JOHN DAVIS · UPDATED 3 YEARS AGO

54 New Notebook Download (620 MB)

US Counties: COVID19 + Weather + Socio/Health data

Daily COVID19 cases + fatalities, daily weather, and socio/econ/health data

A. DATA OVERVIEW

The COVID-19 pandemic has had a devastating impact on the United States, with millions of cases and hundreds of thousands of deaths. In order to prevent the spread of the virus and protect public health, it is important to understand what factors contribute to the spread of COVID-19.

This dataset provides information about COVID-19 in US. It was taken from Kaggle and includes data on health, socioeconomics, and weather, as well as data on the number of COVID-19 cases and deaths over time. This information can be used to identify factors that are associated with high rates of COVID-19.

Additional Information:

- Not all counties appear in the dataset because not all counties have reported a COVID-19 case by the time this dataset was created in Kaggle.
- The data about health includes information on things like the number of people with diabetes, obesity rates, and HIV rates.
- The data about socioeconomics includes information on things like income levels, poverty rates, and the number of uninsured people.
- The data about weather includes information on things like temperature, rain, snow, etc.

B. EXPLORATORY DATA ANALYSIS (EDA)

In this section, we will take a closer look at our data to learn more about it. We will start by finding out how much data we have, what kind of data it is, and how it is structured. Then, we will look for any anomalies. If we find any anomalies, we will need to handle them. Finally, we will use visualization to help us understand the relationships between the different variables in our data.

Step 1: Import all required libraries

```
library(dplyr)
library(ggplot2)
library(plotly)
```

We use the above code to import libraries. These libraries provide us with the tools we need to analyze the data and identify patterns that may contribute to the spread of COVID-19.

Step 2: Load the dataset

```
healthWeather = read.csv("~/pramudya/matkul/data
mining/archive/US_counties_COVID19_health_weather_data.csv")
geometry = read.csv("~/pramudya/matkul/data
mining/archive/us_county_geometry.csv")
socioHealth = read.csv("~/pramudya/matkul/data
```

```
mining/archive/us_county_sociohealth_data.csv")
```

We use the above code to read the dataset files (`US_counties_COVID19_health_weather_data.csv`, `us_county_geometry.csv`, `us_county_sociohealth_data.csv`) and load it into R.

Step 3: Explore the data to find general information

```
head(geometry)
```

We use head(geometry) to show us the first six rows from geometry dataset. By reading through the output, we know that the output briefly informs us that **this dataset provides geographic data and related attributes about states, counties, and some geometric information about geographic boundaries in the United States. This dataset can be used for geographic analysis, map visualization, and spatial modeling.**

Description: df [6 × 7]

	state <chr>	county <chr>	fips <int>
1	ALABAMA	Autauga	1001
2	ALABAMA	Blount	1009
3	ALABAMA	Chambers	1017
4	ALABAMA	Coffee	1031
5	ALABAMA	Colbert	1033
6	ALABAMA	Covington	1039

6 rows | 1-4 of 7 columns

```
dim(geometry)
```

We use the above code to get the dimensions of geometry dataset. The function returns a tuple of two numbers which is the number of rows and columns.

```
[1] 3142    7
```

In our case, the geometry data frame has **3142 rows and 7 columns**.

```
head(socioHealth)
```

We use head(socioHealth) to show us the first six rows from socioHealth dataset. By reading through the output, we know that the output briefly informs us that **this dataset provides a lot of information regarding the social, demographic, health, economic, and environmental aspects of counties in the United States. This information can be used to conduct analyses related to these factors and provide insights into the socioeconomic and health conditions of people in these area.**

Description: df [6 × 181]

	fips <chr>	state <chr>	county <chr>	lat <dbl>	lon <dbl>	total_population <int>	area_sqmi <dbl>	population_density_per_sqmi <dbl>	num_deaths <int>
1	01001	Alabama	Autauga	32.53493	-86.64275	55049	594.4461	92.60553	791
2	01003	Alabama	Baldwin	30.72749	-87.72258	199510	1589.8074	125.49319	2967
3	01005	Alabama	Barbour	31.86959	-85.39321	26614	884.8758	30.07654	472
4	01007	Alabama	Bibb	32.99863	-87.12648	22572	622.5824	36.25544	471
5	01009	Alabama	Blount	33.98088	-86.56738	57704	644.8065	89.49041	1085
6	01011	Alabama	Bullock	32.10053	-85.71569	10552	622.8054	16.94269	203

6 rows | 1-10 of 181 columns

```
dim(socioHealth)
```

We use the above code to get the dimensions of socioHealth dataset. The function returns a tuple of two numbers which is the number of rows and columns.

```
[1] 3144  181
```

In our case, the socioHealth data frame has **3144 rows and 181 columns**.

```
head(healthWeather)
```

We use head(healthWeather) to show us the first six rows from healthWeather dataset.

Description: df [6 × 227]

	date	county	state	fips	cases	deaths	stay_at_home_announced	stay_at_home_effective	lat
	<date>	<chr>	<chr>	<chr>	<int>	<int>	<chr>	<chr>	<dbl>
144	2020-02-12	Bexar	Texas	48029	1	0	no	no	29.44895
155	2020-02-13	Bexar	Texas	48029	2	0	no	no	29.44895
166	2020-02-14	Bexar	Texas	48029	2	0	no	no	29.44895
212	2020-02-18	Bexar	Texas	48029	2	0	no	no	29.44895
224	2020-02-19	Bexar	Texas	48029	2	0	no	no	29.44895
237	2020-02-20	Bexar	Texas	48029	2	0	no	no	29.44895

6 rows | 1-10 of 227 columns

By reading the output, we know that this dataset is probably a combination of socioHealth and geometry datasets. For that we need to check it further.

Here are some of the reasons why I think this dataset is a combination of those two datasets:

- The socioHealth dataset contains information on social, demographic, and health factors.
- The geometry dataset contains information on the geographic location of each county.
- The output shows that the dataset includes information on both social, demographic, health factors, and geographic location.

```
dim(healthWeather)
```

We use the above code to get the dimensions of healthWeather dataset. The function returns a tuple of two numbers which is the number of rows and columns.

```
[1] 34307    227
```

In our case, the healthWeather data frame has 34307 rows and 227 columns.

