

# COVID-19 Vaccination Rates and Google Search Data

Arianna Schmid and Stacey Frank

2021-12-14

## Contents

<b>Introduction</b>	<b>1</b>
Research Questions . . . . .	2
<b>Data Collection</b>	<b>2</b>
1. Google Trends and Keywords . . . . .	2
2. Vaccination Rates . . . . .	5
3. State-level Demographics . . . . .	6
<b>Analysis</b>	<b>7</b>
1. Correlation Analysis . . . . .	7
2. Regression Analysis . . . . .	9
<b>Conclusion</b>	<b>15</b>
<b>References</b>	<b>15</b>

## Introduction

Vaccines to control the Coronavirus Disease 2019 (COVID-19) became available to the public in the first half of 2021. Rejection and indecision towards being vaccinated is evident across the United States. The motivation for this study is to provide a better understanding of reasons for COVID-19 vaccine refusal in the United States. This can help public health messaging campaigns be more targeted and effective when promoting vaccination.

Google data is useful for exploring this topic because there is previous research that people feel freer to search for socially stigmatized topics on Google than they would be to admit such opinions in a survey or another form of data collection. In his study of racial animus's relationship to voting behavior, Stephens-Davidowitz notes that Google searchers are likely to be alone and thus more likely to search for private topics (Stephens-Davidowitz 2014). Since we are interested in understanding how myths and conspiracies are playing into people's choice to receive the vaccine or not, Google searches are a useful source of data about the prevalence and impact of these ideas.

## Research Questions

Our primary research question is what is the relationship, if any, between state-level COVID-19 vaccine rates and the types of Google searches that are made about vaccines in each state? To assist in answering that question, we have two secondary research questions:

1. Are vaccine myths more commonly searched for in states that also have low vaccination rates?
2. Does the relationship between COVID vaccine rates and Google searches change between June and September 2021?

We will use Google search data, state level COVID vaccination rate information from the Centers for Disease Control, and state-level demographic information to address these questions.

## Data Collection

### 1. Google Trends and Keywords

The Centers for Disease Control provides a list of the most common questions about the COVID-19 vaccine, including mainstream questions and myths (CDC 2021). Similarly, the Mayo Clinic provides information on the most common myths surrounding the vaccine (Anon 2021a). Using these two data sources, we constructed a list of 12 keyword search terms. It consists of the two general searches, “covid vaccine” and “covid vaccine near me,” five mainstream questions, such as “covid vaccine side effects,” and 5 myth-related searches such as “covid vaccine microchip.”

We created a vector called “k” to signify “keywords” and referenced this vector in later code to loop over the search terms for which we wanted to collect data. Note that each element in “k” will be renamed based on its index (hits.1, hits.2, ... hits.12) for code efficiency.

Table 1: Google Search Terms in Vector ‘k’

Search Term
covid vaccine
covid vaccine near me
covid vaccine safe
covid vaccine ingredients
covid vaccine pregnant
covid vaccine protect
covid vaccine side effects
covid vaccine microchip
covid vaccine dna
covid vaccine fetal
covid vaccine infertility
covid vaccine magnet

We are interested in studying the Google search hit rate for all 12 terms in “k” for all 50 states and Washington, D.C. Since vaccines were made available to states at different points in time, and the nature of people’s concerns about the vaccine may have changed over time, we pulled search hit rates from 3 different time periods: 1/1/21-9/20/21, 4/1/21-6/20/21, and 7/1/21-9/20/21. The gtrendsR package was used to work with Google Trends queries.

We created the custom function `get.hit.results()` to iterate through all the Google Trends queries we wanted to run without repeating code. The function takes in the date range as an argument, calls `gtrends()` once for

each of the 12 search terms, and returns the hit results by state based on the given date range. The hit rates by state for each search term are then extracted from the list initially generated by the `gtrendsr()` query, and saved into a single tibble containing the hit rates for all 12 search terms for the specified time period. The function is called 3 times, once for each date range, and the result is saved in the 3 objects `hits.results.jan`, `hits.results.june`, and `hits.results.sept`.

