_cases new_case tot_death new_death
##    <chr> <date>         <dbl>    <dbl>     <dbl>     <dbl>
##  1 KS    2021-03-11    297229        0      4851         0
##  2 UT    2021-02-12    359641     1060      1785        11
##  3 AR    2020-02-04         0        0         0         0
##  4 MP    2021-12-06      1104        0         5         2
##  5 PW    2021-05-09         0        0         0         0
##  6 UT    2022-01-01    636992        0      3787         0
##  7 HI    2020-06-05       661        8        17         0
##  8 AK    2021-07-27     71521      235       377         0
##  9 HI    2021-10-26     80876       69       883         0
## 10 OK    2021-07-26    475578     1028      7488         8
## # ... with 49,670 more rows
```

From the vaccine dataset, useful and interesting variables has been selected for making further analysis. Also, renamed the variable date and state. For state and date, each dataset had different variable names so I changed it so that it is consistent. Also, changed the variable "date" to mdy format.

```r
#clean vaccine dataset
vaccine_data <- vaccine_data %>%
  select(Location, Date, Distributed, Distributed_Janssen, Distributed_Moderna, Distributed_Pfizer, Adm
  rename(state = Location, date = Date) %>%
   mutate(date = mdy(date))
vaccine_data
```

```
## # A tibble: 32,408 x 11
##    state date       Distributed Distributed_Janssen Distributed_Moderna
##    <chr> <date>           <dbl>               <dbl>               <dbl>
##  1 NJ    2022-04-29    22037155              967300             7931680
##  2 ID    2022-04-29     3406890              157300             1330980
##  3 CA    2022-04-29    90341055             3722700            32441980
##  4 VA2   2022-04-29     8103220              626600             3944180
##  5 AS    2022-04-29      117010                 600               24100
##  6 VA    2022-04-29    19679585              782600             6996500
##  7 KS    2022-04-29     6050215              256300             2322240
##  8 CO    2022-04-29    12511675              492000             4596240
##  9 RI    2022-04-29     2593725               89100              986700
## 10 ND    2022-04-29     1386800               53000              531720
## # ... with 32,398 more rows, and 6 more variables: Distributed_Pfizer <dbl>,
## #   Administered_Dose1_Recip <dbl>, Series_Complete_Yes <dbl>,
## #   Series_Complete_Janssen <dbl>, Series_Complete_Moderna <dbl>,
## #   Series_Complete_Pfizer <dbl>
```

While scrapping the table for the unemployment dataset, the table didn't come out very clean as the column "Over the year" was further bifurcated into two columns "Change" and "Rank". For cleaning this dataset, I did various steps:

1) Deleted the first and second rows as first row had repeated column names and second row had the information for the United States and not the state.
2) Deleted the last row because information in the last row was not quite useful
3) As the dataset had duplicated names for the variable "Over the year", I used make.names which would automatically make own names.
4) Renamed the variables according to my choice
5) The covid dataset and vaccine dataset had state abbreviations for the variable "State" but this dataset contained full names of the state so full names were changed to abbreviations for showing consistency throughout the project.
6) Lastly, after performing these steps, a specific row having NA was deleted

```r
#clean unemployment dataset
unemployment2019_2020<- unemployment2019_2020[-c(1, 2),]
unemployment2019_2020 <- head(unemployment2019_2020, -1)
names(unemployment2019_2020) <- make.names(names(unemployment2019_2020), unique=TRUE)
unemployment2019_2020data <- unemployment2019_2020 %>%
rename(`2019_rate` = `X2019rate`, `2020_rate` = `X2020rate`,`change` = `Over.the.year`, rank = `Over.th
unemployment2019_2020data$state <- state.abb[match(unemployment2019_2020data$state,state.name)]
unemployment2019_2020data<- unemployment2019_2020data[-1,]
unemployment2019_2020data
```

```
## # A tibble: 51 x 5
##    state `2019_rate` `2020_rate` change rank
```

```
##    <chr> <chr>       <chr>       <chr>  <chr>
##  1 NE    3.0         4.1         1.1    1
##  2 SD    2.8         4.3         1.5    2
##  3 UT    2.6         4.7         2.1    3
##  4 WY    3.7         5.8         2.1    3
##  5 ME    2.8         5.0         2.2    5
##  6 MT    3.6         5.8         2.2    5
##  7 KY    4.1         6.4         2.3    7
##  8 ID    3.0         5.5         2.5    8
##  9 IA    2.6         5.1         2.5    8
## 10 MS    5.4         7.9         2.5    8
## # ... with 41 more rows
```

Performed similar cleaning process to that of the above unemployment dataset

```
unemployment2020_2021<- unemployment2020_2021[-c(1, 2),]
unemployment2020_2021 <- head(unemployment2020_2021, -1)
names(unemployment2020_2021) <- make.names(names(unemployment2020_2021), unique=TRUE)
unemployment2020_2021data <- unemployment2020_2021 %>%
rename(`2020_rate` = `X2020rate`, `2021_rate` = `X2021rate`,`change` = `Over.the.year`, rank = `Over.th
unemployment2020_2021data$state <- state.abb[match(unemployment2020_2021data$state,state.name)]
unemployment2020_2021data<- unemployment2020_2021data[-1,]
unemployment2020_2021data
```

```
## # A tibble: 51 x 5
##    state `2020_rate` `2021_rate` change rank
##    <chr> <chr>       <chr>       <chr>  <chr>
##  1 HI    12.0        5.7         -6.3   1
##  2 NV    13.5        7.2         -6.3   1
##  3 MI    10.0        5.9         -4.1   3
##  4 MA    9.4         5.7         -3.7   4
##  5 FL    8.2         4.6         -3.6   5
##  6 IN    7.2         3.6         -3.6   5
##  7 RI    9.2         5.6         -3.6   5
##  8 WA    8.5         5.2         -3.3   8
##  9 LA    8.7         5.5         -3.2   9
## 10 NH    6.7         3.5         -3.2   9
## # ... with 41 more rows
```

Added sql connection here Also, changed variable "date" in covid dataset to as.character

```
con <- DBI::dbConnect(RSQLite::SQLite(), dbname = "Final_project.sqlite")

covid_data$date <- as.character(covid_data$date)
```

While creating a new table, overwrite = T was used in order to prevent getting an error about the existing table when running the program again

```
dbWriteTable(con, "covid_trend", covid_data, overwrite = T)
```

Using sql, I tried getting the month out of the "date" variable for only the year 2020

```
select sum(new_case) as total_cases, sum(new_death) as total_death,
                        strftime('%m', date) as month
                       from covid_trend where date >= '2020-01-01' and date <= '2020-12-31'
                        group by month
```

```
covid_trend_2020 <- covid_2020
```

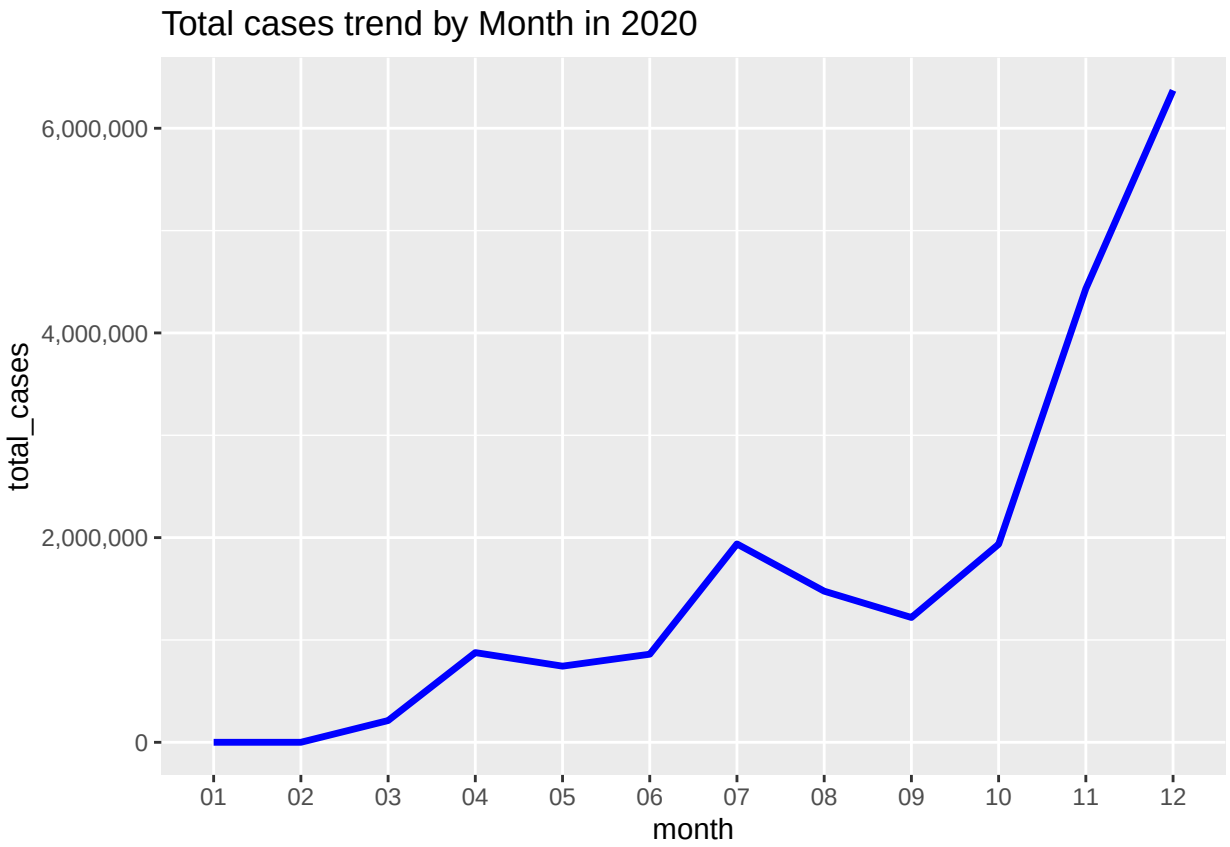
Using sql, I tried getting the month out of the "date" variable for only the year 2021