Step 4: Check if the healthWeather dataset is a combination of socioHealth dataset and geometry dataset

```
sameColumns <- intersect(names(geometry), names(healthWeather))

if (length(sameColumns) > 0) {
  cat("Same columns:", paste(sameColumns, collapse = ", "))
} else {
  cat("No same column between geometry and healthWeather data")
}
```

We use the code above to check whether there are any common columns between geometry and healthWeather dataset. If there are any common columns between the two datasets, it will print those column names. If there are no common columns, it will print a message indicating that there are no same columns.

```
Same columns: state, county, fips
```

Based on the output of the code, it appears that the datasets geometry and healthWeather have three common columns: state, county, and fips.

```
sameColumns <- intersect(names(socioHealth), names(healthWeather))

if (length(sameColumns) > 0) {
  cat("Number of same columns:", length(sameColumns))
} else {
  cat("No same column between socioHealth and healthWeather data")
}
```

We use the code above to check whether there are any common columns between socioHealth and healthWeather dataset. If there are any common columns between the two datasets, it will print the number of the common columsn. If there are no common columns, it will print a message indicating that there are no same columns.

```
Number of same columns: 181
```

Based on the output, we can see that **the number of same columns in socioHealth and healthWeather is the same as the number of columns in socioHealth dataset.** This means that the healthWeather dataset contains the same information with the socioHealth dataset.

The output of the two codes suggests that the healthWeather dataset is likely a combination of the socioHealth and geometry datasets. However, not all of the contents of the two datasets are in the healthWeather dataset. The date column in the healthWeather dataset is not present in the socioHealth or geometry datasets. This suggests that the healthWeather dataset is a more comprehensive dataset than the other two datasett because it shows the number of Covid cases and deaths over time. Therefore, **from now on until the end I will only use the healthWeather dataset.**

Step 5: Check if the healthWeather dataset has any missing values

```
print(paste("healthWeather:", any(is.na(healthWeather))))
```

We used the code above to check if there are any missing values (NA) in the healthWeather dataset.

```
[1] "healthWeather: TRUE"
```

As we can see, **the dataset contains missing values.**

```
colSums(is.na(healthWeather))
```

We used the code above to show us the number of missing values in each column of the dataset.

```
                                   date                                  county
                                      0                                       0
                                  state                                    fips
                                      0                                     163
                                  cases                                  deaths
                                      0                                   16655
                 stay_at_home_announced                  stay_at_home_effective
                                      0                                       0
                                    lat                                     lon
                                  17835                                   17835
                       total_population                               area_sqmi
                                  17835                                   17835
           population_density_per_sqmi                              num_deaths
                                  17835                                   74408
       years_of_potential_life_lost_rate                percent_fair_or_poor_health
                                  74408                                   17835
average_number_of_physically_unhealthy_days  average_number_of_mentally_unhealthy_days
                                  17835                                   17835
                 percent_low_birthweight                         percent_smokers
                                  35382                                   17835
             percent_adults_with_obesity                   food_environment_index
                                  17835                                   22449
             percent_physically_inactive  percent_with_access_to_exercise_opportunities
                                  17835                                   18701
              percent_excessive_drinking          num_alcohol_impaired_driving_deaths
                                  17835                                   24107
                      num_driving_deaths  percent_driving_deaths_with_alcohol_involvement
                                  24107                                   24107
                      num_chlamydia_cases                           chlamydia_rate
                                  45401                                   45401
                         teen_birth_rate                            num_uninsured
                                  45172                                   17835
                      percent_uninsured                 num_primary_care_physicians
                                  17835                                   51561
           primary_care_physicians_rate                            num_dentists
                                  51047                                   38665
```

```
                          dentist_rate            num_mental_health_providers
                                 38665                                  67855
             mental_health_provider_rate    preventable_hospitalization_rate
                                 67855                                  25000
           percent_with_annual_mammogram                  percent_vaccinated
                                 20804                                  20649
             high_school_graduation_rate                    num_some_college
                                 37113                                  17835
                            population                  percent_some_college
                                 17835                                  17835
                    num_unemployed_CHR                           labor_force
                                 17835                                  17835
                percent_unemployed_CHR           percent_children_in_poverty
                                 17835                                  17835
             eightieth_percentile_income          twentieth_percentile_income
                                 18326                                  18326
                          income_ratio        num_single_parent_households_CHR
                                 18326                                  17853
                    num_households_CHR    percent_single_parent_households_CHR
                                 17853                                  17853
                      num_associations               social_association_rate
                                 17835                                  17835
           annual_average_violent_crimes                  violent_crime_rate
                                 61879                                  61879
                      num_injury_deaths                    injury_death_rate
                                 33978                                  33978
                     average_daily_pm2_5          presence_of_water_violation
                                 24632                                  28439
        percent_severe_housing_problems           severe_housing_cost_burden
                                 17835                                  17835
                          overcrowding                inadequate_facilities
                                 17835                                  17835
             percent_drive_alone_to_work           num_workers_who_drive_alone
                                 18090                                  18090
     percent_long_commute_drives_alone                       life_expectancy
                                 18090                                  29083
                          num_deaths_2               age_adjusted_death_rate
                                 26804                                  26804
                          num_deaths_3                  child_mortality_rate
                                299236                                 299236
                          num_deaths_4                 infant_mortality_rate
                                462963                                 462963
      percent_frequent_physical_distress      percent_frequent_mental_distress
                                 17835                                  17835
             percent_adults_with_diabetes                         num_hiv_cases
                                 17835                                 215528
                    hiv_prevalence_rate                      num_food_insecure
                                215528                                  17835
                  percent_food_insecure                     num_limited_access
                                 17835                                  22449
 percent_limited_access_to_healthy_foods              num_drug_overdose_deaths
                                 22449                                 347038
             drug_overdose_mortality_rate              num_motor_vehicle_deaths
                                347038                                 109543
             motor_vehicle_mortality_rate             percent_insufficient_sleep
                                109543                                  17835
                         num_uninsured_2                   percent_uninsured_2
                                 17835                                  17835
                         num_uninsured_3                   percent_uninsured_3
                                 17835                                  17835
       other_primary_care_provider_rate            percent_disconnected_youth
                                 22999                                 435025
               average_grade_performance           average_grade_performance_2
                                169680                                 178054
                 median_household_income  percent_enrolled_in_free_or_reduced_lunch
                                 17835                                  48446
                       segregation_index                   segregation_index_2
                                260344                                  87899
                          homicide_rate                          num_deaths_5
                                457337                                 186307
              suicide_rate_age_adjusted               num_firearm_fatalities
                                186307                                 222459
                  firearm_fatalities_rate                   juvenile_arrest_rate
                                222459                                 257075
average_traffic_volume_per_meter_of_major_roadways                 num_homeowners
                                 17835                                  17835
```

