1/17/22
The Analytical Alliance
Zoe Aiello
Violeta Lio King

The Effect of Vaccination Status on SARS-CoV-2 Transmission

Problem Statement

The problem of vaccine hesitancy has the impact of reducing people's willingness to get vaccinated which allows SARS-CoV-2 to spread, so a good starting point would be to evaluate the relationship between vaccination status and SARS-CoV-2 transmission rate.

Obtain

COVID-19 and vaccine hesitancy have transformed the world over the past three years. The purpose of this study is to investigate the relationship between vaccination status and COVID-19 case data. The CDC provides public access to COVID-19 and vaccination tracking data through their website data.cdc.gov. Two datasets were chosen to fulfill the key goal of this investigation: one which tracked vaccination status by county, and another which tracked case and death counts by state (Data is split up across multiple files to begin with: 1pt). The first dataset records various case and death measurements for COVID-19 in all states every 2 weeks. The information of interest in this dataset was the confirmed cases for each state. The second dataset recorded various vaccination information by county in the US including the percentage of the population with at least one vaccination dose, the percentage that has completed the vaccination series, and the percentage with a booster (Data set is composed of more than one type of related data: 2pts, A county is related to one state, a state is related to many counties). The information of interest in this dataset was the percent of the population with at least one dose. Both of these datasets were obtained from the CDC website by making HTTP requests through an API (Data set needs to be accessed in a way other than connecting to a database or downloading a file: 1pt). The datasets came in a JSON format and were converted into dataframes for preparation (4 point dataset).

Scrub

To prepare the data for analysis, the two datasets were merged as follows. The COVID cases dataframe was parsed so that it only included the rows containing the submission date, state, and confirmed case data and any rows that contained missing values were dropped from the dataframe. The vaccination dataset only included data for January 22, 2022. But the COVID cases dataset did not include any data that was recorded on that exact date, so confirmed cases that were recorded in January 2022 were used as comparison for the vaccination data. For states that had more than one submission for January 2022 (maximum of 3 submissions for one state), the average of the confirmed cases was used to represent the states COVID cases for January. Since the vaccination dataframe already only included values from January 2022, the dataset was

parsed to only include the columns containing the state and percent of the population with at least one vaccination dose. Rows with missing values were dropped from the dataframe. Since the data was taken by county, the percent population with at least one vaccination dose was averaged, grouping by state so that there would be one data point for each state. Finally, the two dataframes were merged on "state" so that for each state there was one value for confirmed cases and one value for percent of population with at least one dose. After all missing values were removed and all non-January-2022 submissions were removed, there was data available for 12 states.

Once the dataframe had been prepared, it was converted to CSV so that it could be analyzed in R. Code was written to export the dataframe to JSON format and write it into a file. The file conversions and writing was accomplished using the Pandas library in Python.

Explore

The CSV file that was generated was uploaded into R to conduct an exploration of the data. A plot was created to compare confirmed COVID cases and percent vaccination for each state (Figure 1). The plot depicts twelve points - one for each state that had data for January 2022. The x-axis represents the percent of population with at least one dose of the vaccine and the y-axis represents the number of confirmed cases of COVID-19. Based on the plot, there appeared to be a potential positive linear relationship between the two parameters and more investigation was needed to assess its appropriateness for a Simple Linear Regression (SLR) model.

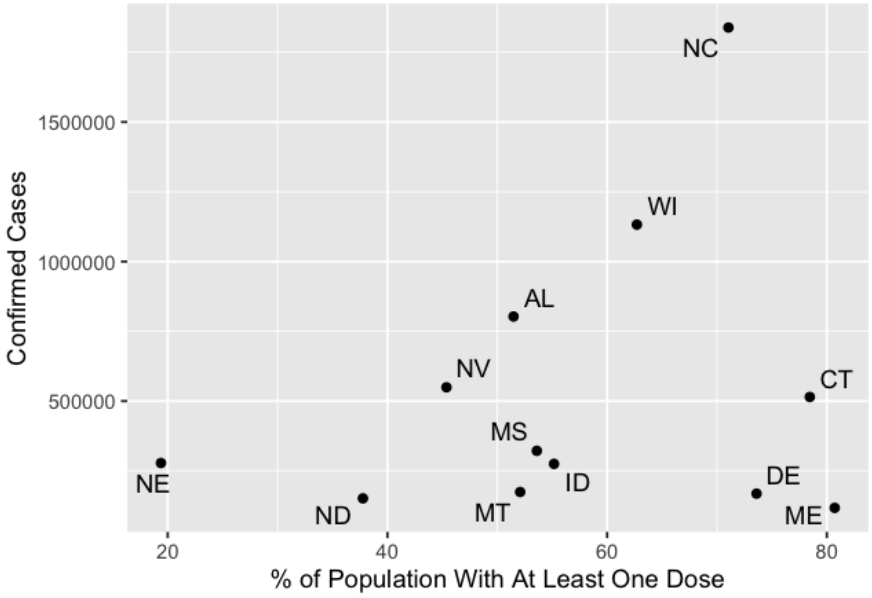


Figure 1. Twelve state's confirmed cases vs. percent of the population with at least one dose. Creation of plot proves that data was successfully loaded into R.