```
select sum(new_case) as total_cases, sum(new_death) as total_death,
                        strftime('%m', date) as month
                       from covid_trend where date >= '2021-01-01' and date <= '2021-12-31'
                       group by month
```

```
covid_trend_2021 <- covid_2021
```

Created a line plot here for the total cases in each Month in the year 2020

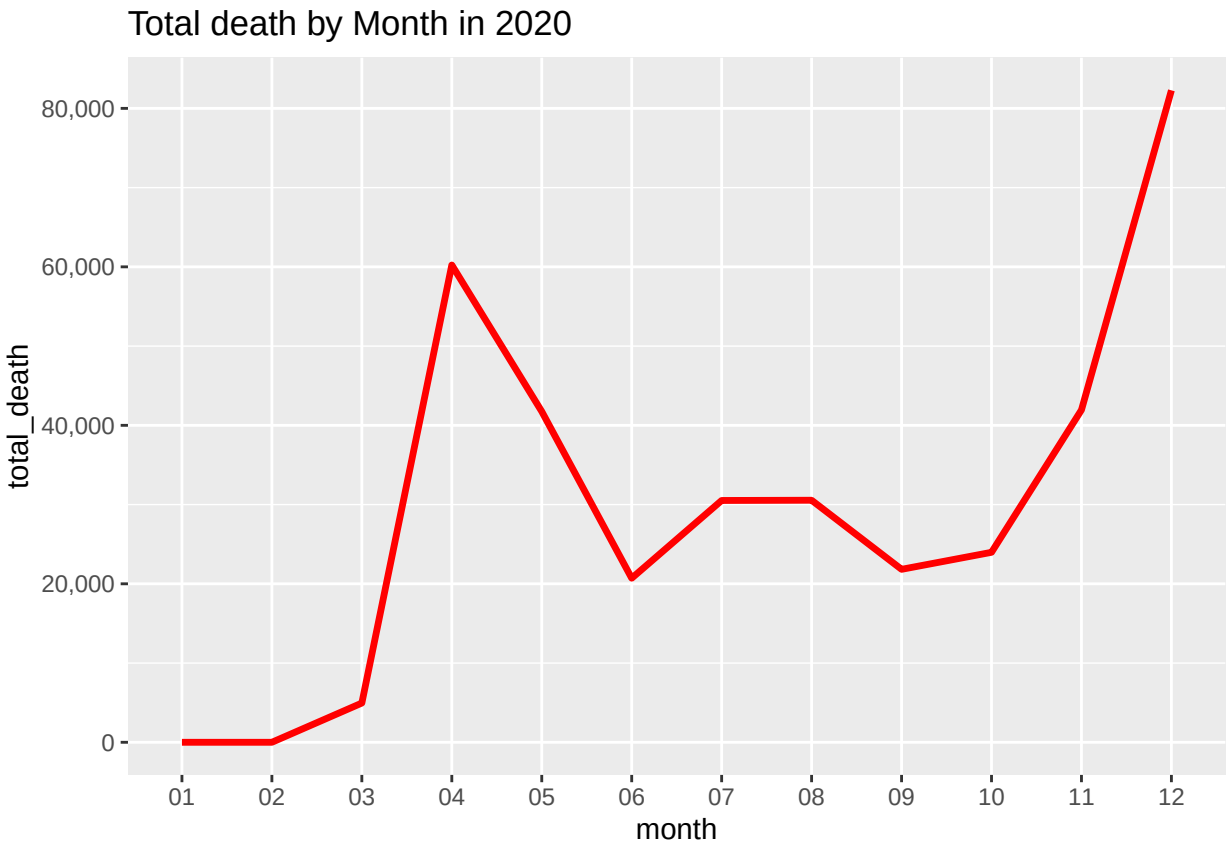
```
covid_trend_2020$month <- as.character.Date(covid_trend_2020$month)

#Total cases trend by Month in 2020

ggplot(covid_trend_2020, aes(month)) +
  geom_line(aes(y = total_cases),group=1, colour = "blue", size=1.2) +
    scale_y_continuous(labels = comma) + ggtitle("Total cases trend by Month in 2020")
```



Created a line plot here for the total deaths in each Month in the year 2020

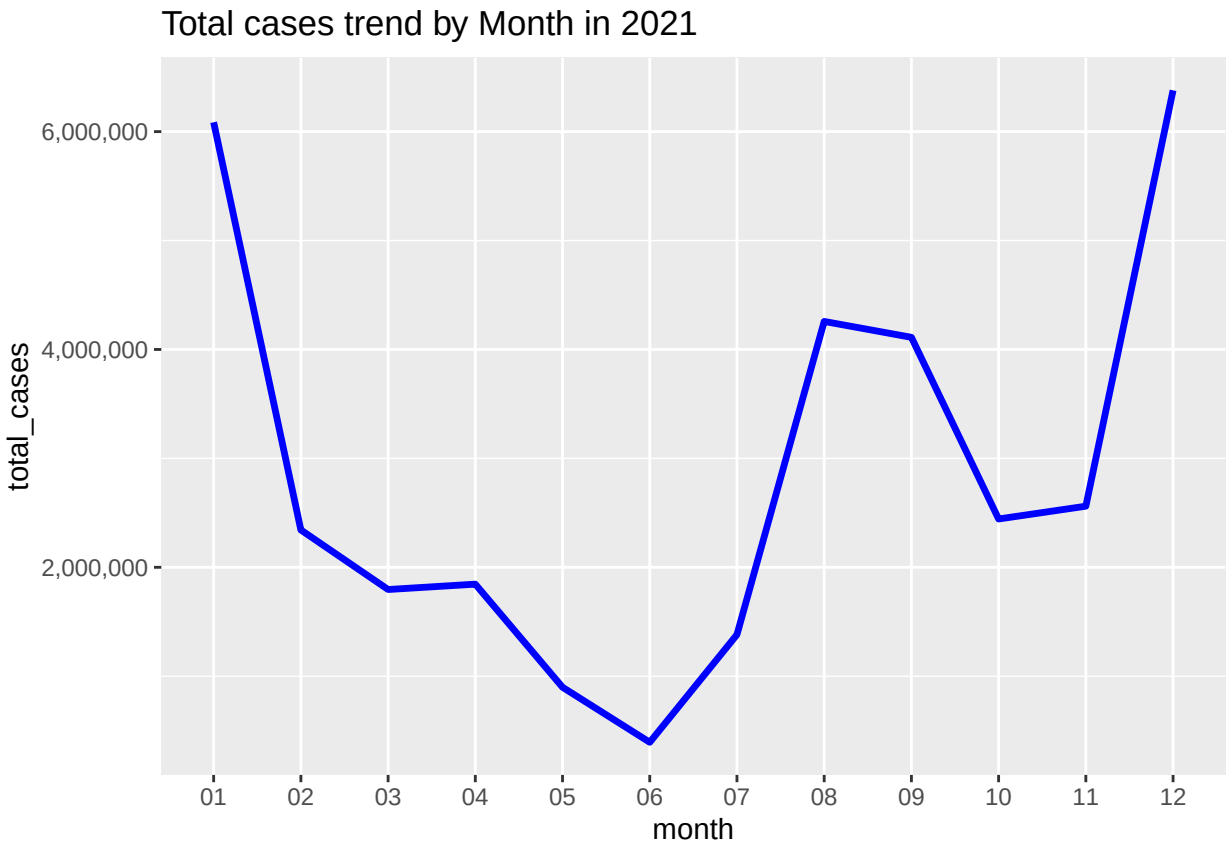
Here, we can see the relationship between total cases and total deaths in the year 2020. It is possible to interpret from the covid cases plot and death plot that when there is an increase in the total cases, total deaths also increase. However, there are some outliers in the graph which can be provide contradictory results for our conclusion The Month of December had the highest cases and deaths.