As we can see **most of our data in the dataset contains missing values**. Since it is overwhelming too handle, I will just drop the row with the missing values.

```
healthWeather <- na.omit(healthWeather)
print(paste("healthWeather:", any(is.na(healthWeather))))
```

We use the above code to drop all the rows that has missing values and check again whether our dataset is already clean from missing values or not.

```
[1] "healthWeather: FALSE"
```

As we can see our **dataset is already clean** (doesn't contains any missing values anymore)

```
write.csv(healthWeather, "~/pramudya/matkul/data mining/LEC & LAB/LAB Project/healthWeather.csv", row.names = F)
```

We use the code above to save our current datframe that has been **cleaned into a csv files.**

Step 6: Gain additional information

```
print(length(unique(healthWeather$county)))
```

We use the code above to display the number of counties in the dataset.

```
[1] 200
```

Based on the output, we can see that **our dataset has 200 counties**.

```
print(length(unique(healthWeather$state)))
```

We use the code above to display the number of states in the dataset.

```
[1] 23
```

Based on the output we can see that the dataset contains information on 23 states. This means that **the 200 counties in the dataset are spread across 23 states**.

```
unique(healthWeather$state)
```

We use the code above to display all the states in the dataset.

```
 [1] "Texas"          "Oregon"         "California"     "Florida"        "Georgia"        "North Carolina"
 [7] "Colorado"       "Maryland"       "Indiana"        "Minnesota"      "Kansas"         "Missouri"
[13] "Virginia"       "Ohio"           "Michigan"       "Mississippi"    "Alabama"        "Illinois"
[19] "Maine"          "Wisconsin"      "Arizona"        "Montana"        "Rhode Island"
```

Based on the output of, we can conclude that the dataset contains information from various states in the United States. Some of the states included in the dataset are **Texas, Oregon, California, Florida, Georgia, North Carolina, Colorado, Maryland, Indiana, Minnesota, Kansas, Missouri, Virginia, Ohio, Michigan, Mississippi, Alabama, Illinois, Maine, Wisconsin, Arizona, Montana, and Rhode Island**.

```
format(sum(healthWeather$cases), big.mark = ",")
```

We use the code above to display the total number of cases from all the counties in the dataset.

```
[1] "195,693,032"
```

Based on the output, it can be seen that the dataset shows that **there has been a total of 195,693,032 cases of COVID-19 in the 23 states** included in the dataset.

```
format(sum(healthWeather$deaths), big.mark = ",")
```

We use the code above to display the total number of deaths from Covid cases from all counties in the dataset.

```
[1] "4,821,640"
```

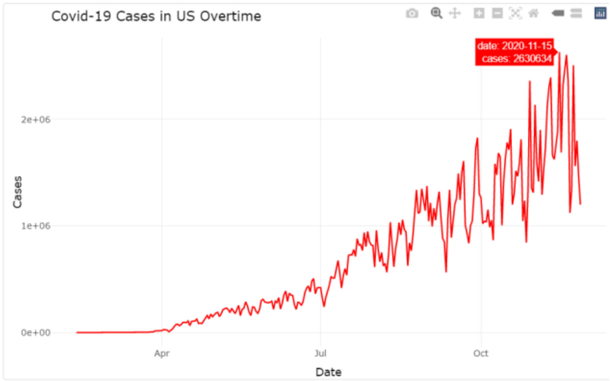
Based on the output, it can be seen that **the total number of deaths due to Covid in the dataset is 4,821,640 deaths**.

```
healthWeather$date <- as.Date(healthWeather$date)
```

The code above is used to **convert the date column in the dataset to the Date format.**

```
casesOvertime <- aggregate(cases ~ date, data = healthWeather, sum)
plot1 <- ggplot(casesOvertime, aes(y = cases, x = date)) + geom_line(color
= "red", stat="identity") + labs(title = "Covid-19 Cases in US Overtime",
x = "Date", y = "Cases") + theme_minimal()
ggplotly(plot1)
```

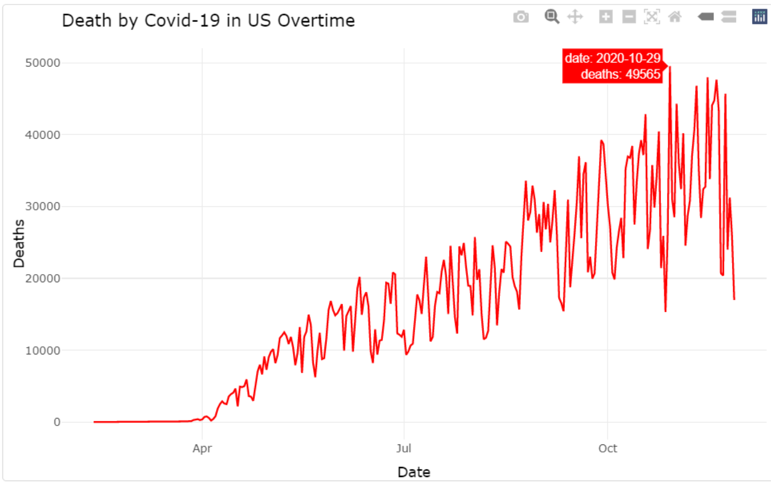
We use the code above to visualize Covid-19 Cases in US overtime.