Summary statistics for the SLR were generated in R. The relevant values for the slope, intercept and the standard error were 3082, 351191, and 535400, respectively. In addition, a plot with the regression line and the confidence interval was also produced (Figure 2). Though initially promising, the wideness of the confidence intervals on the left and right seemed to suggest that the constant variance assumption was violated. The Simple Linear Regression model is not robust to violations of the constant variance assumption.

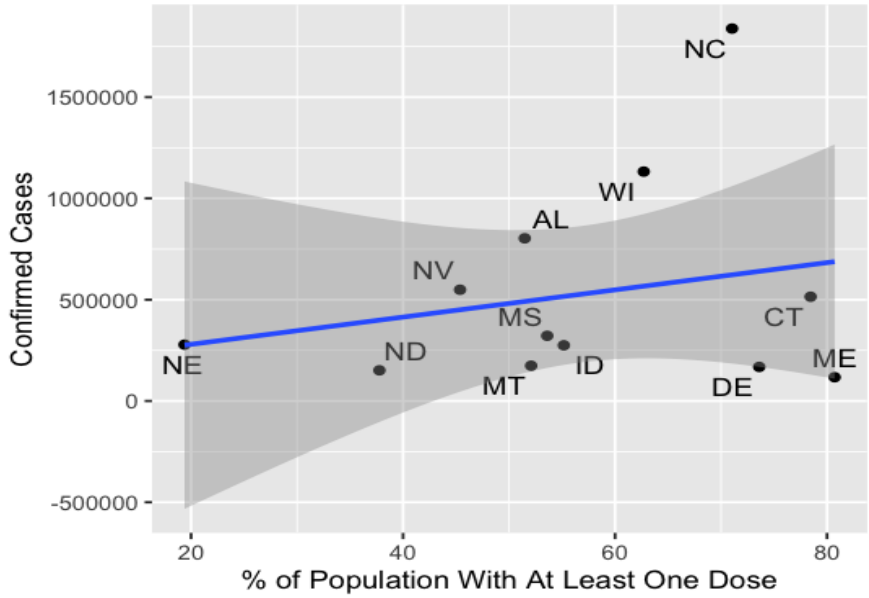


Figure 2. Twelve state's confirmed cases vs. percent of the population with at least one dose overlaid with the residual line (blue) and confidence intervals (shaded region). Creation of plot proves that data was successfully loaded into R.

Model

Since the constant variance assumption appeared to be violated, a lack-of-fit F-test was conducted on the data to test the linearity assumption. The data was fit to both the Simple Linear Regression model and the Separate Means model. If the fit of the Separate Means model is better, then it can be concluded that the relationship between the percent of the population with at least one vaccine dose and the number of confirmed cases is not linear. The two models are compared by an analysis of variance (ANOVA) test which is available as a function in R. The ANOVA comparison resulted in a p-value of zero meaning that, at a significance level of 0.05, there is strong evidence that the relationship between the two parameters is not linear.

Further modeling was performed with K-means to determine how the data grouped together. The fviz_nbclust() function from the factoextra package in R was used to determine that the optimal amount of clusters for the data was three. A k-means model was used on the standardized data; it separated the data into three clusters of sizes 7, 1, and 4 (Figure 3). The

smallest cluster of size = 1, included North Carolina which had the largest confirmed cases by far and about a mid-to-high vaccination percentage. The mid-sized cluster, with size = 4, included Wisconsin, Connecticut, Alabama, and Nevada. This cluster had both confirmed cases and percent vaccination that were fairly mid-range. The largest cluster which included Delaware, Idaho, Maine, Mississippi, Montana, North Dakota and Nebraska all had low confirmed case numbers but varied considerably in their percent vaccinated.