```
#Total death by Month in 2020

ggplot(covid_trend_2020, aes(month)) +
  geom_line(aes(y = total_death),group=1, colour = "red", size=1.2) +
    scale_y_continuous(labels = comma)+ ggtitle("Total death by Month in 2020")
```



Created a line plot here for the total cases in each Month in the year 2021

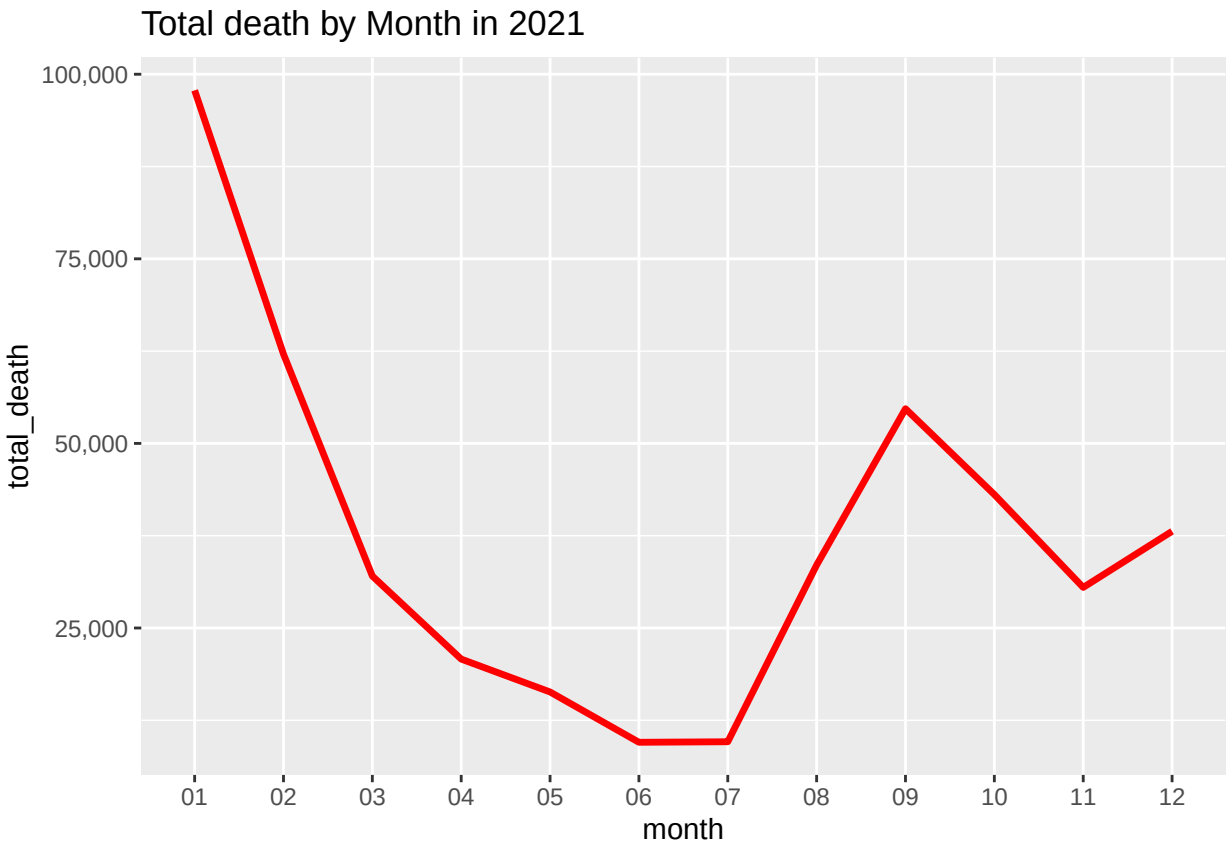
```
#Total cases trend by Month in 2021

ggplot(covid_trend_2021, aes(month)) +
  geom_line(aes(y = total_cases),group=1, colour = "blue", size=1.2) +
    scale_y_continuous(labels = comma)+ ggtitle("Total cases trend by Month in 2021")
```



Created a line plot here for the total deaths in each Month in the year 2021

Here, we can see the relationship between total cases and total deaths in the year 2021. It is possible to interpret from the covid cases plot and death plot that when there is an decrease in the total cases, total deaths also decrease. However, there are some outliers in the graph which can be provide contradictory results for our conclusion The Month of January had the highest cases and deaths.

```
#Total death by Month in 2021

ggplot(covid_trend_2021, aes(month)) +
  geom_line(aes(y = total_death),group=1, colour = "red", size=1.2)  +
    scale_y_continuous(labels = comma)+ ggtitle("Total death by Month in 2021")
```



Using r code, new variable for month was created by selecting month from "date" variable Also, new variable for year was created using ifelse function The variable month was changed from numeric (01, 02) to month names (Jan, Feb)

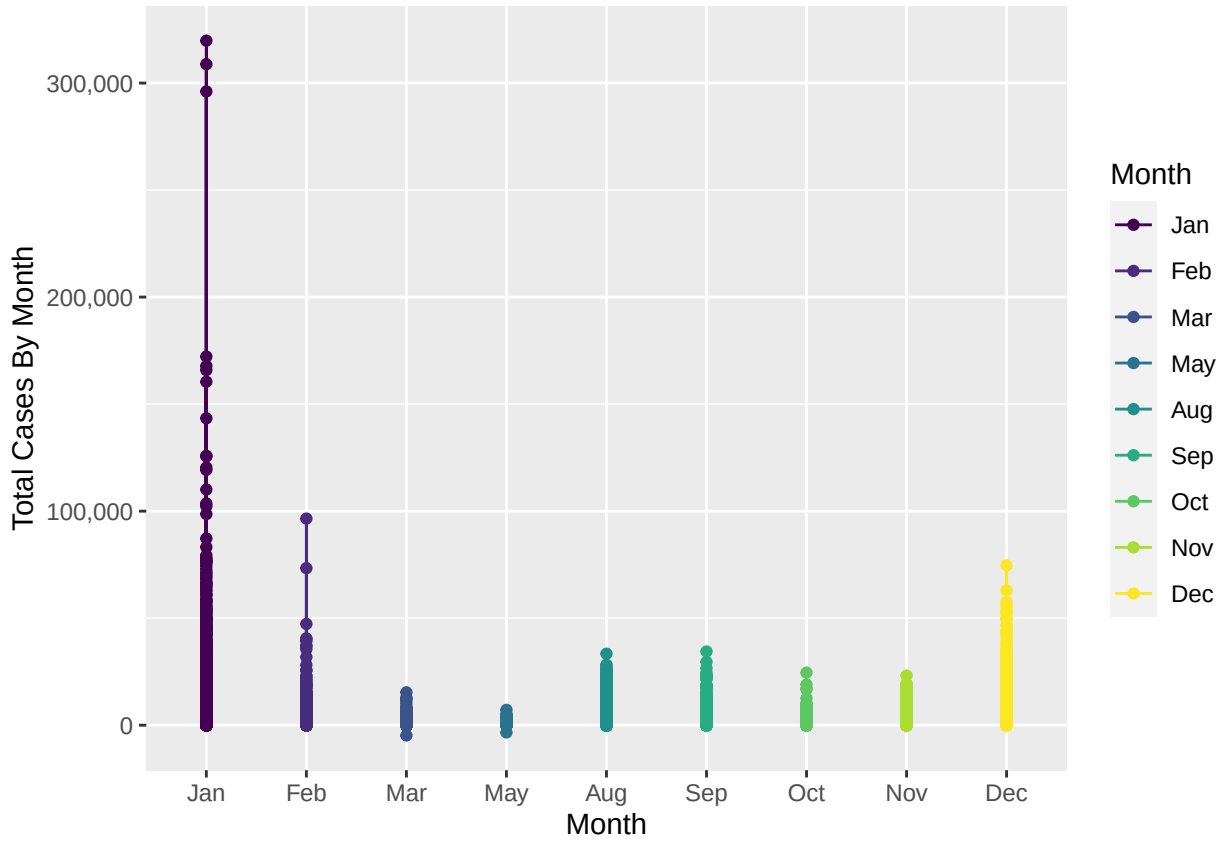
```
covid_data2 <- covid_data%>%
  mutate(month = map_chr(str_split(date, "-"),2), year = ifelse(date >= as.Date("2020-01-01") & date <=
  summarize(state = state, new_case = new_case, Month = month.abb[as.numeric(month)], date = date,year =
covid_data2
```

```
## # A tibble: 49,680 x 5
##    state new_case Month date       year
##    <chr>    <dbl> <chr> <chr>      <chr>
##  1 KS           0 Mar   2021-03-11 2021
##  2 UT        1060 Feb   2021-02-12 2021
##  3 AR           0 Feb   2020-02-04 2020
##  4 MP           0 Dec   2021-12-06 2021
##  5 PW           0 May   2021-05-09 2021
##  6 UT           0 Jan   2022-01-01 2022
##  7 HI           8 Jun   2020-06-05 2020
##  8 AK         235 Jul   2021-07-27 2021
##  9 HI          69 Oct   2021-10-26 2021
## 10 OK        1028 Jul   2021-07-26 2021
## # ... with 49,670 more rows
```

Created a line plot to show the impact of increasing total covid cases on deaths for the years combined 2020 and 2021 for each month

The month of January for the year 2020 as well as 2021 peaked the number of covid cases

```
covid_data2$Month = factor(covid_data2$Month,levels=c("Jan","Feb","Mar", "April","May","June","July","Au
covid_data2 %>%
  filter(!is.na(Month)) %>%
  group_by(Month) %>%
ggplot(aes(Month, new_case, color = Month)) +  geom_point() +  geom_line() + ylab("Total Cases By Month
    scale_y_continuous(labels = comma)
```



Here, the covid data was filtered for the year 2020 and 2021 Performed grouping and summarizing to find the total new_cases and total new_deaths for each state and year

```
covid_data1 <- covid_data %>%
    filter(date >= as.Date("2020-01-01"), date <= as.Date("2021-12-31")) %>%
  summarize(state = state, date = date, new_case = new_case, new_death = new_death,`year` = ifelse(date

covid_data1 <- covid_data1 %>%
  group_by(state, year) %>%
  summarise(total_cases = sum(new_case), total_deaths = sum(new_death))
```

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