The output shows that the number of Covid-19 cases per day has been increasing over time. This suggests that the virus is spreading rapidly. There have been some drops in the number of cases, but these have been temporary. The highest number of cases on a single day was 2,630,634, which occurred on November 15, 2020.

```
deathsOvertime <- aggregate(deaths ~ date, data = healthWeather, sum)
plot2 <- ggplot(deathsOvertime, aes(y = deaths, x = date)) +
geom_line(color = "red", stat="identity") + labs(title = "Death by Covid-
19 in US Overtime", x = "Date", y = "Deaths") + theme_minimal()
ggplotly(plot2)
```

We use the code above to visualize Deaths by Covid-19 in US overtime.



The output shows that the number of Covid-19 deaths per day has been increasing over time. There have been some drops in the number of deaths, but these have been temporary. The highest number of deaths on a single day was 49,565, which occurred on October 29, 2020.

```
casesPerState <- aggregate(cases ~ state, data = healthWeather, sum)
print(casesPerState)
```

We use the code above to display the number of Covid-19 cases for each state in the dataset.

Description: df [23 x 2]

state <chr>	cases <int>
Alabama	3603940
Arizona	77676
California	3411555
Colorado	4585596
Florida	47018461
Georgia	8154282
Illinois	7719544
Indiana	6398149
Kansas	3469527
Maine	341132
Maryland	5059194
Michigan	13048550
Minnesota	5381918
Mississippi	1158922
Missouri	6687829
Montana	100763
North Carolina	2998033
Ohio	9695947
Oregon	238780
Rhode Island	2117144
Texas	53487346
Virginia	3030634
Wisconsin	7908110

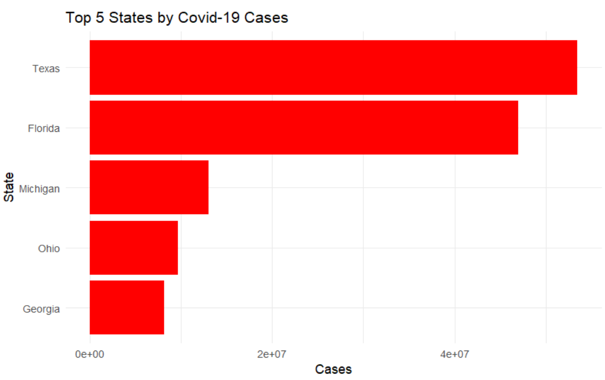
1-10 of 23 rows

11-20 of 23 rows

From the output we can see that **there are some states that have a very large number of cases compared to other states**. This could be due to a number of factors, such as the state's socio economic and weather.

```
casesPerState <- casesPerState[order(-casesPerState$cases), ]
top5 <- head(casesPerState, 5)
top5$state <- factor(top5$state, levels = top5$state[order(top5$cases)])

ggplot(data = top5, aes(x = cases, y = state)) +
  geom_bar(stat = "identity", fill = "red") +
  labs(title = "Top 5 States by Covid-19 Cases", x = "Cases", y = "State")
+ theme_minimal()
```

We use the code above to visualize the 5 states with the highest number of Covid cases.



The output shows that **Texas and Florida have the highest number of COVID-19 cases, followed by Michigan, Ohio, and Georgia.** The number of cases in Texas and Florida is significantly higher than in the other states.

```
belowPoverty <- aggregate(num_below_poverty ~ state, healthWeather, mean)
belowPoverty <- belowPoverty[order(-belowPoverty$num_below_poverty), ]
belowPoverty$num_below_poverty <- round(belowPoverty$num_below_poverty)
belowPoverty <- belowPoverty[, c("state", "num_below_poverty")]
```

```
belowPoverty$num_below_poverty <-
as.integer(belowPoverty$num_below_poverty)

print(belowPoverty)
```

We use the code above to display the num of people below poverty line in each state and display it as an integer.

	state <chr>	num_below_poverty <int>
3	California	162560
17	North Carolina	122638
20	Rhode Island	104178
5	Florida	97671
21	Texas	86725
12	Michigan	85202
13	Minnesota	62048
4	Colorado	61391
6	Georgia	57289
18	Ohio	48404

1-10 of 23 rows

	state <chr>	num_below_poverty <int>
1	Alabama	45008
23	Wisconsin	40356
8	Indiana	37765
15	Missouri	37624
11	Maryland	36044
7	Illinois	32231
19	Oregon	31183
9	Kansas	30482
22	Virginia	27534
10	Maine	22074

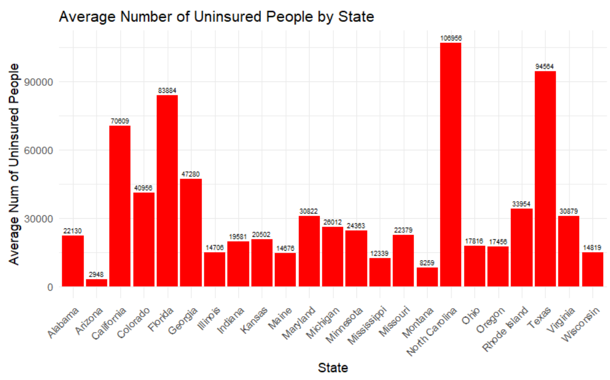
11-20 of 23 rows

	state <chr>	num_below_poverty <int>
14	Mississippi	15774
16	Montana	12875
2	Arizona	7419