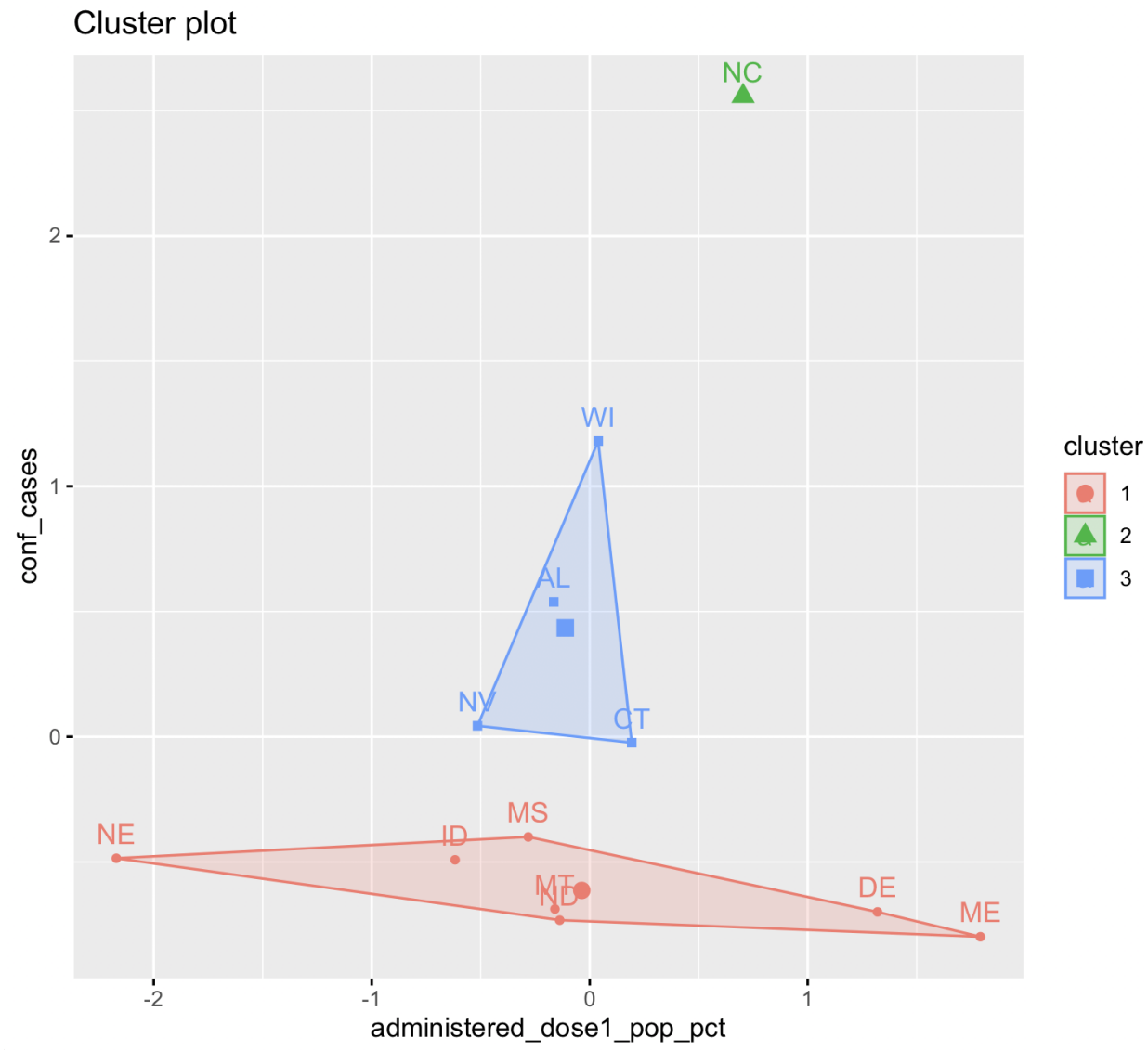


Figure 3. Clusters generated by the k-means model (3 clusters of sizes 1, 4, and 7). Creation of plot proves that data was successfully loaded into R.

Interpret

The Simple Linear Regression analysis estimated that for every 1% increase in population with at least one dose, there was an increase of 3082 cases. Additionally, it was estimated that

when 0% of the population had at least one dose, the number of cases would be 351191. The standard error was very high at +/- 535400 cases. However, the Simple Linear Regression was later found to be an inadequate model for the data. Though the exploration of the data initially suggested that there could be a positive linear relationship between the two parameters (Figure 1), the lack-of-fit F-test provided strong evidence that there is no linear relationship between the number of covid cases and the percentage of people with at least one vaccination dose (p-value = 0).