```
 covid_data1
```

```
## # A tibble: 120 x 4
## # Groups:   state [60]
##    state year  total_cases total_deaths
##    <chr> <chr>       <dbl>        <dbl>
##  1 AK    2020        45771          281
##  2 AK    2021       105812          585
##  3 AL    2020       362179         7188
##  4 AL    2021       542775         9550
##  5 AR    2020       225138         3676
##  6 AR    2021       341348         5192
##  7 AS    2020            3            0
##  8 AS    2021            8            0
##  9 AZ    2020       520559         8864
## 10 AZ    2021       860929        15365
## # ... with 110 more rows
```

As we know 2020, covid cases peaked in the year 2020 so it was interesting to find out the states that had the highest cases in the year 2020.

Here are the top 10 states having the highest total_cases in the year 2020

```
highestcovid <- covid_data1 %>%
  filter(year == 2020) %>%
  arrange(desc(total_cases)) %>%
  head(10)
highestcovid
```

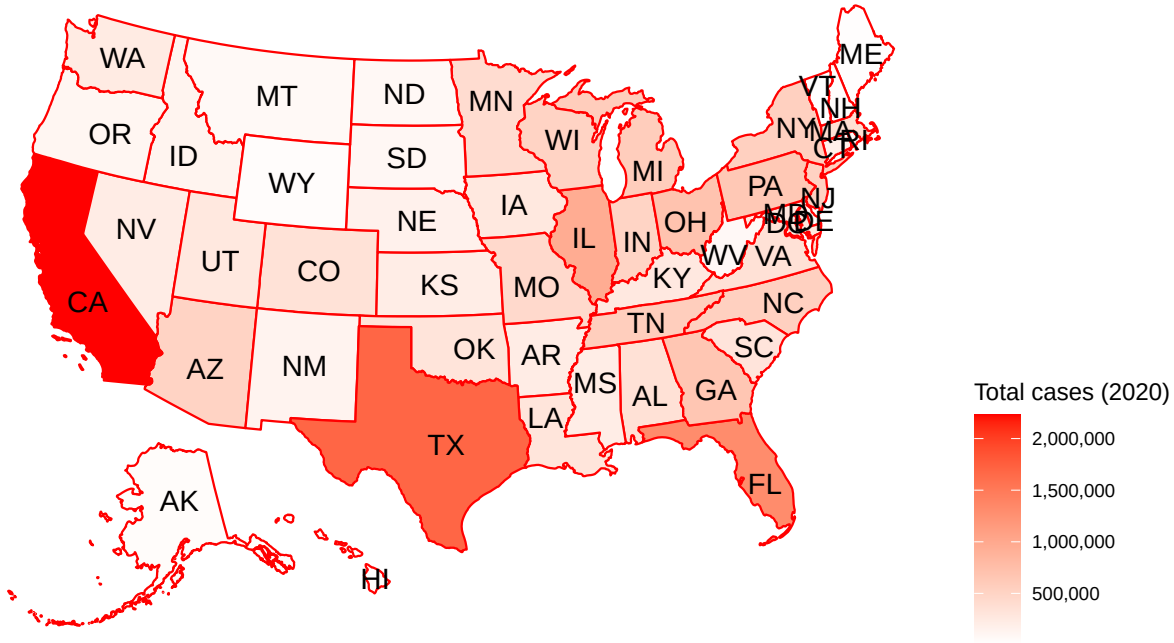
```
## # A tibble: 10 x 4
## # Groups:   state [10]
##    state year  total_cases total_deaths
##    <chr> <chr>       <dbl>        <dbl>
##  1 CA    2020      2231552        25374
##  2 TX    2020      1688697        31282
##  3 FL    2020      1313982        23285
##  4 IL    2020       963389        17978
##  5 OH    2020       700380         8962
##  6 GA    2020       666452        10468
##  7 PA    2020       648569        15978
##  8 MI    2020       589728        13816
##  9 TN    2020       576336         6810
## 10 NY    2020       548154        12566
```

Created an interactive plot for the total cases by state

Here, our statistical observations were supported by the plot California had the highest covid cases and Texas had the second highest covid cases in the year 2020. Also, we can see Florida and Illinois comes after Texas.

```
#COVID CASES IN THE YEAR 2020

covidmap2020 <- covid_data1 %>%
   filter(year == "2020")
plot_usmap(data = covidmap2020, values = "total_cases", color = "red", labels = TRUE) +
  scale_fill_continuous(
```

```
    low = "white", high = "red", name = "Total cases (2020)", label = scales::comma
  ) + theme(legend.position = "right")
```



Here are the top 10 states having the highest total_cases in the year 2021

```
highestcovid <- covid_data1 %>%
  filter(year == 2021) %>%
  arrange(desc(total_cases)) %>%
  head(10)
highestcovid
```

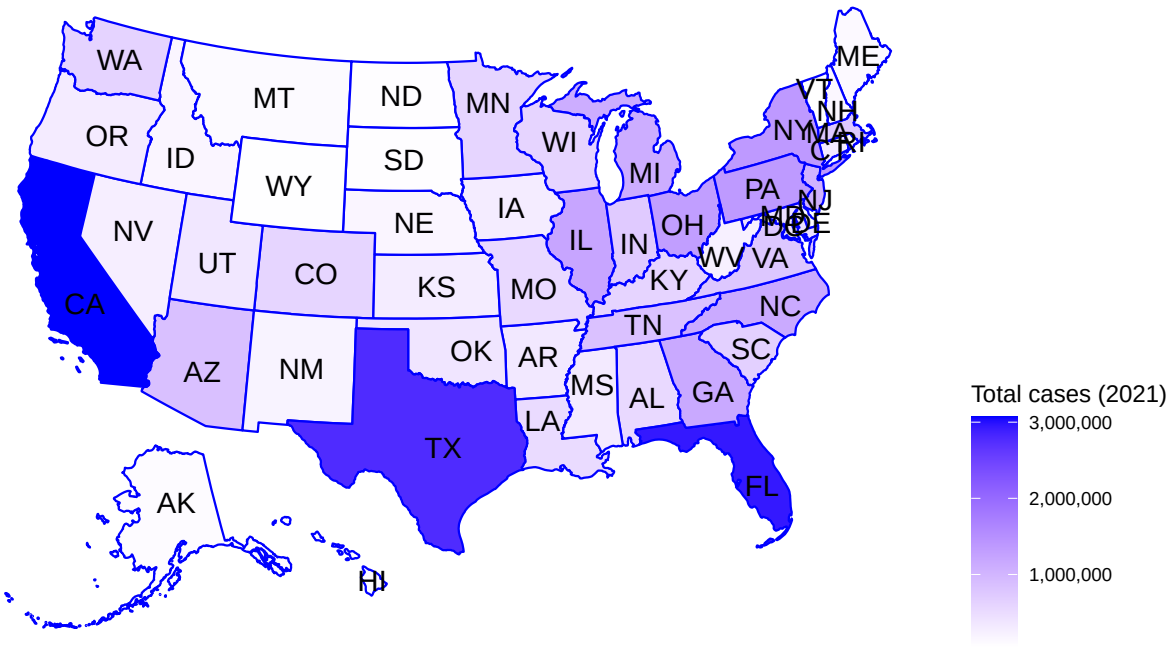
```
## # A tibble: 10 x 4
## # Groups:   state [10]
##    state year  total_cases total_deaths
##    <chr> <chr>       <dbl>        <dbl>
##  1 CA    2021      3068727        50473
##  2 FL    2021      2935045        39824
##  3 TX    2021      2771073        43209
##  4 NY    2021      1376452        11367
##  5 PA    2021      1370900        20727
##  6 OH    2021      1311561        16722
##  7 IL    2021      1217620        13039
##  8 GA    2021      1173427        20509
##  9 NC    2021      1149546        11857
## 10 MI    2021      1120543        11412
```

Created an interactive plot for the total cases by state

Here, our statistical observations were supported by the plot California had the highest covid cases and Florida had the second highest covid cases in the year 2020. Also, we can see Texas and New York comes after Florida.

Here, it is little hard to see if NY was in the top 5 states that had the most covid cases. I tried making some changes in the code but I was unable to increase the size of it.

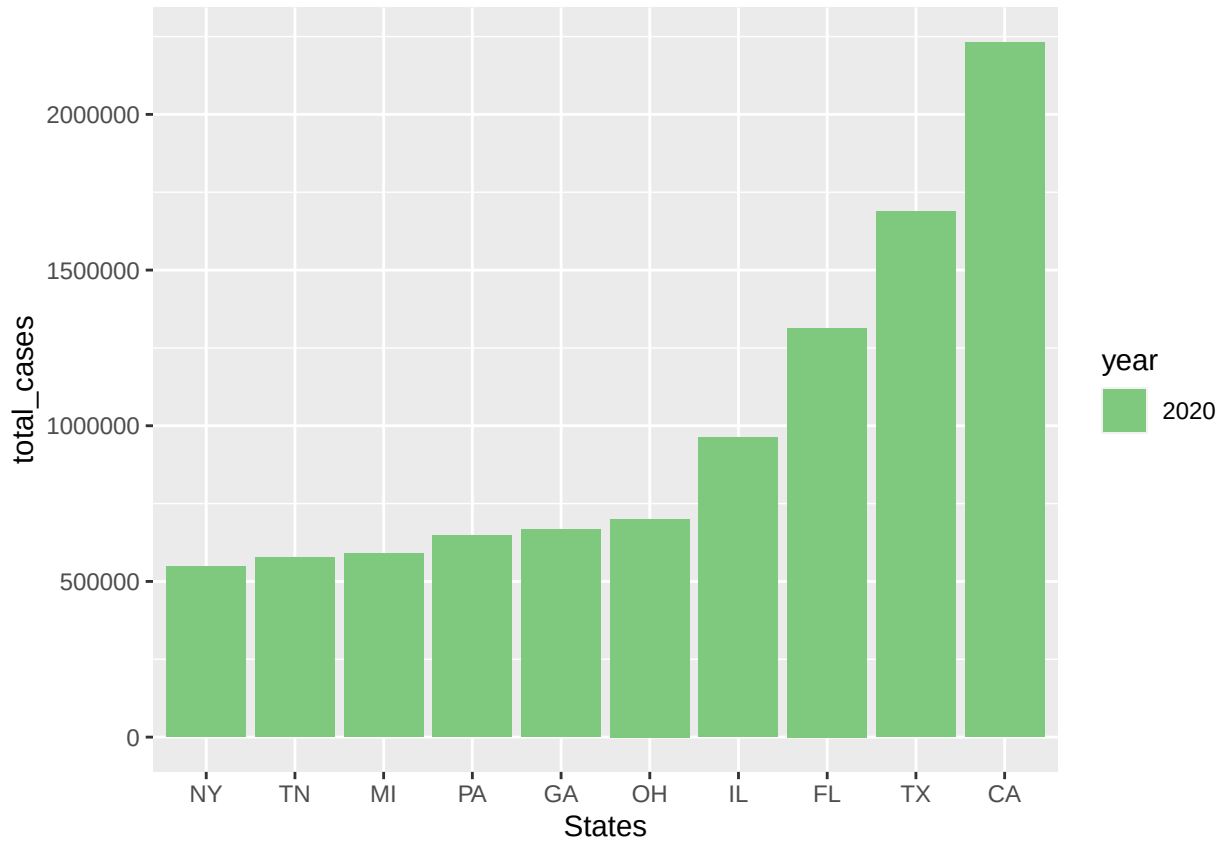
```
#COVID CASES IN THE YEAR 2021