*#User-defined function to run all needed Google Trends queries*

```
get.hits.results <- function(date){
  for (i in 1:length(k)){
    new_frame <- paste("Keyword",i,sep = "")
    assign(new_frame, gtrends(k[i], geo = "US",
                              time = date, low_search_volume = T)
          )
  }

  hits_results <- Keyword1$interest_by_region %>%
    left_join(Keyword2$interest_by_region, by = "location") %>%
    left_join(Keyword3$interest_by_region, by = "location") %>%
    left_join(Keyword4$interest_by_region, by = "location") %>%
    left_join(Keyword5$interest_by_region, by = "location") %>%
    left_join(Keyword6$interest_by_region, by = "location") %>%
    left_join(Keyword7$interest_by_region, by = "location") %>%
    left_join(Keyword8$interest_by_region, by = "location") %>%
    left_join(Keyword9$interest_by_region, by = "location") %>%
    left_join(Keyword10$interest_by_region, by = "location") %>%
    left_join(Keyword11$interest_by_region, by = "location") %>%
    left_join(Keyword12$interest_by_region, by = "location") %>%
    as_tibble() %>%
    select(c(1,2,6,10,14,18,22,26,30,34,38,42,46))

  hits_results %<>% rename( hits.1 = hits.x,
                           hits.2 = hits.y,
                           hits.3 = hits.x.x,
                           hits.4 = hits.y.y,
                           hits.5 = hits.x.x.x,
                           hits.6 = hits.y.y.y,
                           hits.7 = hits.x.x.x.x,
                           hits.8 = hits.y.y.y.y,
                           hits.9 = hits.x.x.x.x.x,
                           hits.10 = hits.y.y.y.y.y,
                           hits.11 = hits.x.x.x.x.x.x,
                           hits.12 = hits.y.y.y.y.y.y)

  print(hits_results)
}

hits.results.jan <- get.hits.results("2021-01-1 2021-09-20")
hits.results.june <- get.hits.results("2021-04-1 2021-06-20")
hits.results.sept <- get.hits.results("2021-07-1 2021-09-20")
```

Next, we are interested in getting the count of states that actually have a Google Trends ranking for each search term. We do this to ensure we are using search terms that are popular enough to have generated a hit rate, which will allow us to actually conduct our analysis. We accomplish this through the user-created

get.search.terms() function. This function takes the hits results object from get.hit.results() as an argument, and returns the total number of states that have a reported hit rate for each search term in “k.” Again, the function is called 3 times for each time period, and results are saved to the objects search.terms.jan, search.terms.june, and search.terms.sept. As an example, we see in Table 6 that for April-June, “covid vaccine,” “covid vaccine near me,” and “covid vaccine side effects” have reported hit rates in all 50 states and Washington DC. On the other hand, “covid vaccine microchip” was only searched often enough to generate a hit rate in 17 states.

Table 2: January-September: Number of States with Hit Rates by Search

Variable Name	Search Term	Number of States with Hit Rate
hits.1	covid vaccine	51
hits.2	covid vaccine near me	51
hits.3	covid vaccine safe	51
hits.4	covid vaccine ingredients	51
hits.5	covid vaccine pregnant	50
hits.6	covid vaccine protect	44
hits.7	covid vaccine side effects	51
hits.8	covid vaccine microchip	23
hits.9	covid vaccine dna	48
hits.10	covid vaccine fetal	31
hits.11	covid vaccine infertility	48
hits.12	covid vaccine magnet	43

Table 3: April-June: Number of States with Hit Rates by Search

Variable Name	Search Term	Number of States with Hit Rate
hits.1	covid vaccine	51
hits.2	covid vaccine near me	51
hits.3	covid vaccine safe	50
hits.4	covid vaccine ingredients	47
hits.5	covid vaccine pregnant	45
hits.6	covid vaccine protect	40
hits.7	covid vaccine side effects	51
hits.8	covid vaccine microchip	17
hits.9	covid vaccine dna	43
hits.10	covid vaccine fetal	26
hits.11	covid vaccine infertility	44
hits.12	covid vaccine magnet	40

Table 4: July-September: Number of States with Hit Rates by Search

Variable Name	Search Term	Number of States with Hit Rate
hits.1	covid vaccine	51
hits.2	covid vaccine near me	51
hits.3	covid vaccine safe	49
hits.4	covid vaccine ingredients	49
hits.5	covid vaccine pregnant	49

Variable Name	Search Term	Number of States with Hit Rate
hits.6	covid vaccine protect	42
hits.7	covid vaccine side effects	51
hits.8	covid vaccine microchip	10
hits.9	covid vaccine dna	41
hits.10	covid vaccine fetal	28
hits.11	covid vaccine infertility	42
hits.12	covid vaccine magnet	23