The output shows that **California has the highest number of people living below the poverty line, with 162,560 people**. Arizona has the lowest number of people living below the poverty line, with 7,419 people.

```
uninsured <- aggregate(num_uninsured ~ state, healthWeather, mean)
uninsured$num_uninsured <- round(uninsured$num_uninsured)
uninsured <- uninsured[, c("state", "num_uninsured")]
uninsured$num_uninsured <- as.integer(uninsured$num_uninsured)

ggplot(data = uninsured, aes(x = state, y = num_uninsured)) +
  geom_bar(stat = "identity", fill = "red") +
  geom_text(aes(label = num_uninsured), vjust = -0.5, color = "black",
size = 2) +
  labs(title = "Number of Uninsured People by State", x = "State", y =
"Average Num of Uninsured People") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

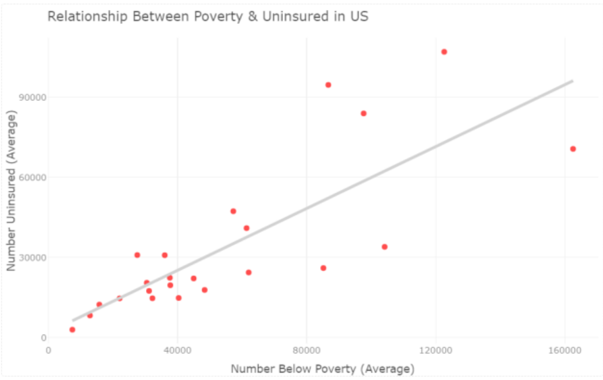
We use the code above to create a bar plot showing the average number of uninsured people by state.



The output shows that **North Carolina has the highest average number of uninsured people per state, with 106,956 people.** This means that, on average, there are 106,956 people in North Carolina who do not have health insurance.

```
averages <- aggregate(cbind(num_uninsured, num_below_poverty) ~ state,
data = healthWeather, mean)
averages$num_uninsured <- round(averages$num_uninsured)
averages$num_below_poverty <- round(averages$num_below_poverty)

plot3 <- ggplot(data = averages, aes(x = num_below_poverty, y =
num_uninsured)) + geom_point(color="red") +
geom_smooth(method = "lm", se = FALSE, color = "grey") + labs(title =
"Relationship Between Poverty & Uninsured in US",x = "Number Below Poverty
(Average)", y = "Number Uninsured (Average)") + theme_minimal()
ggplotly(plot3)
```

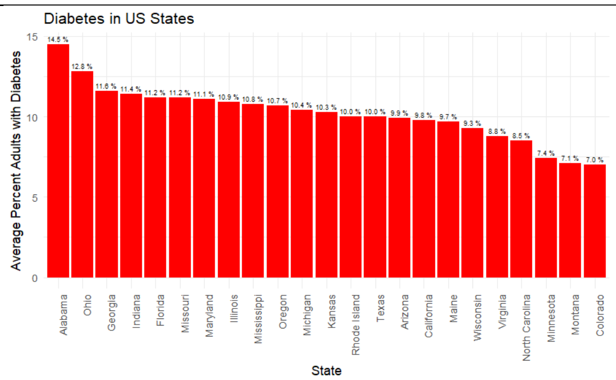
we use the code above to create a scatter plot that can show the relationship between num of below poverty and num of uninsured.



The output shows that there is a correlation between poverty and lack of health insurance. This means that people who are living below the poverty line are more likely to be uninsured. It is most likely because people who are living below the poverty line may not be able to afford health insurance.

```
diabetes <- aggregate(percent_adults_with_diabetes ~ state,
healthWeather, mean)
diabetes$percent_adults_with_diabetes <-
round(diabetes$percent_adults_with_diabetes, 1)
diabetes <- diabetes[, c("state", "percent_adults_with_diabetes")]
diabetes$state <- factor(diabetes$state, levels = diabetes$state[order(-
diabetes$percent_adults_with_diabetes)])

ggplot(data = diabetes, aes(x = state, y =
percent_adults_with_diabetes)) +
  geom_bar(stat = "identity", fill = "red") +
  geom_text(aes(label = paste(sprintf("%.1f",
percent_adults_with_diabetes), "%")), vjust = -0.5, color = "black",
size = 2) +
  labs(title = "Diabetes in US States", x = "State", y = "Average
Percent Adults with Diabetes") +
  theme_minimal() + theme(axis.text.x = element_text(angle = 90, hjust =
1))
```

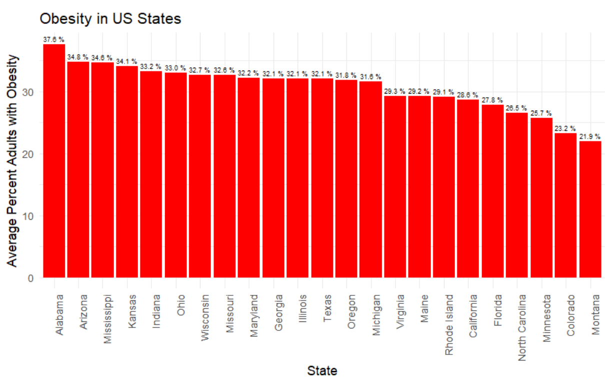
We use the code above to create a bar plot showing the average percent adults with diabetes per state and then sort them from the state with the highest average percent adults with diabetes to the lowest.



From this we can see that **the state with the highest average percent adult with diabetes is Alabama** while the lowest is Colorado with a value of 7%.

```
obesity <- aggregate(percent_adults_with_obesity ~ state, healthWeather, mean)
obesity$percent_adults_with_obesity <- round(obesity$percent_adults_with_obesity, 1)
obesity <- obesity[, c("state", "percent_adults_with_obesity")]
obesity$state <- factor(obesity$state, levels = obesity$state[order(-obesity$percent_adults_with_obesity)])

ggplot(data = obesity, aes(x = state, y = percent_adults_with_obesity)) +
  geom_bar(stat = "identity", fill = "red") +
  geom_text(aes(label = paste(sprintf("%.1f", percent_adults_with_obesity), "%")), vjust = -0.5, color = "black", size = 2) +
  labs(title = "Obesity in US States", x = "State", y = "Average Percent Adults with Obesity") +
  theme_minimal() + theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

We use the code above to create a bar plot showing the average percent adults with obesity per state and then sort them from the state with the highest average percent adults with obesity to the lowest.