covidmap2021 <- covid_data1 %>%
   filter(year == "2021")
plot_usmap(data = covidmap2021, values = "total_cases", color = "blue", labels = TRUE) +
  scale_fill_continuous(
    low = "white", high = "blue", name = "Total cases (2021)", label = scales::comma
  ) + theme(legend.position = "right")
```



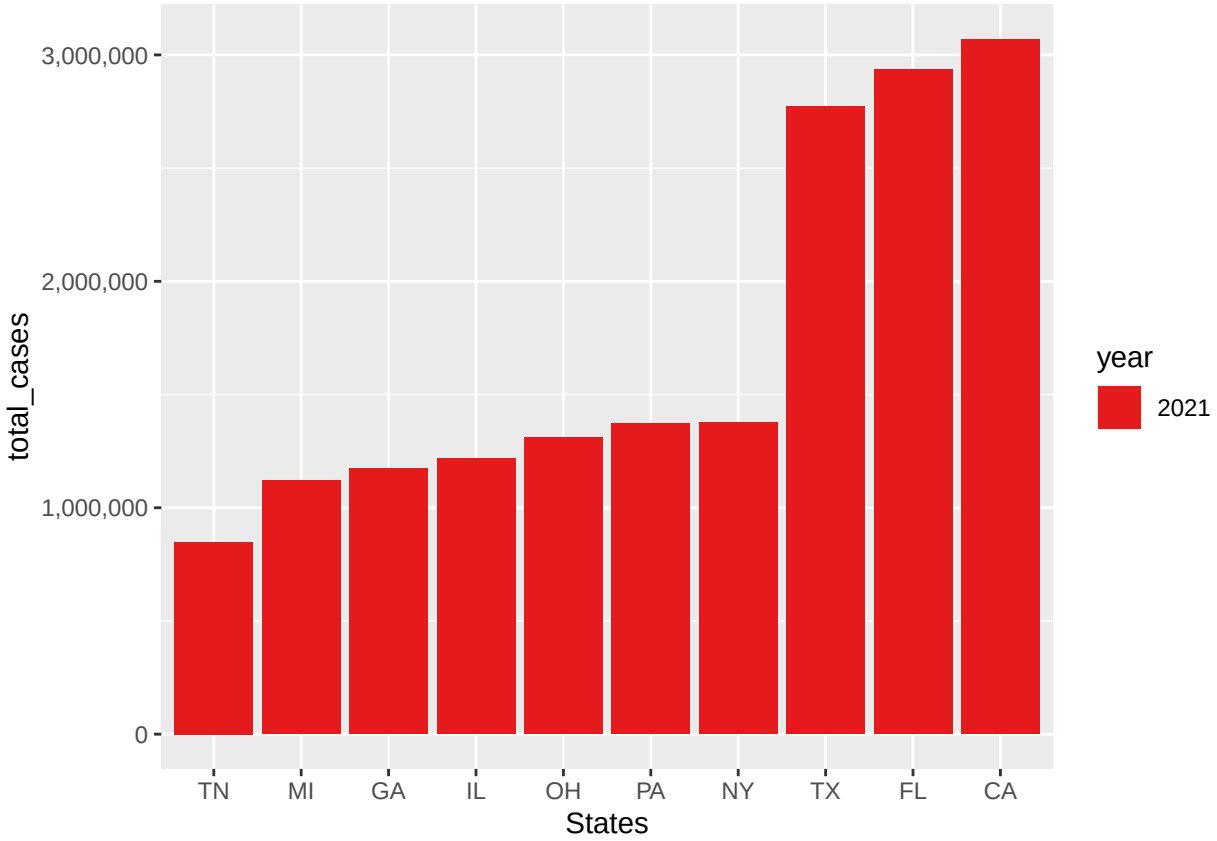
In order to show the skills for bar graph, I have also add the bar graph for covid cases

```
covidplot2020 <- covid_data1 %>%
   filter(year == "2020") %>%
  arrange(desc(total_cases))

covidplot2020 %>% head(n=10) %>%
 ggplot(aes(fct_reorder(state,total_cases), total_cases, fill=year)) +
  geom_bar(position="dodge",stat="identity") +
```

```
  scale_fill_brewer(type = "qual", palette = 1) +
  xlab("States")
```




```
covidplot2021 <- covid_data1 %>%
   filter(year == "2021", state %in% c("CA","TX","FL","IL","OH","GA","PA","MI","TN","NY")) %>%
  arrange(desc(total_cases))

 covidplot2021 %>% head(n=10) %>%
 ggplot(aes(fct_reorder(state,total_cases), total_cases, fill=year)) +
  geom_bar(position="dodge",stat="identity") +
  scale_fill_brewer(type = "qual", palette = 6) +
     xlab("States") +
    scale_y_continuous(labels = comma)
```



Here, I merged both the unemployment dataset Steps:

1) Selected useful variables from first unemployment
2) Performed a left-join for joining both the unemployment datasets. The merged dataset "unemployment_by_state" now have unemployment rates by state for the years 2019, 2020, and 2021.
3) I saved this data in a csv file.

```
unemployment_by_state <-unemployment2019_2020data%>%
  select(state,`2019_rate`, `2020_rate`) %>%
  left_join(unemployment2020_2021data)
```

```
## Joining, by = c("state", "2020_rate")
```

```
unemployment_by_state<- unemployment_by_state%>%
   select(state, `2019_rate`, `2020_rate`,`2021_rate`)
unemployment_by_state <- unemployment_by_state[complete.cases(unemployment_by_state), ]
unemployment_by_state
```

```
## # A tibble: 50 x 4
##    state `2019_rate` `2020_rate` `2021_rate`
##    <chr> <chr>       <chr>       <chr>
##  1 NE    3.0         4.1         2.5
##  2 SD    2.8         4.3         3.1
##  3 UT    2.6         4.7         2.7
##  4 WY    3.7         5.8         4.5
```

```
##  5 ME    2.8         5.0         4.6
##  6 MT    3.6         5.8         3.4
##  7 KY    4.1         6.4         4.7
##  8 ID    3.0         5.5         3.6
##  9 IA    2.6         5.1         4.2
## 10 MS    5.4         7.9         5.6
## # ... with 40 more rows
```