## 2. Vaccination Rates

We now have data for our Google search hits from January-September, April-June, and July-September. Next, we are interested in finding the corresponding information on vaccination rates for each state. This data is acquired by using RSocrata to pull COVID vaccine data from the Centers for Disease Control website via their API. After pulling the data into R, we did some basic data cleaning, including converting variables to a numeric type and dropping rows for U.S. territories and federal entities for which we did not have corresponding Google Trends data. We also recoded the state variable, which previously used the two-letter postal code for state (AK, AZ, etc.), to use the full state name. This was done so that the CDC data could be joined with the Google Trends data by state.

After cleaning, the datasets `vax.June21` and `vax.Sept21` are created, where `vax.June21` will correspond to the Google Trends hit results from April-June, and `vax.Sept21` will correspond to January-September and July-September Google Trends hit results. We are focusing on the vaccination rates in June and September 2021 because June is around the time when vaccines became widely available to anyone who wanted one, while September 21, 2021 is the day before boosters for some at-risk groups were approved by the CDC. By doing this, we are restricting our analysis to the time before boosters were available. Our variables of interest are `series_complete_pop_pct`, which is the percentage of the population in each state that has completed their vaccine series, and `admin_per_100k`, which is the number of vaccines administered in each state per 100,000 people.

The tables below list the top 10 states by percent of the population fully vaccinated in June and September 2021.

Table 5: Top 10 States for % Population Vaccinated: June 2021

State	Percent Pop. Fully vaccinated	Administered Per 100K
Vermont	64.3	134607
Massachusetts	60.0	125936
Maine	59.9	117770
Connecticut	59.0	121188
Rhode Island	57.3	118098
New Hampshire	54.9	110353
New Jersey	54.6	111391
Maryland	53.8	110157
Washington	52.9	109782
New Mexico	52.5	109500

Table 6: Top 10 States for % Population Vaccinated: September 2021

State	Percent Pop. Fully vaccinated	Administered Per 100K
Vermont	69.0	142674
Connecticut	68.0	139362
Maine	67.8	133752
Massachusetts	67.4	140598
Rhode Island	67.1	135818
New Jersey	63.6	128346
Maryland	63.4	129903
New York	62.7	130789
Washington	62.6	130222
New Mexico	62.3	130771

### 3. State-level Demographics

We now have our search hit results and vaccine rates. Before studying any correlations, we consider certain state-level demographic factors that could potentially have an impact on our analysis. Data of interest includes state population counts, voter information, household income, age group, and race.

State population numbers were web scraped from Wikipedia using Selector Gadget (Anon 2021b). These state counts were then joined with the `vax.June21` and `vax.Sept21` frames created in Part 2. In addition, information on the Trump vote share in the 2020 presidential election for each state was downloaded from the Cook Political Report (Anon n.d.a). Unnecessary columns were deleted and columns of interest were renamed. The final spreadsheet was saved as `vote.xlsx`.

Median household income information from the 2018 Current Population Survey was downloaded from the U.S. Census Bureau website and saved as `med.income.xlsx` (United States Census Bureau n.d.). Unnecessary columns were deleted and columns of interest were selected: `state` (location) and `median income` (`med.income`).

Percent of state population by age group was pulled from the Kids Count Data Center (Anon n.d.b) The raw Excel data was downloaded. Columns from years 2011-2019 were deleted since we are focusing on the most recently available data, in this case 2020. The `spread()` function from the `tidyr` package was then used to reshape from long to wide so we can split up the categorical Age Group Variable into the 3 separate variables: `ages 18 to 24`, `ages 25 to 64`, and `ages 65 and over`. The final spreadsheet was saved as `percent.age.xlsx`.

Race data from the 2020 Current Population Survey was pulled as a csv file from the Kaiser Family Foundation website (Anon 2021c). It was then converted from CSV to `xlsx`, unnecessary columns were deleted, columns of interest were renamed, and reported percentages of `<.01` were replaced with `0` to make all cell values integers. Categories for American Indian/Alaska Native, Native Hawaiian/Other Pacific Islander, and Multiple Races were combined into a single ‘Other’ race category. This was saved as `State Race Data.xlsx`.