The k-means model confirmed that there did not seem to be a significant association between percent vaccination and confirmed cases in our data. The model separated the clusters by case numbers fairly consistently (Figure 3). However, while clusters 2 and 3 appeared to have similar percent vaccination rates, cluster 3 had lots of variability in its vaccination percentages. However, there are limitations to the analysis.

Confounding variables that influence the number of COVID-19 cases were not accounted for in our analysis. Population density is a confounding variable in our study that affects the spread of the virus. North Carolina, the state with the largest number of cases in our data and a fairly high vaccination rate, has a population density of 196.1 residents per square mile (US Census Bureau, QuickFacts) . Nebraska, the state with the lowest case numbers and the lowest vaccination rate in our data, has a population density of 23.8 residents per square mile (US Census Bureau, QuickFacts). Nebraska's low population density may account for its low case numbers despite its low vaccination rate. The public's attitude and adherence to CDC guidelines could also play a strong influence on the propagation of the virus. In addition, public policy and each state's individual approach to mitigating the spread of COVID-19 (such as lockdown measures and mask mandates) could be lending a much stronger influence than vaccination rates. Additional research must be conducted to quantify the effect of these confounding variables on the spread of COVID-19. Further research may help government agencies tasked with reducing the spread of COVID-19 to understand what causes vaccine hesitancy, what the effects of vaccine hesitancy are on COVID-19 transmission, and how to effectively quell fears about the vaccine.

Samples

1 page of initial COVID case data:

[{'submission_date': '2021-03-11T00:00:00.000', 'state': 'KS', 'tot_cases': '297229', 'conf_cases': '241035.0', 'prob_cases': '56194', 'new_case': '0.0', 'pnew_case': '0', 'tot_death': '4851', 'new_death': '0.0', 'pnew_death': '0', 'created_at': '2021-03-12T15:20:13.190', 'consent_cases': 'Agree', 'consent_deaths': 'N/A'}, {'submission_date': '2021-02-12T00:00:00.000', 'state': 'UT', 'tot_cases': '359641', 'conf_cases': '359641.0', 'prob_cases': '0', 'new_case': '1060.0', 'pnew_case': '0', 'tot_death': '1785', 'conf_death': '1729.0', 'prob_death': '56', 'new_death': '11.0', 'pnew_death': '2', 'created_at': '2021-02-13T14:50:08.565', 'consent_cases': 'Agree', 'consent_deaths': 'Agree'}, {'submission_date': '2021-03-01T00:00:00.000', 'state': 'CO', 'tot_cases': '438745', 'conf_cases': '411869.0', 'prob_cases': '26876', 'new_case': '677.0', 'pnew_case': '60', 'tot_death': '5952',

'conf_death': '5218.0', 'prob_death': '734', 'new_death': '1.0', 'pnew_death': '0', 'created_at': '2021-03-01T00:00:00.000', 'consent_cases': 'Agree', 'consent_deaths': 'Agree'}, {'submission_date': '2020-02-04T00:00:00.000', 'state': 'AR', 'tot_cases': '0', 'new_case': '0.0', 'tot_death': '0', 'new_death': '0.0', 'created_at': '2020-03-26T16:22:39.452', 'consent_cases': 'Not agree', 'consent_deaths': 'Not agree'}, {'submission_date': '2020-08-22T00:00:00.000', 'state': 'AR', 'tot_cases': '56199', 'new_case': '547.0', 'pnew_case': '0', 'tot_death': '674', 'new_death': '11.0', 'pnew_death': '0', 'created_at': '2020-08-23T14:15:28.102', 'consent_cases': 'Not agree', 'consent_deaths': 'Not agree'}, {'submission_date': '2020-07-17T00:00:00.000', 'state': 'MP', 'tot_cases': '37', 'conf_cases': '37.0', 'prob_cases': '0', 'new_case': '1.0', 'pnew_case': '0', 'tot_death': '2', 'conf_death': '2.0', 'prob_death': '0', 'new_death': '0.0', 'pnew_death': '0', 'created_at': '2020-07-19T00:00:00.000', 'consent_cases': 'Agree', 'consent_deaths': 'Agree'}, {'submission_date': '2020-08-12T00:00:00.000', 'state': 'AS', 'tot_cases': '0', 'new_case': '0.0', 'pnew_case': '0', 'tot_death': '0', 'new_death': '0.0', 'pnew_death': '0', 'created_at': '2020-08-13T14:12:28.259'}, {'submission_date': '2020-06-05T00:00:00.000', 'state': 'HI', 'tot_cases': '661', 'new_case': '8.0', 'pnew_case': '0', 'tot_death': '17', 'new_death': '0.0', 'pnew_death': '0', 'created_at': '2020-06-06T10:31:37.000', 'consent_cases': 'Not agree', 'consent_deaths': 'Not agree'}, {'submission_date': '2021-07-27T00:00:00.000', 'state': 'AK', 'tot_cases': '71521', 'new_case': '235.0', 'pnew_case': '0', 'tot_death': '377', 'new_death': '0.0', 'pnew_death': '0', 'created_at': '2021-07-28T13:51:39.509', 'consent_cases': 'N/A', 'consent_deaths': 'N/A'}, {'submission_date': '2021-12-27T00:00:00.000', 'state': 'CO', 'tot_cases': '896403', 'conf_cases': '820472.0', 'prob_cases': '75931', 'new_case': '10153.0', 'pnew_case': '543', 'tot_death': '10077', 'conf_death': '8922.0', 'prob_death': '1155', 'new_death': '3.0', 'pnew_death': '0', 'created_at': '2021-12-28T17:00:54.976', 'consent_cases': 'Agree', 'consent_deaths': 'Agree'}, {'submission_date': '2021-08-01T00:00:00.000', 'state': 'GA', 'tot_cases': '1187107', 'conf_cases': '937515.0', 'prob_cases': '249592', 'new_case': '3829.0', 'pnew_case': '1144', 'tot_death': '21690', 'conf_death': '18725.0', 'prob_death': '2965', 'new_death': '7.0', 'pnew_death': '0', 'created_at':