```
write.csv(unemployment_by_state,'unemployment_by_state.csv')
```

It is quite interesting to find out the effects of covid on unemployment. From the past visualizations, we found out the states that had the most covid cases in the year 2020 and 2021.

Therefore, in order to check the covid effects on unemployment, I will be taking 7 states that has most covid cases in 2020 and 2021

5 states with most covid cases "TX","FL","IL","OH" in 2020 5 states with most covid cases "FL","TX","NY", "PA in 2021

In order to see if there is any change in unemployment rate in the year 2020, it is important to find the unemployment rates for 2019 to observe the change.

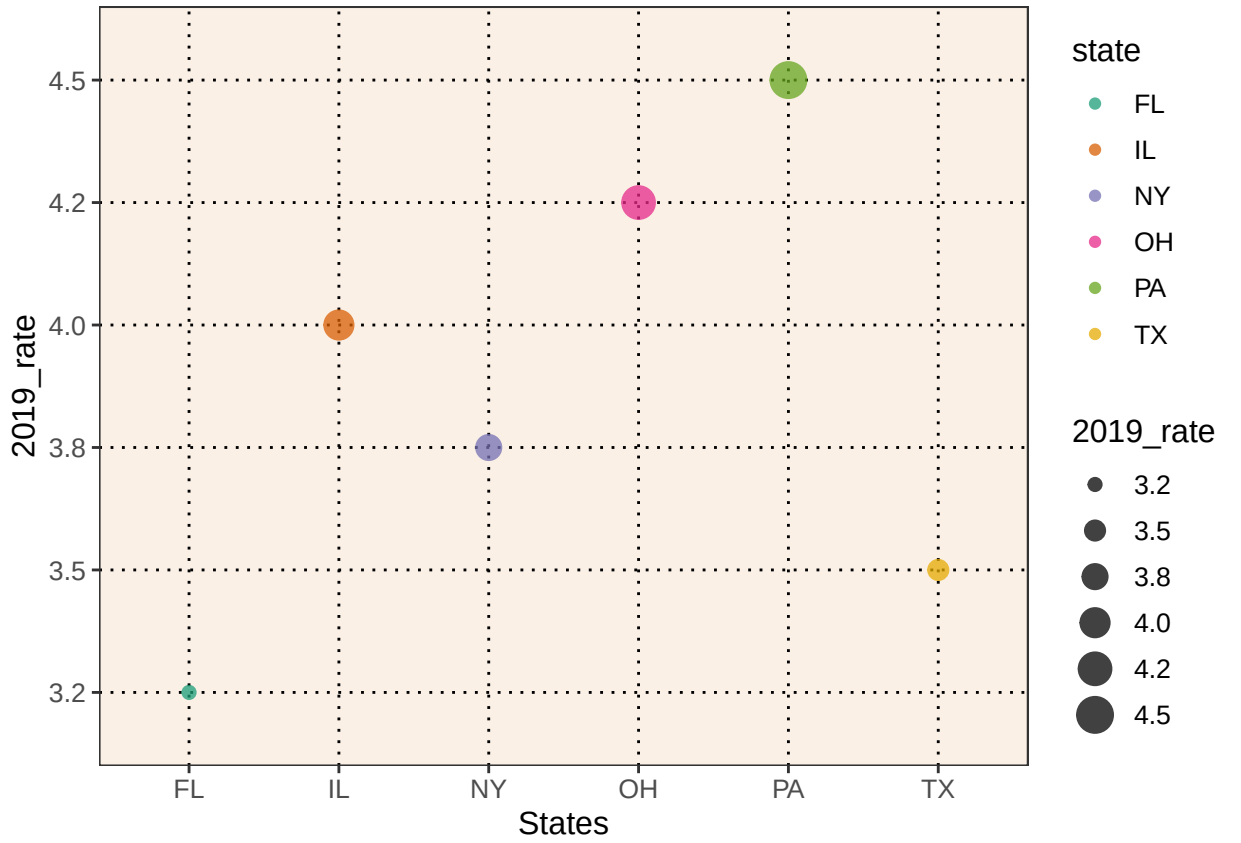
The range of the unemployment rate in 2019 was 3.2 to 4.5

```
unemployment_by_state2019 <-unemployment_by_state %>%
   filter(state %in% c("TX","FL","IL","OH","PA","NY")) %>%
 arrange(desc(`2019_rate`))


unemployment_by_state2019 %>%
  filter(!is.na(state)) %>%
  head(n=7) %>%
  ggplot(aes(fct_relevel( state,
  `2019_rate`), `2019_rate`,
    size = `2019_rate`, color = state )) +
  geom_point(alpha = 0.75) +
  scale_color_brewer(type = "qual", palette = 2) +
  theme_bw() +
  theme(text = element_text(size = 12)) +
  xlab("States") +
 theme(panel.background = element_rect(fill = "linen")) +
  theme(panel.grid.major = element_line(linetype = "dotted", color = "black"))
```

```
## Warning: Unknown levels in `f`: 4.5, 4.2, 4.0, 3.8, 3.5, 3.2
```

```
## Warning: Using size for a discrete variable is not advised.
```

```
## Warning: Unknown levels in `f`: 4.5, 4.2, 4.0, 3.8, 3.5, 3.2
```



Here is the scatterplot adjusted by size and color for the 7 states that had the most cases in 2020 and 2021

The range of the unemployment rate in 2020 was 7.7 to 9.9

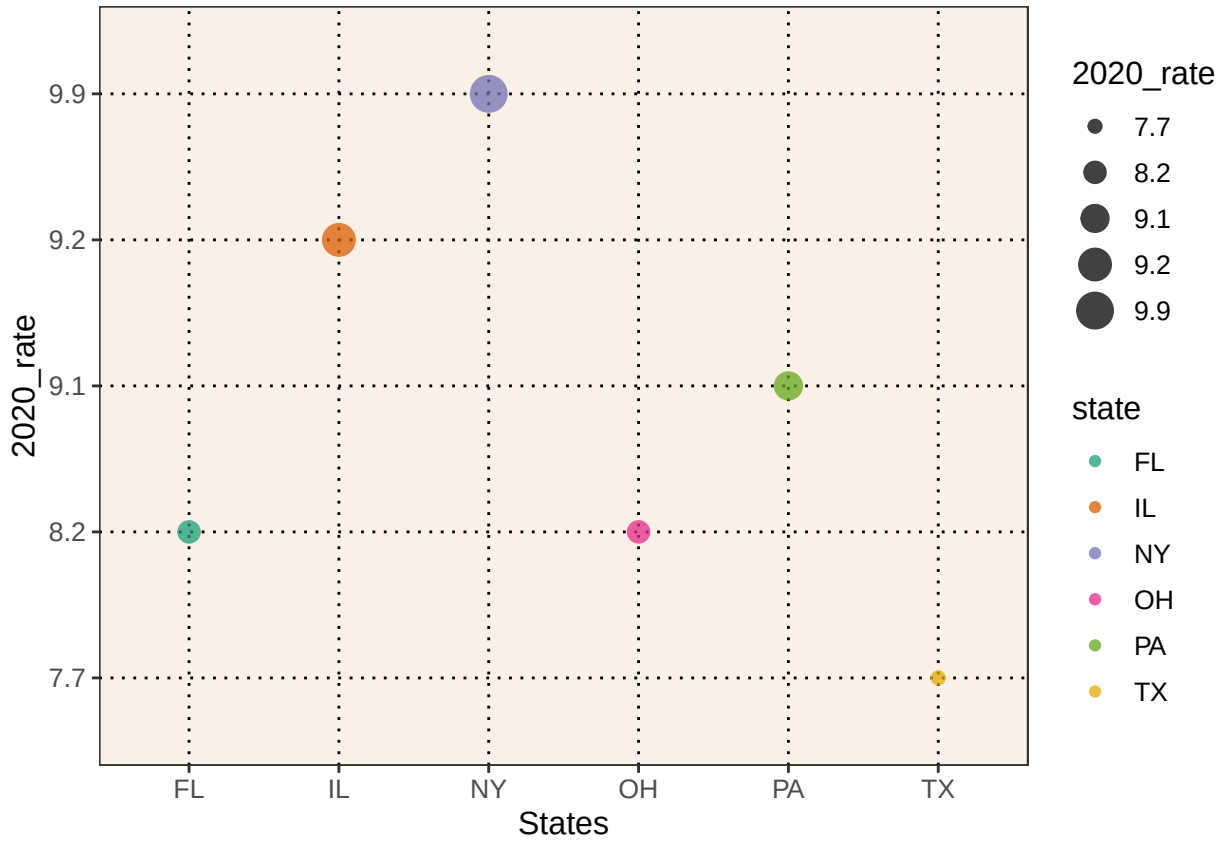
Here, we can definitely see an increase in the range of the unemployment

```
unemployment_by_state2020 <-unemployment_by_state %>%
   filter(state %in% c("TX","FL","IL","OH","PA","NY")) %>%
 arrange(desc(`2020_rate`))

unemployment_by_state2020 %>%
  filter(!is.na(state)) %>%
  head(n=7) %>%
  ggplot(aes(fct_relevel( state,
  `2020_rate`), `2020_rate`,
    size = `2020_rate`, color = state )) +
  geom_point(alpha = 0.75) +
  scale_color_brewer(type = "qual", palette = 2) +
  theme_bw() +
  theme(text = element_text(size = 12)) +
  xlab("States") +
 theme(panel.background = element_rect(fill = "linen")) +
  theme(panel.grid.major = element_line(linetype = "dotted", color = "black"))
```