These four excel files were loaded into the FOCD Github public repository. They were then joined together in R and saved as the object “`cov`” (for covariates). This object was then joined with the two CDC vaccine rate data frames created in Part 2 (`vax.June21` or `vax.Sept21`).

Finally, the three `gtrends` datasets from Part 1 (`hits.results.jan`, `hits.results.june`, `hits.results.sept`) are joined with the updated `vax.June21` or `vax.Sept21`, depending on the dates the Trends are covering. A user-defined function called `join.gtrends.vaccine()` accomplishes this and saves the 3 final data sets as `Jan01.analysis`, `Sept21.analysis`, and `June21.analysis`. Now they are ready for analysis.

```
## This function joins the gtrends datasets with vaccine info datasets
```

```

join.gtrends.vaccine <- function (hits.results.month,vax.month){

  month.analysis <- vax.month %>%
    select(location,date, admin_per_100k, series_complete_pop_pct,
           pct.vote.rep, med.income, pct.18.to.24, pct.25.to.64, pct.65.over,
           pct.white, pct.black, pct.hispanic, pct.asian, pct.other.multiple) %>%
    full_join(hits.results.month, by = "location") %>%
    arrange(location)

  print(month.analysis)

}

Jan01.analysis <- join.gtrends.vaccine(hits.results.jan,vax.Sept21)
Sept21.analysis <- join.gtrends.vaccine(hits.results.sept,vax.Sept21)
June21.analysis <- join.gtrends.vaccine(hits.results.june,vax.June21)

```

## Analysis

### 1. Correlation Analysis

```

## This function pulls the correlations for all 3 data sets
get.correlations <- function(month.analysis){
  #Loop for correlations for each search term
  j <- c("hits.1", "hits.2",
        "hits.3", "hits.4",
        "hits.5", "hits.6",
        "hits.7", "hits.8",
        "hits.9", "hits.10",
        "hits.11","hits.12")

  correlations <- data.frame(estimate=numeric(26), p.value=numeric(26))

  for(i in 15:ncol(month.analysis)){
    test <- cor.test(month.analysis[, i], month.analysis$series_complete_pop_pct)
    correlations$estimate[i] = test$estimate
    correlations$p.value[i] = test$p.value
  }

  correlations %<>%
    slice_tail(n=12) %>%
    cbind(j,k) %>%
    relocate(estimate, p.value, .after = k)

  correlations %<>% rename(var_name = j, search = k)

  print(correlations)

}