1 page of initial Vaccination data:

dose1_recip_5pluspop_pct': '42.1', 'administered_dose1_recip_12plus': '4596', 'administered_dose1_recip_12pluspop_pct': '46.2', 'administered_dose1_recip_18plus': '4299', 'administered_dose1_recip_18pluspop_pct': '48.8', 'administered_dose1_recip_65plus': '1322', 'administered_dose1_recip_65pluspop_pct': '61.9', 'series_complete_yes': '4027', 'series_complete_pop_pct': '34.1', 'series_complete_5plus': '4027', 'series_complete_5pluspop_pct': '36.4', 'series_complete_12plus': '3978', 'series_complete_12pluspop_pct': '40', 'series_complete_18plus': '3734', 'series_complete_18pluspop_pct': '42.4', 'series_complete_65plus': '1200', 'series_complete_65pluspop_pct': '56.2', 'booster_doses': '1277', 'booster_doses_vax_pct': '31.7', 'booster_doses_18plus': '1277', 'booster_doses_18plus_vax_pct': '34.2', 'booster_doses_50plus': '1047', 'booster_doses_50plus_vax_pct': '46.1', 'booster_doses_65plus': '683',

'booster_doses_65plus_vax_pct': '56.9', 'svi_ctgy': 'D', 'series_complete_pop_pct_svi': '14', 'series_complete_5pluspop_pct_svi': '14', 'series_complete_12pluspop_pct_svi': '15', 'series_complete_18pluspop_pct_svi': '15', 'series_complete_65pluspop_pct_svi': '15', 'metro_status': 'Metro', 'series_complete_pop_pct_ur_equity': '2', 'series_complete_5pluspop_pct_ur_equity': '2', 'series_complete_12pluspop_pct_ur_equity': '3', 'series_complete_18pluspop_pct_ur_equity': '3', 'series_complete_65pluspop_pct_ur_equity': '3', 'census2019': '11823', 'census2019_5pluspop': '11070', 'census2019_12pluspop': '9941', 'census2019_18pluspop': '8816', 'census2019_65pluspop': '2134'}, {'date': '2022-01-26T00:00:00.000', 'fips': '30019', 'mmwr_week': '4', 'recip_county': 'Daniels County', 'recip_state': 'MT', 'completeness_pct': '95.8', 'administered_dose1_recip': '753', 'administered_dose1_pop_pct': '44.6', 'administered_dose1_recip_5plus': '753', 'administered_dose1_recip_5pluspop_pct': '46.8', 'administered_dose1_recip_12plus': '742', 'administered_dose1_recip_12pluspop_pct': '51', 'administered_dose1_recip_18plus': '715', 'administered_dose1_recip_18pluspop_pct': '53.4', 'administered_dose1_recip_65plus': '304', 'administered_dose1_recip_65pluspop_pct': '63.6', 'series_complete_yes': '695', 'series_complete_pop_pct': '41.1', 'series_complete_5plus': '695', 'series_complete_5pluspop_pct': '43.2', 'series_complete_12plus': '688', 'series_complete_12pluspop_pct': '47.3', 'series_complete_18plus': '666', 'series_complete_18pluspop_pct': '49.8', 'series_complete_65plus': '292', 'series_complete_65pluspop_pct': '61.1', 'booster_doses': '321', 'booster_doses_vax_pct': '46.2', 'booster_doses_18plus': '321', 'booster_doses_18plus_vax_pct': '48.2', 'booster_doses_50plus': '268', 'booster_doses_50plus_vax_pct': '56.9', 'booster_doses_65plus': '187', 'booster_doses_65plus_vax_pct': '64.0', 'svi_ctgy': 'A', 'series_complete_pop_pct_svi': '3', 'series_complete_5pluspop_pct_svi': '3', 'series_complete_12pluspop_pct_svi': '3', 'series_complete_18pluspop_pct_svi': '3', 'series_complete_65pluspop_pct_svi': '3', 'metro_status': 'Non-metro', 'series_complete_pop_pct_ur_equity': '7', 'series_complete_5pluspop_pct_ur_equity': '7', 'series_complete_12pluspop_pct_ur_equity': '7', 'series_complete_18pluspop_pct_ur_equity': '7'