```
## Warning: Unknown levels in 'f': 9.9, 9.2, 9.1, 8.2, 7.7
```

```
## Warning: Using size for a discrete variable is not advised.
```

```
## Warning: Unknown levels in `f`: 9.9, 9.2, 9.1, 8.2, 7.7
```



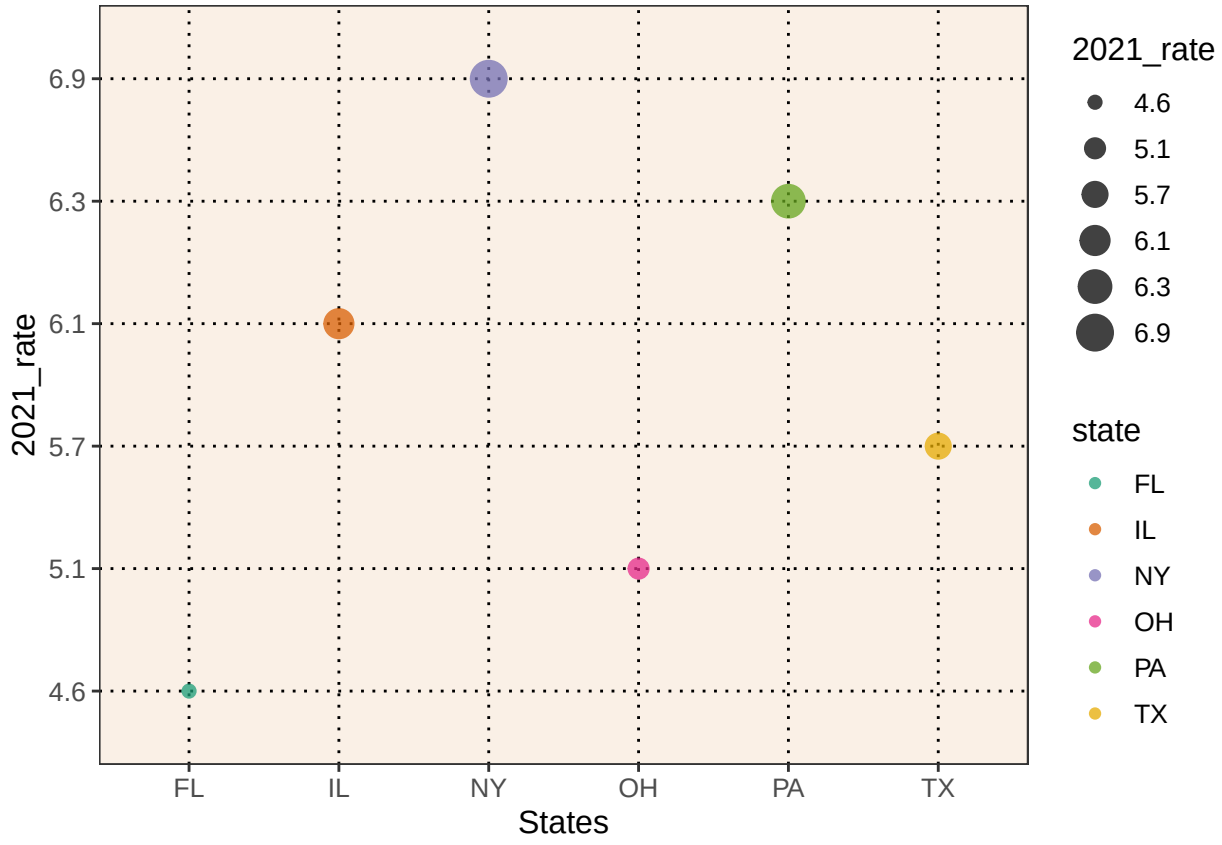
Here is the scatterplot adjusted by size and color for the 7 states that had the most cases in 2020 and 2021

The range of the unemployment rate in 2020 was 4.6 to 6.9

In 2021, we can see the unemployment rates going back to normal like in 2019

```
unemployment_by_state2021 <- unemployment_by_state %>%
  filter(state %in% c("TX","FL","IL","OH","PA","NY")) %>%
 arrange(desc(`2021_rate`))


unemployment_by_state2021%>%
  filter(!is.na(state)) %>%
  head(n=7) %>%
  ggplot(aes(fct_relevel( state,
   `2021_rate`), `2021_rate`,
    size = `2021_rate`, color = state )) +
  geom_point(alpha = 0.75) +
  scale_color_brewer(type = "qual", palette = 2) +
  theme_bw() +
  theme(text = element_text(size = 12)) +
  xlab("States")+
 theme(panel.background = element_rect(fill = "linen")) +
  theme(panel.grid.major = element_line(linetype = "dotted", color = "black"))
```

```
## Warning: Unknown levels in `f`: 6.9, 6.3, 6.1, 5.7, 5.1, 4.6

## Warning: Using size for a discrete variable is not advised.

## Warning: Unknown levels in `f`: 6.9, 6.3, 6.1, 5.7, 5.1, 4.6
```



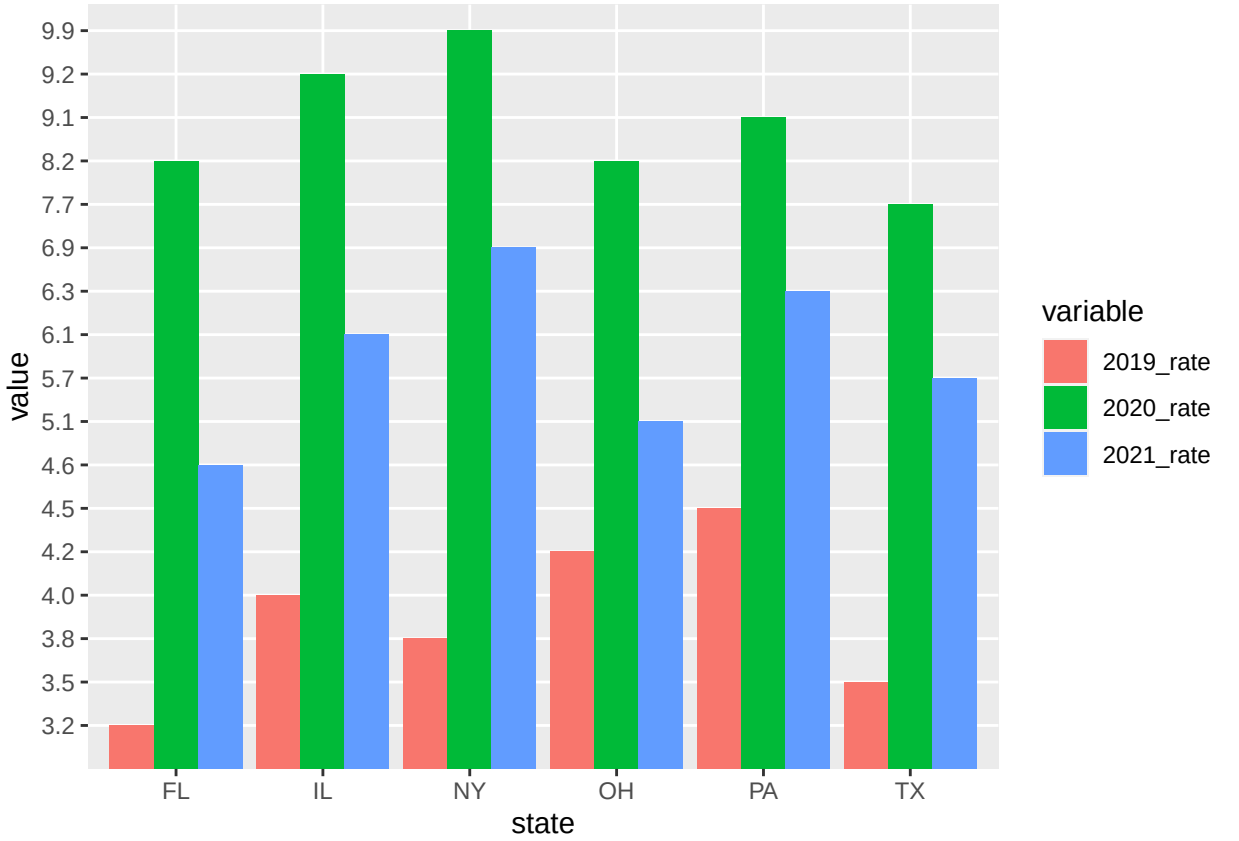
Here is the bar graph showing the unemployment rates of the 7 states in all the years combined 2019, 2020, and 2021

It is easier to see the trend here

We can connect back this to the covid trend. In 2020, there was an increase in the covid cases In 2021, there was a decrease in covid cases

Therefore, it is possible to interpret that increase in covid cases can have an impact on the unemployment rates. As covid cases increases, unemployment also increases

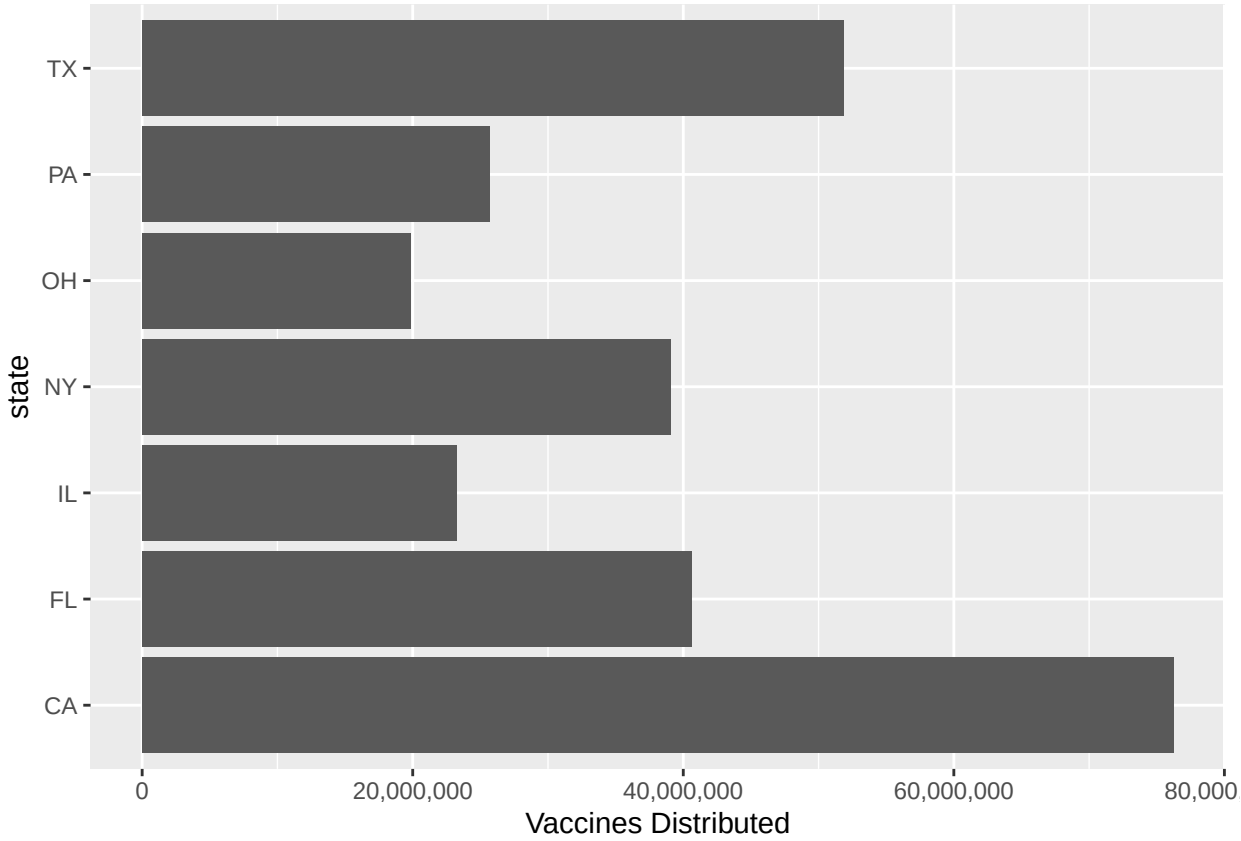
```
#Bar graph of unemployment data for 2019, 2020, 2021