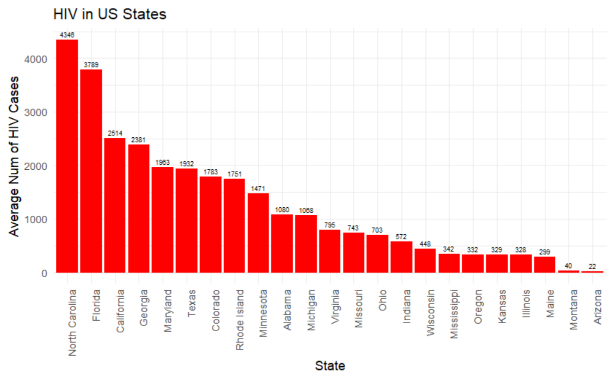


From this we can see that the state with **the highest average percent adult with obesity is Alabama with 37.6% while the lowest is Montana with 21.9%.**

```
HIV <- aggregate(num_hiv_cases ~ state, healthWeather, mean)
HIV$num_hiv_cases <- round(HIV$num_hiv_cases)
HIV <- HIV[, c("state", "num_hiv_cases")]
HIV$num_hiv_cases <- as.integer(HIV$num_hiv_cases)
```

```
HIV$state <- factor(HIV$state, levels = HIV$state[order(-
HIV$num_hiv_cases)])

ggplot(data = HIV, aes(x = state, y = num_hiv_cases)) +
  geom_bar(stat = "identity", fill = "red") +
  geom_text(aes(label = num_hiv_cases), vjust = -0.5, color = "black",
size = 2) +
  labs(title = "HIV in US States", x = "State", y = "Average Num of HIV
Cases") +
  theme_minimal() + theme(axis.text.x = element_text(angle = 90, hjust =
1))
```

We use the code above to create a bar plot showing the average number of HIV cases per state and then sort them from the state with the highest average number of HIV cases to the lowest.



From this we can see that the state with the highest average num of HIV cases is Northern California with 4346 while the lowest is Arizona with average 22 cases.

```
meanTempPerState <- aggregate(mean_temp ~ state, data = healthWeather,
function(x) round(mean(x), 1))
print(meanTempPerState)
```

We use the code above to show the average mean temperature of each state in the US.

state <chr>	mean_temp <dbl>
Alabama	72.2
Arizona	79.5
California	72.5
Colorado	64.2
Florida	78.0
Georgia	70.3
Illinois	60.9
Indiana	62.0
Kansas	65.7
Maine	55.2

1-10 of 23 rows

state <chr>	mean_temp <dbl>
Maryland	63.5
Michigan	58.5
Minnesota	58.2
Mississippi	72.3
Missouri	64.6
Montana	51.2
North Carolina	68.5
Ohio	60.3
Oregon	55.7
Rhode Island	57.1

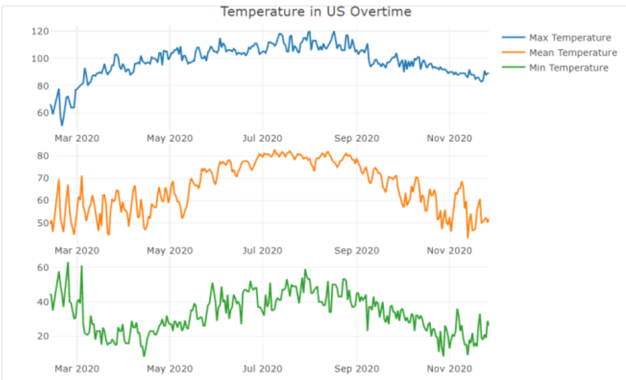
11-20 of 23 rows

state <chr>	mean_temp <dbl>
Texas	75.5
Virginia	65.0
Wisconsin	56.6

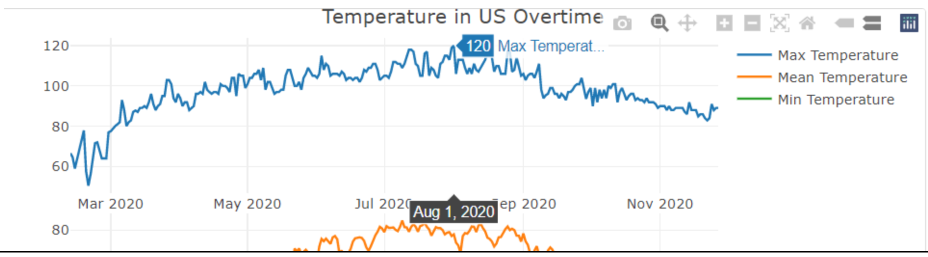
The output shows that **Arizona has the highest average mean temperature, at 79.5 °F. Montana has the lowest average mean temperature, at 51.2 °F.** Arizona is a desert state, which means that it has hot-dry summers and mild winters. Montana is a mountainous state, which means that it has cold winters and warm summers.

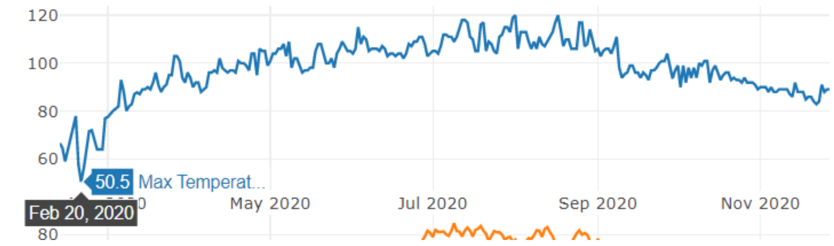
```
maxTempOvertime <- aggregate(max_temp ~ date, data = healthWeather,
function(x) round(max(x), 1))
meanTempOvertime <- aggregate(mean_temp ~ date, data = healthWeather,
function(x) round(mean(x), 1))
minTempOvertime <- aggregate(min_temp ~ date, data = healthWeather,
function(x) round(min(x), 1))

plot4 <- plot_ly(data = maxTempOvertime, x = ~date, y = ~max_temp, type =
'scatter', mode = 'lines', name = 'Max Temperature') %>%
  layout(title = 'Temperature in US Overtime', xaxis = list(title =
'Date'), yaxis = list(title = 'Max Temperature °F'))

plot5 <- plot_ly(data = meanTempOvertime, x = ~date, y = ~mean_temp, type
= 'scatter', mode = 'lines', name = 'Mean Temperature') %>%
  layout(title = 'Temperature in US Overtime', xaxis = list(title =
'Date'), yaxis = list(title = 'Mean Temperature °F'))

plot6 <- plot_ly(data = minTempOvertime, x = ~date, y = ~min_temp, type =
'scatter', mode = 'lines', name = 'Min Temperature') %>%
  layout(title = 'Temperature in US Overtime', xaxis = list(title =
'Date'), yaxis = list(title = 'Min Temperature °F'))

subplot(plot4, plot5, plot6, nrows = 3)
```

We use the code above to create a plot that shows the maximum temperature, mean temperature, and minimum temperature over time. We then plot these on one plot with three rows and one column.