Initial merged dataframe

	state	administered_dose1_pop_pct	conf_cases
0	KS	52.477778	473910.0
1	MA	31.780000	1059963.0
2	MP	0.000000	3853.5
3	NYC	NaN	1604492.0
4	OK	49.618182	603113.6
5	PR	85.284615	254785.0
6	WV	60.233333	288786.0

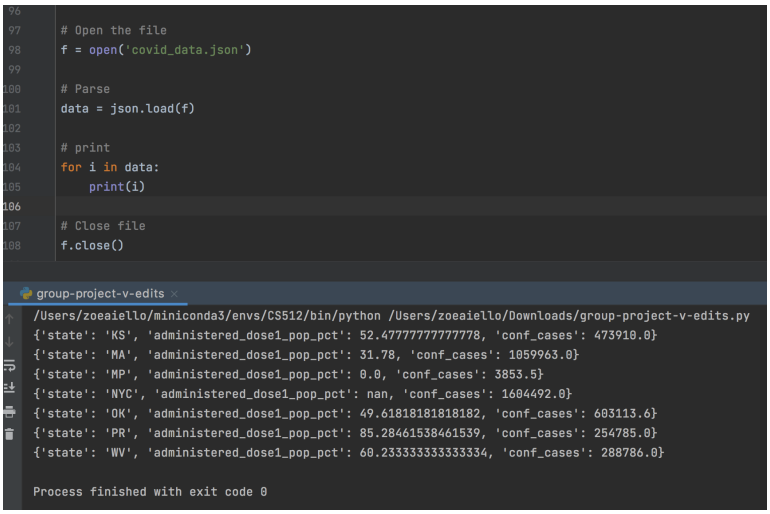
CSV Data

,state,administered_dose1_pop_pct,conf_cases
0,AL,49.01818181818181,771788.0
1,CA,60.888888888888886,7123571.0
2,DE,39.15,168161.0
3,ID,52.00769230769231,260286.0
4,IL,57.593103448275855,2837861.0
5,ME,83.65,117053.0
6,MS,55.09000000000001,348573.0
7,NC,68.01666666666664,1688664.5
8,ND,51.858823529411765,150314.5
9,NE,25.05483870967742,288653.5
10,NV,41.583333333333336,497084.0
11,WI,62.848148148148134,1102366.5

JSON Data

[{"state":"AL","administered_dose1_pop_pct":51.1294117647,"conf_cases":771788.0},{"state":"CA","administered_dose1_pop_pct":65.6047619048,"conf_cases":7123571.0},{"state":"DE","administered_dose1_pop_pct":null,"conf_cases":168161.0},{"state":"ID","administered_dose1_pop_pct":47.05625,"conf_cases":260286.0},{"state":"IL","administered_dose1_pop_pct":59.5933333333,"conf_cases":2837861.0},{"state":"ME","administered_dose1_pop_pct":78.2333333333,"conf_cases":117053.0},{"state":"MS","administered_dose1_pop_pct":50.9269230769,"conf_cases":348573.0},{"state":"NC","administered_dose1_pop_pct":69.26,"conf_cases":1688664.5},{"state":"ND","administered_dose1_pop_pct":52.4909090909,"conf_cases":150314.5},{"state":"NE","administered_dose1_pop_pct":15.8375,"conf_cases":288653.5},{"state":"NV","administered_dose1_pop_pct":61.25,"conf_cases":497084.0},{"state":"WI","administered_dose1_pop_pct":61.8523809524,"conf_cases":1102366.5}]