unemployment_plot <- unemployment_by_state%>%
   filter(state %in% c("TX","FL","IL","OH","PA","NY")) %>%
 arrange(desc(`2019_rate`))

dfm1 <- pivot_longer(unemployment_plot, -state, names_to="variable", values_to="value")

ggplot(dfm1,aes(x = state,y = value)) +
    geom_bar(aes(fill = variable),stat = "identity",position = "dodge")
```



What could be the reason behind decreasing covid cases after 2020? Maybe: vaccination Here, we will see the vaccine data visualizations

The Total Distribution of Vaccines by the end of 2021

```
vaccine_data1 <- vaccine_data%>%
   filter(date == "2021-12-31",state %in% c("CA","TX","FL","IL","OH","PA","NY")) %>%
  group_by(state) %>%
  summarize(Distributed = sum(Distributed)) %>%
  arrange(desc(Distributed)) %>%
 ggplot(aes(Distributed, state)) +
  geom_bar(position="dodge",stat="identity") +
  scale_fill_brewer(type = "qual", palette = 1) +
  xlab("Vaccines Distributed") + scale_x_continuous(labels = comma)

vaccine_data1
```



I have filtered the date "2022-04-29" in order to find out the total vaccine distribution till April 2022

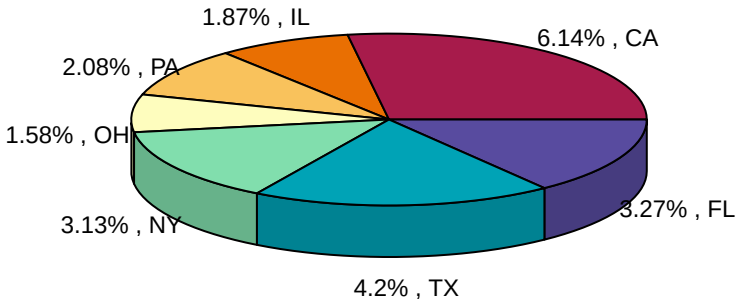
Here, the pie chart is little small so I have provided an image in the pdf

```
vaccine_filtered <- vaccine_data  %>%
  filter(date == "2022-04-29")

Total <- vaccine_filtered %>%
  mutate(percent = paste0(round(vaccine_filtered$Distributed/sum(vaccine_filtered$Distributed) * 100, 2
  filter(state %in% c("CA","TX","FL","IL","OH","PA","NY"))


label <- paste(Total$percent,",",Total$state )

pdf("pie_chart.pdf")

pie3D(Total$Distributed,
      col = hcl.colors(length(Total$Distributed), "Spectral"),
      labels = label, labelcex = 0.75)
dev.off()
```

```
## pdf
##   2
```

```
#Writing same code again to print it out
pie3D(Total$Distributed,
```

```
      col = hcl.colors(length(Total$Distributed), "Spectral"),
      labels = label, labelcex = 0.75)
```



It is interesting to find out what type was vaccine was distributed the most in US states

Here, by dividing the type of distributed vaccine with the total vaccine distribution, the proportion of all the three types of distributed vaccine is calculated

```
vaccine_prop <- vaccine_data %>%
    filter(state %in% c("CA","TX","FL","IL","OH","PA","NY")) %>%
   group_by(state) %>%
  summarise(Distributed = sum(Distributed),Distributed_Janssen = sum(Distributed_Janssen), Distributed_
vaccine_prop <- vaccine_prop %>%
  mutate(Janssen_prop = Distributed_Janssen/Distributed, Moderna_prop =Distributed_Moderna/Distributed,
  select(-Distributed, -Distributed_Janssen, -Distributed_Moderna, -Distributed_Pfizer)
vaccine_prop
```

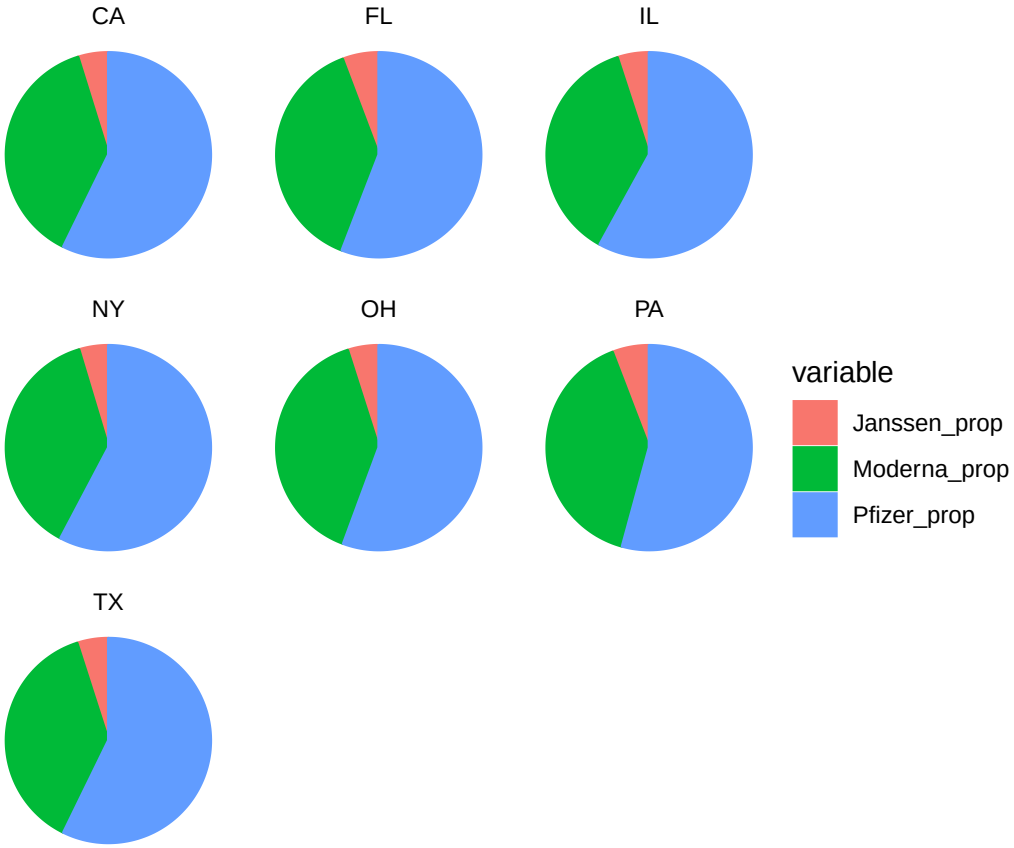
```
## # A tibble: 7 x 4
##   state Janssen_prop Moderna_prop Pfizer_prop
##   <chr>        <dbl>        <dbl>       <dbl>
## 1 CA          0.0464        0.374       0.564
## 2 FL          0.0551        0.378       0.550
## 3 IL          0.0486        0.364       0.571
## 4 NY          0.0452        0.371       0.569
## 5 OH          0.0462        0.388       0.548
## 6 PA          0.0568        0.393       0.534
```

```
## 7 TX          0.0487        0.372       0.564
```

Here is the pie chart of type of vaccine distributed in some of the US states

We can see that the vaccine that was distributed the most in all the states is Pfizer

```
vaccine_propplot <-
  pivot_longer(vaccine_prop, -state, names_to="variable", values_to="value")


vaccine_propplot$state <- factor(vaccine_propplot$state)
vaccine_propplot$variable <- factor(vaccine_propplot$variable)

ggplot(data=vaccine_propplot, aes(x=" ", y=value, group=variable, colour=variable, fill=variable)) +
         geom_bar(width = 1, stat = "identity") +
         coord_polar("y") +
         facet_grid(.~ state) + facet_wrap(~state, ncol = 3)+ theme_void()
```




```
vaccine_propplot
```

```
## # A tibble: 21 x 3
##    state variable      value
##    <fct> <fct>         <dbl>
##  1 CA    Janssen_prop 0.0464
##  2 CA    Moderna_prop 0.374
```

```
##  3 CA    Pfizer_prop  0.564
##  4 FL    Janssen_prop 0.0551
##  5 FL    Moderna_prop 0.378
##  6 FL    Pfizer_prop  0.550
##  7 IL    Janssen_prop 0.0486
##  8 IL    Moderna_prop 0.364
##  9 IL    Pfizer_prop  0.571
## 10 NY    Janssen_prop 0.0452
## # ... with 11 more rows
```