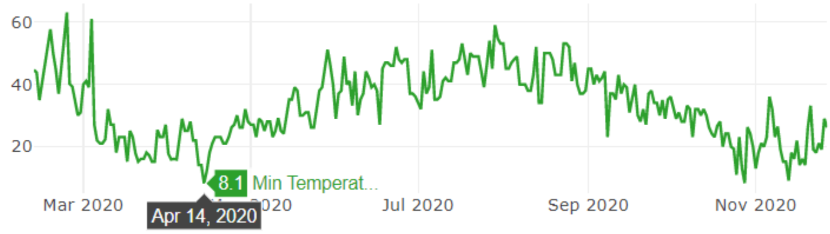


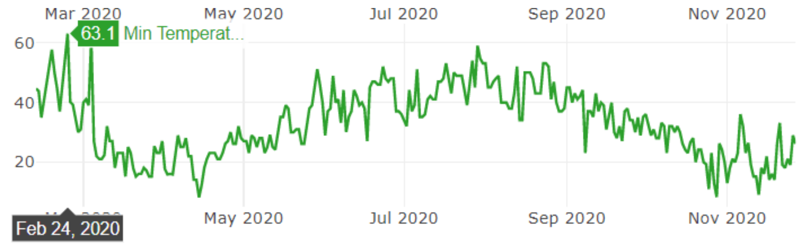
The plot above shows that the highest maximum temperature was on August 1, 2020, with a temperature of 120 °F. The lowest maximum temperature was on February 20, 2020, with a temperature of 50.5 °F. The high temperatures in August are due to the fact that the sun is directly overhead at that time of year.





We can also see that the lowest minimum temperature was on April 14, 2020, with a temperature of 8.1 °F and the highest minimum temperature was on February 24, 2020, with a temperature of 63.1 °F.






```
print(paste(max(healthWeather$max_temp), "°F"))
```

We use the code above to show the highest max temp in the dataset.

```
[1] "120 °F"
```

We can see that **the highest max temp is 120 °F.**

```
print(paste(round(mean(healthWeather$mean_temp),1), "°F"))
```

We use the code above to show the average mean temp of the dataset.

```
[1] "67.2 °F"
```

We can see that **the average mean temp is 67.2 °F**.

```
print(paste(round(min(healthWeather$min_temp),1), "°F"))
```

We use the code above to show the lowest min temp in the dataset.

```
[1] "8.1 °F"
```

We can see that **the lowest min temp is 8.1 °F.**

C. CONCLUSION

In this analysis, I utilized a dataset comprising COVID-19, socioeconomic, and weather data to gain insights into the spread of the virus and its correlation with various factors in the United States. By examining the dataset and performing data exploration, we were able to draw several conclusions:

1. There are 3 datasets: the geometry dataset, which provided geographic information and boundaries of states and counties; the socioHealth dataset,

which offered comprehensive data on social, demographic, health, economic, and environmental aspects of counties; and the healthWeather dataset, which appeared to be a combination of the socioHealth and geometry datasets, containing information on COVID-19 cases, deaths, and weather conditions.

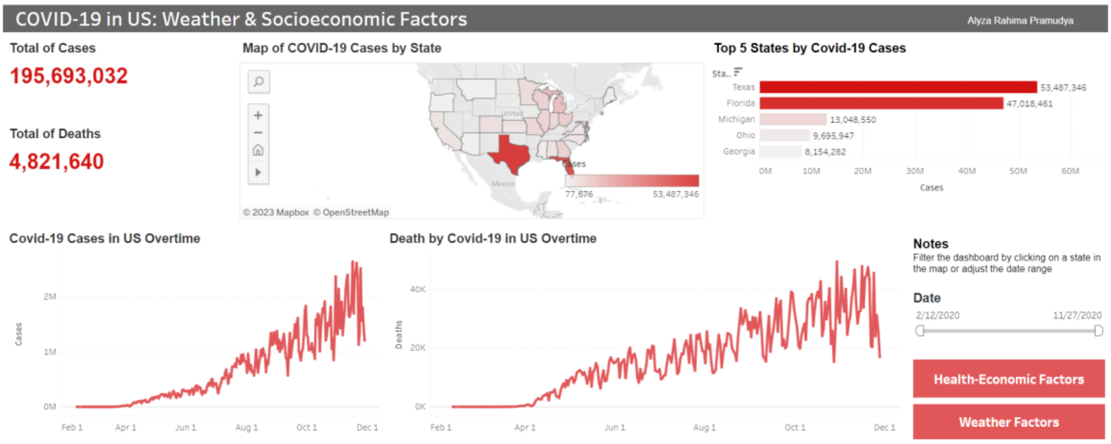
2. The dimensions of the datasets are as follows: The geometry dataset has 3142 rows and 7 columns, the socioHealth dataset has 3144 rows and 181 columns, and the healthWeather dataset has 34307 rows and 227 columns.
3. By comparing the columns in the datasets, we identified the same columns between the geometry and healthWeather datasets (states, regions, and fips). In addition, the number of columns in common between the socioHealth and healthWeather datasets is equal to the number of columns in the socioHealth dataset, indicating that the healthWeather dataset aggregates socioHealth data.
4. The dataset included information from 200 counties in 23 states, including Texas, Oregon, California, Florida, Georgia, North Carolina, Colorado, Maryland, Indiana, Minnesota, Kansas, Missouri, Virginia, Ohio, Michigan, Mississippi, Alabama, Illinois, Maine, Wisconsin, Arizona, Montana, and Rhode Island.
5. The total number of COVID-19 cases in the dataset was 195,693,032, with 4,821,640 deaths attributed to the virus.
6. There was a notable correlation between poverty and lack of health insurance, suggesting that individuals living below the poverty line were more likely to be uninsured. This finding emphasizes the socioeconomic disparities in access to healthcare and the importance of addressing these disparities to mitigate the impact of COVID-19.
7. Temperature and humidity were also considered as potential factors influencing COVID-19 transmission. However, warmer regions in the dataset exhibited distinct socioeconomic and health demographics, with higher rates of poverty, obesity, and diabetes.