- **File parses without error**:



Relevant scripts with comments

Script used to prepare data in Python

```
import requests
import json
import html
import pandas as pd

# Get COVID case data
url = 'https://data.cdc.gov/resource/9mfq-cb36.json'
headers = {
  'X-App-Token': "AIccX8t5UlmYvloE9OdAxodtf",
  }
response = requests.get(url, headers=headers)
response_info = json.loads(response.text)

# Convert COVID case data into dataframe
results_df = pd.DataFrame.from_records(response_info)

# Get Covid vaccination data
url_2 = 'https://data.cdc.gov/resource/8xkx-amqh.json'
headers_2 = {
  'X-App-Token': "AIccX8t5UlmYvloE9OdAxodtf",
  }
response_vac = requests.get(url_2, headers=headers_2)
response_info_vac = json.loads(response_vac.text)

# Convert COVID case data into dataframe
results_df_vac = pd.DataFrame.from_records(response_info_vac)

# https://pandas.pydata.org/pandas-docs/version/0.8.1/indexing.html
# Filter out data that was not recorded in 2022
criterion = results_df['submission_date'].map(lambda x: x.startswith('2022'))
filtered_df = results_df[criterion]

# Parse columns of interest
# https://stackoverflow.com/questions/34682828/extracting-specific-
```

```
# selected-columns-to-new-dataframe-as-a-copy
parsed_df = filtered_df[['submission_date', 'state', 'conf_cases']]

# drop missing values
# https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.dropna.html
dropped_df = parsed_df.dropna()

# Since all data from the second dataframe is from January 2022, let's make a new dataframe
# with our information of interest (percent of the population with at least one dose and state). We
# will merge the two dataframes on state.

# create a dataframe out of the columns of interest
new_vac_df = results_df_vac[['recip_state', 'administered_dose1_pop_pct']]

# drop missing values
# https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.dropna.html
dropped_vac = new_vac_df.dropna()

# merge the two dataframes on 'state'
# https://stackoverflow.com/questions/33086881/merge-two-python-pandas-data-frames-of-
# different-length-but-keep-all-rows-in-out
final_df = dropped_df.merge(dropped_vac, how='left', left_on='state', right_on='recip_state')

# Change the percent administered dose column to numeric
# https://stackoverflow.com/questions/15891038/change-column-type-in-pandas
final_df['administered_dose1_pop_pct'] =
final_df['administered_dose1_pop_pct'].apply(pd.to_numeric)

# Average the percent administered dose per state
# https://stackoverflow.com/questions/30482071/how-to-calculate-mean-values-grouped-on-
# another-column-in-pandas
ave_final_df = final_df.groupby('state', as_index=False)['administered_dose1_pop_pct'].mean()

# Change the conf_cases column to numeric
final_df['conf_cases'] = final_df['conf_cases'].apply(pd.to_numeric)

# Average the conf_cases by state
conf_cases_df = final_df.groupby('state', as_index=False)['conf_cases'].mean()
```

```
# Merge the dataframes: ave_final_df includes the state and ave % of population with at least one
# dose and conf_cases_df includes the state and average confirmed cases per state. Merge on
# state
last_df_for_real = ave_final_df.merge(conf_cases_df, on='state')

#### CONVERT TO JSON ####
json_list = json.loads(json.dumps(list(last_df_for_real.T.to_dict().values())))

with open('covid_data.json', 'w') as fp:
  json.dump(json_list, fp)

#### CONVERT TO CSV ####
last_df_for_real.to_csv('covid_data.csv')

#### CONVERT FROM JSON TO CSV ####
json_df = pd.read_json('covid_data.json')
json_df.to_csv('converted_json.csv', index=False)

#### CONVERT FROM CSV TO JSON ####
csv_df = pd.read_csv('covid_data.csv')
csv_df.to_json('converted_csv.json')
```