```

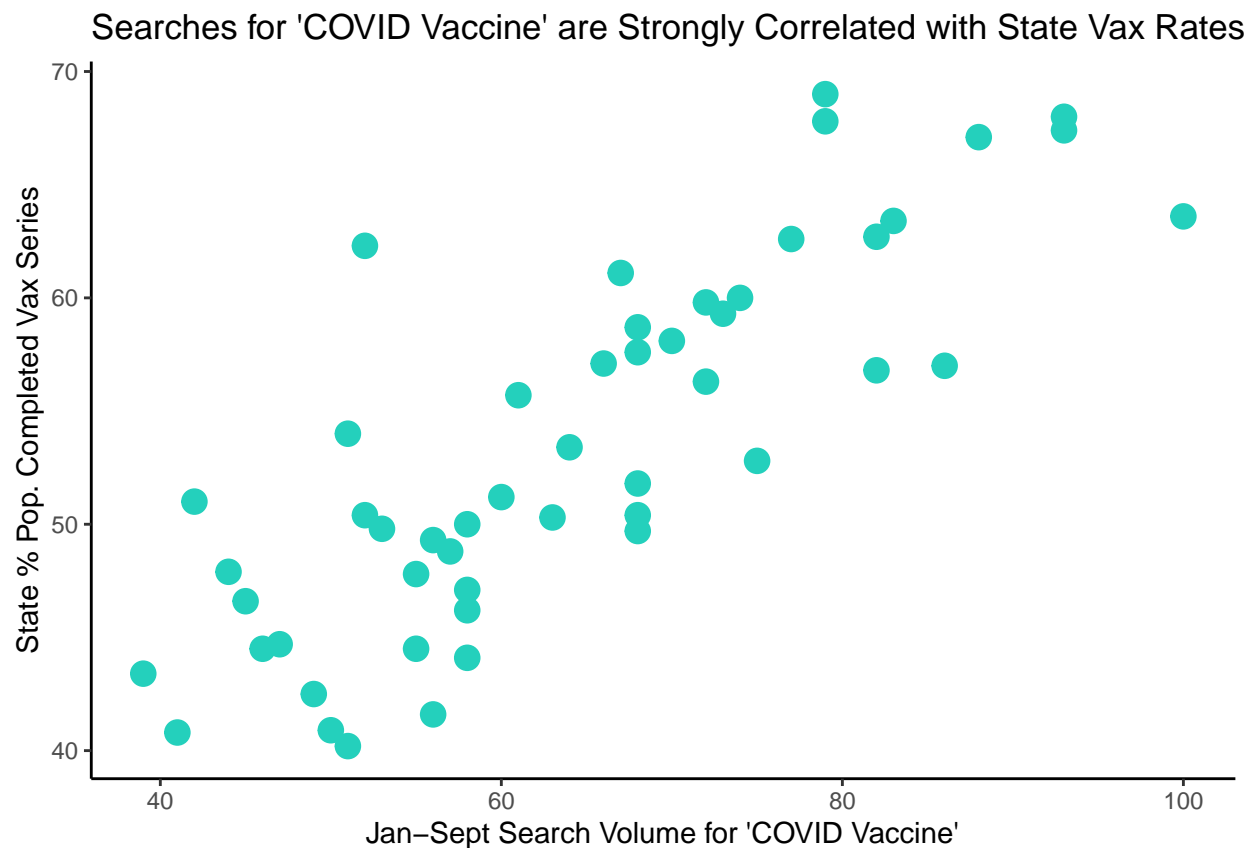
```
Jan01.correlations <- get.correlations(Jan01.analysis)
Sept21.correlations <- get.correlations(Sept21.analysis)
June21.correlations <- get.correlations(June21.analysis)
```

```
##Plotting of correlations
```

```
#Jan-Sept Searches
```

```
# using series_complete_pop_pct as measure for state vaccination rate
```

```
ggplot(Jan01.analysis) + geom_point(aes(hits.1, series_complete_pop_pct), color = '#24d0bc', size = 4) +
  labs(y = "State % Pop. Completed Vax Series", x = "Jan-Sept Search Volume for 'COVID Vaccine'") +
  ggtitle("Searches for 'COVID Vaccine' are Strongly Correlated with State Vax Rates") +
  theme_classic()
```



```
ggsave("covid.correlation.Jan.png")
```

```
## Saving 6.5 x 4.5 in image
```