D. TABLEAU

Here are the links to the Tableau dashboards and screenshots of each dashboard. I created three dashboards in Tableau: the main dashboard, the health economic factors dashboard, and the weather factors dashboard. To access the health economic factors and weather factors dashboards, click the red button in the bottom right corner of the main dashboard.

MAIN DASHBOARD

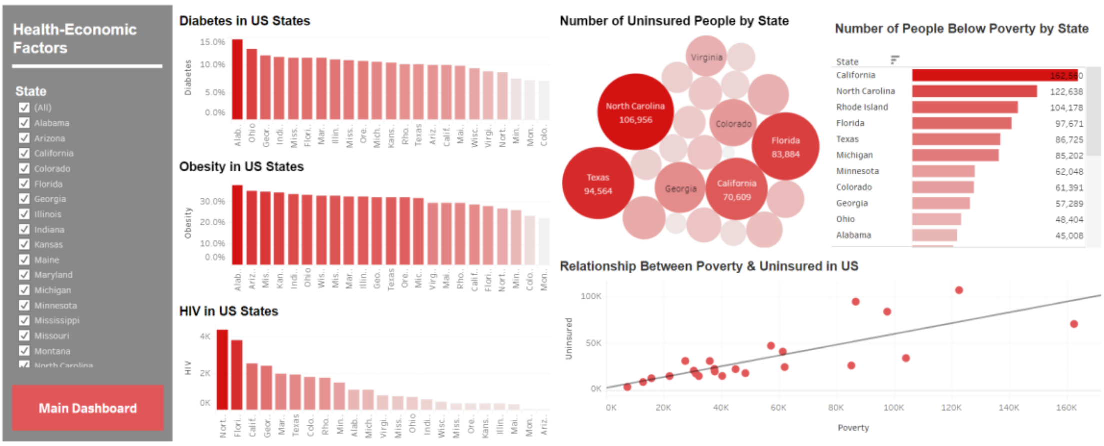
The main dashboard provides an overview of the COVID-19 pandemic in the United States. It shows the total number of cases and deaths, as well as the number of cases and deaths in each state. It also shows how the number of cases and deaths has changed over time.



This information can be used to track the progress of the pandemic and to identify areas where the virus is spreading rapidly. This information can also be used to compare the performance of different states in terms of their ability to control the spread of the virus.

HEALTH-ECONOMIC FACTORS DASHBOARD

The health economic factors dashboard provides more detailed information on the health factors that may be associated with the spread of COVID-19, such as obesity rates, diabetes rates, and HIV rates. It also shows us more detailed information about the number of people below the poverty line and the number of uninsured people in each state.

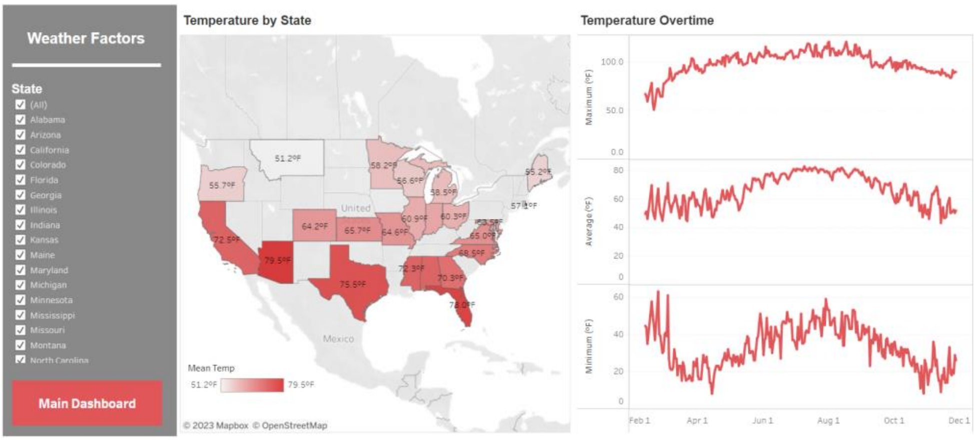


The information provided in this dashboard can be used to develop targeted interventions to prevent the spread of the virus and to protect people who are at high risk of serious illness because the dashboard shows that states with higher rates of poverty and uninsured people also tend to have higher rates of obesity, diabetes, and HIV. This suggests that there may be a relationship between poverty, a lack of health insurance, and the spread of COVID-19.

WEATHER FACTORS DASHBOARD

The weather factors dashboard provides more detailed information on temperatures that may be associated with the spread of COVID-19. According to scientists from

Aix-Marseille University in France, the COVID-19 virus is less stable in warm temperatures and can be killed by heat. This means that the virus is less likely to spread in hot weather.



However, the number of cases in states with higher temperatures is even more than those with lower ones. This is because the socioeconomic conditions in areas with higher temperatures in America tend to be lower. This means that people in these areas are more likely to live in crowded housing, have difficulty accessing healthcare, and be uninsured not like the areas in the Pacific Northwest, the Midwest, and the East Coast even though these areas have lower temperatures.

E. GITHUB

I created a repository on github for this project and it can be accessed using this link [Github Repo]