Cluster Analysis Script

```
library(cluster)
library(mclust)
library(ggplot2)
covid <- read.csv(file = 'covid_data.csv')

# Create a data frame that only contains the dose column and confirmed cases
covid <- read.csv(file = 'covid_data.csv')
row.names(covid) <- covid$state
covid <- subset(covid, select = -c(state, X))

# https://uc-r.github.io/kmeans_clustering
# standardize data
covid_df <- scale(covid)

# Visualize data to get an idea of the type of cluster analysis to do
ggplot(covid, aes(x=administered_dose1_pop_pct, y=conf_cases)) +
```

```
  geom_point() +
  ggrepel::geom_text_repel(aes(label = state)) +
  labs(x = '% of Population With At Least One Dose', y ='Confirmed Cases')

#### K-MEANS ####
# https://search.r-project.org/CRAN/refmans/factoextra/html/fviz_nbclust.html
# Determine how many clusters is appropriate for K-means model
fviz_nbclust(covid_df, kmeans, method = "wss") +
  geom_vline(xintercept = 3, linetype = 2)

# Use K-means clustering to divide the whole group into three clusters
fit_kmeans <- kmeans(covid_df, 3)

# Visualize the clusters
plot(covid_df, col = fit_kmeans$cluster)
points(fit_kmeans$centers, col = 1 : 2, pch = 8, cex = 2)

# Another visualization, used in report
# https://uc-r.github.io/kmeans_clustering
fviz_cluster(fit_kmeans, data = covid_df)

#Checking the results
str(fit_kmeans)
cluster_kmeans <- fit_kmeans$cluster
```

Simple Linear Regression script

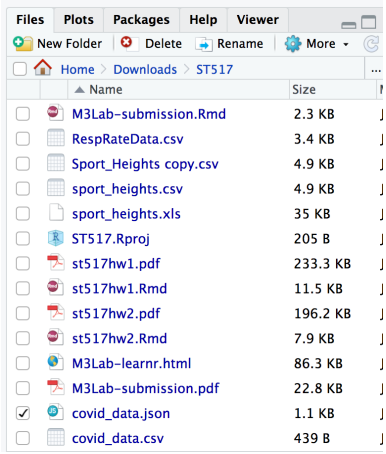
```
library("ggrepel")
library("rjson")
setwd("~/Downloads/ST517")
result <- read.csv(file = "covid_data.csv")
library(ggplot2)
library(backports)
library(broom)

# A plot of the data
ggplot(result, aes(x=administered_dose1_pop_pct, y=conf_cases)) +
  geom_point() +
  ggrepel::geom_text_repel(aes(label = state)) +
  labs(x = '% of Population With At Least One Dose', y ='Confirmed Cases')
```

```
# linear model, administered_dose1_pop_pct is the explanatory variable
slr_covid <- lm(conf_cases~administered_dose1_pop_pct, data = result)
summary(slr_covid)

# Estimate B0: 351191, estimate B1: 3082, estimate SE: 535400

# Replot points but this time with the regression line overlaid
ggplot(result, aes(x=administered_dose1_pop_pct, y=conf_cases)) +
  geom_point() +
  ggrepel::geom_text_repel(aes(label = state)) +
  labs(x = '% of Population With At Least One Dose', y ='Confirmed Cases') +
  geom_smooth(method = "lm")
# Constant variance assumption appears to be violated

# Evaluate the validity of the model using residuals
covid_diag <- augment(slr_covid)
qplot(conf_cases, .resid, data=covid_diag)
qplot(.fitted, .resid, data = covid_diag)

# First plot shows a systematic change in the variance across COVID cases
# Let's do a lack of fit F test
mod_sep_covid <- lm(log(conf_cases)~factor(administered_dose1_pop_pct), data = result)
mod_slr_covid <- lm(log(conf_cases)~ administered_dose1_pop_pct, data = result)
anova(mod_slr_covid, mod_sep_covid)

# p = 0, the SLR model is not adequate
```

Proof that data was successfully loaded into R:

- Although plots prove data was successfully loaded into R, here is some additional proof.
- No error was shown after running the code and files show that covid_data.json was loaded successfully:

```
> result_json <- fromJSON(file = "covid_data.json")
>
```



Division of Labor

The division of labor was created with the intent of letting both team members have an opportunity to be a part of multiple parts of the analysis. Both team members contributed to writing this report. Zoe was in charge of finding the data sets, preparing them for analysis, conducting the simple linear regression in R, and generating the associated plots. Violeta was in charge of coding the portions of the script that converted the data frame to a CSV file, the code that generated the JSON files and the portion that converted the CSV files to JSON and back again. The last conversion was not used in this project but was done with the intent to fulfill assignment requirements and could potentially come in useful in future projects. Finally, Violeta also carried out the cluster analysis using the K-means model and the dendrogram (not included in the report) and was able to identify potential areas of interest for future projects.

Sources:

- United States COVID-19 Cases and Deaths by State over Time | Data | Centers for Disease Control and Prevention
- COVID-19 Vaccinations in the United States,County | Data | Centers for Disease Control and Prevention
- US Census Bureau Quick Facts