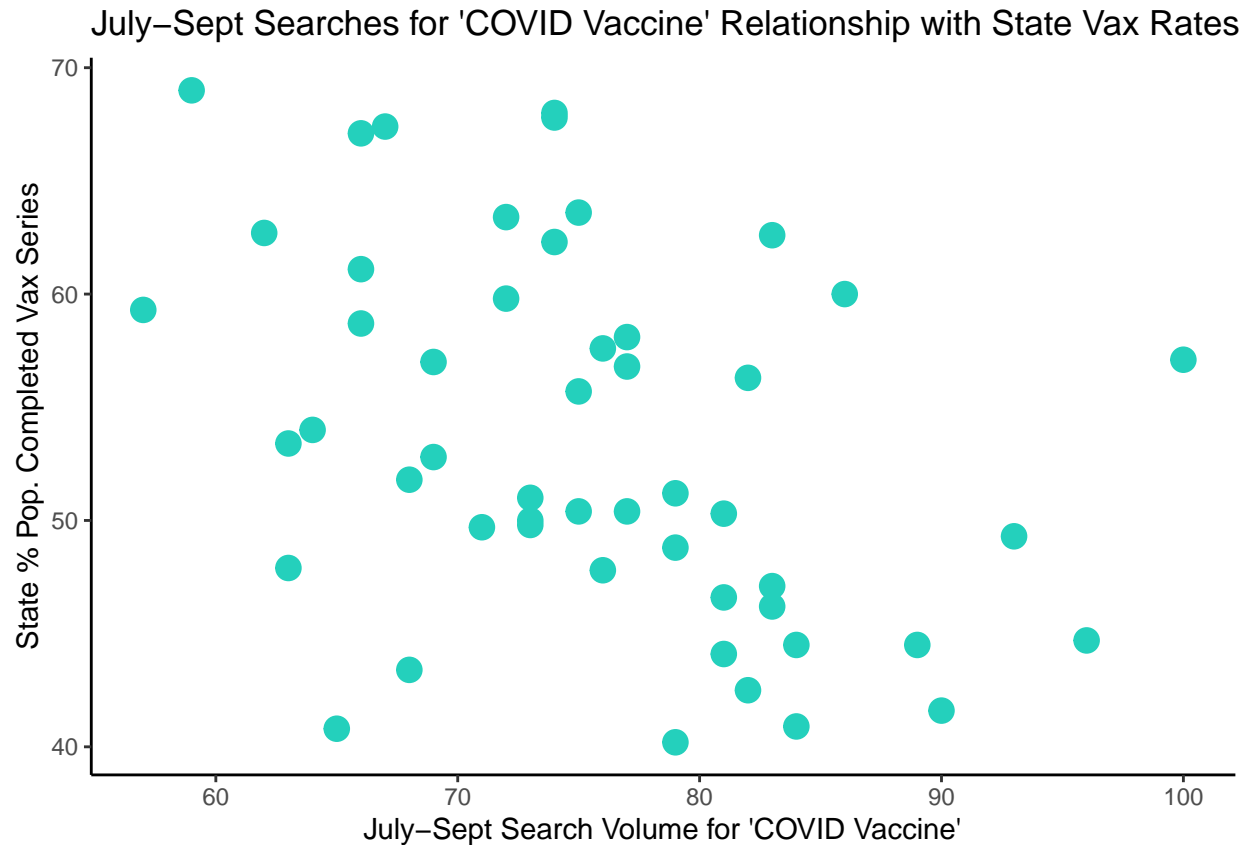
```
##Plotting of correlations continued
```

```
#July-September Searches
```

```
# using series_complete_pop_pct as measure for state vaccination rate
```

```
ggplot(Sept21.analysis) + geom_point(aes(hits.1, series_complete_pop_pct), color = '#24d0bc', size = 4) +
  labs(y = "State % Pop. Completed Vax Series", x = "July-Sept Search Volume for 'COVID Vaccine'") +
  ggtitle("July-Sept Searches for 'COVID Vaccine' Relationship with State Vax Rates") +
  theme_classic()
```





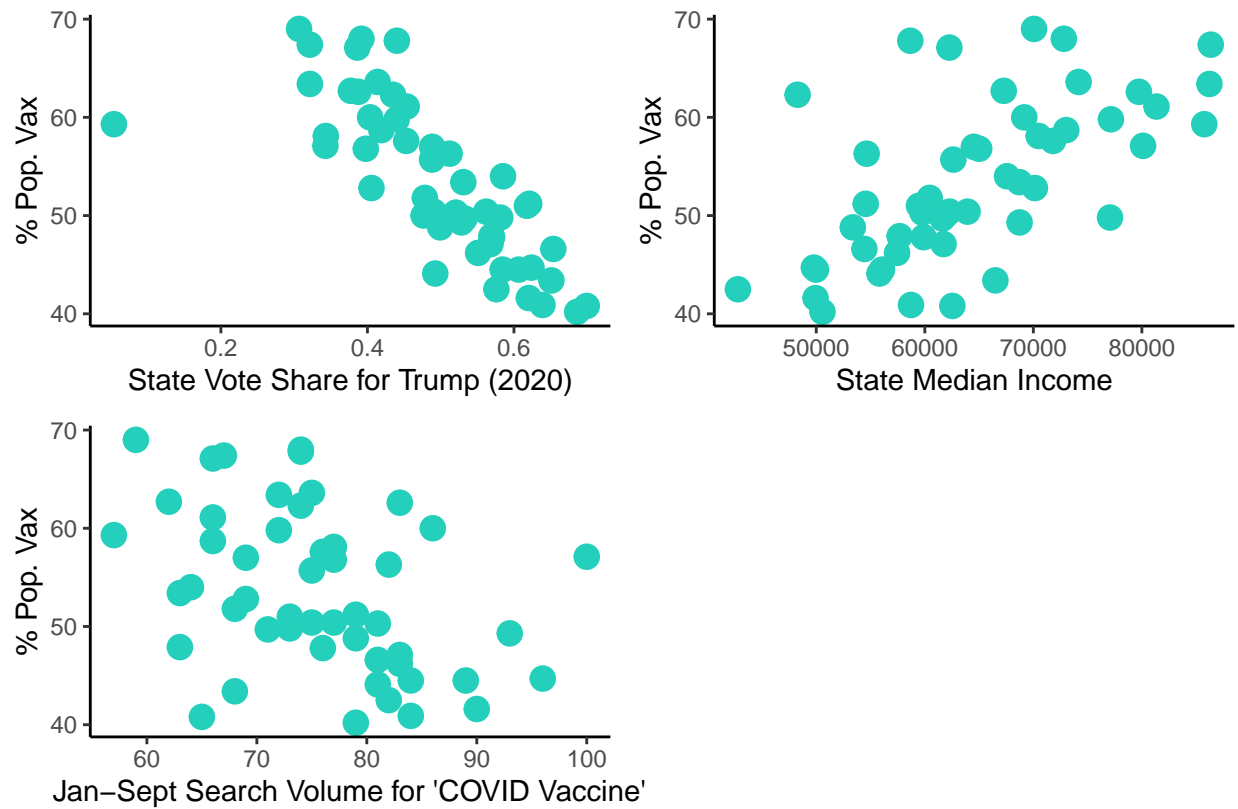
```
ggsave("covid.correlation.Sept.png")
```

```
## Saving 6.5 x 4.5 in image
```

## 2. Regression Analysis

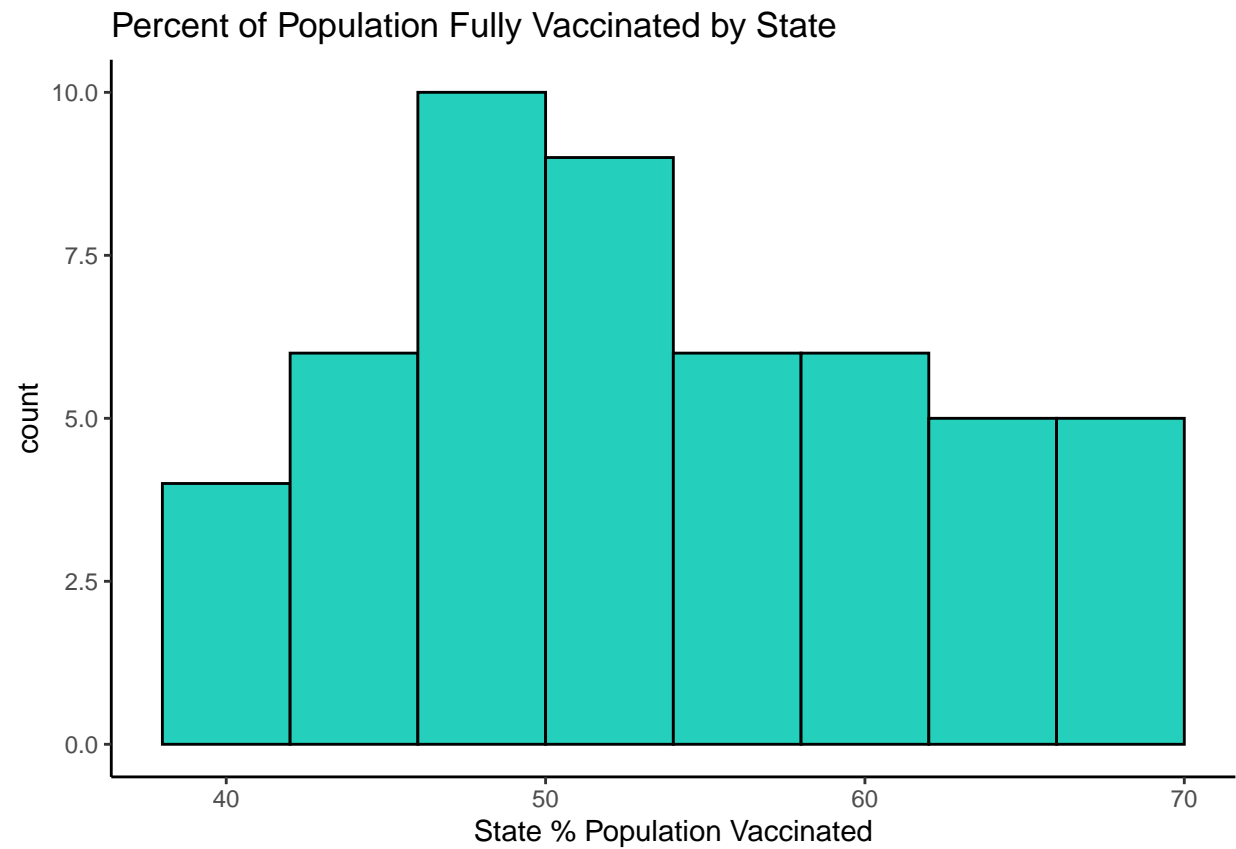
Given the lack of significant results in our correlation analysis, we next sought to use linear regression to determine the primary predictors of state vaccination rates.

Predictor Variables Associated with State Vaccination Rates



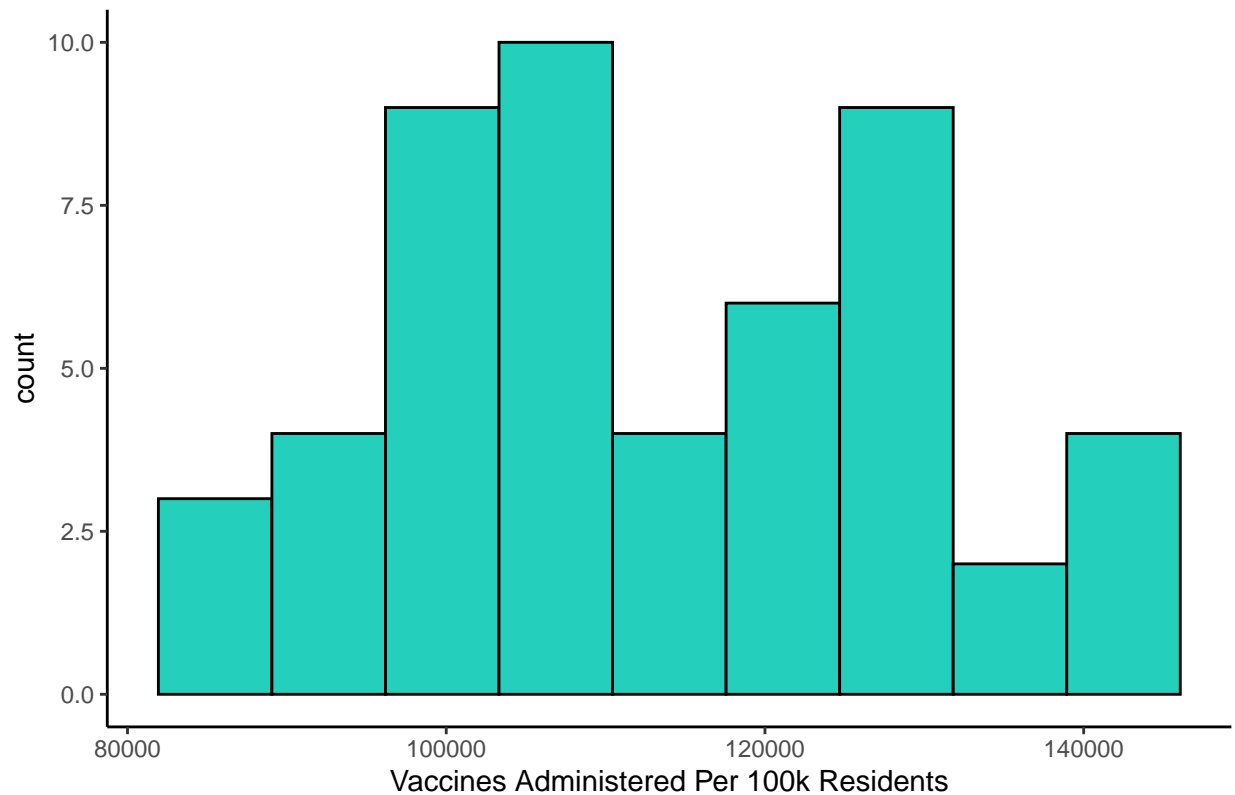
*##histogram of outcome variable*

```
ggplot(Jan01.analysis) + geom_histogram(aes(series_complete_pop_pct), color = "black", fill = '#24d0bc') +
  ggtitle("Percent of Population Fully Vaccinated by State") +
  labs(x = "State % Population Vaccinated") +
  theme_classic()
```



```
ggplot(Jan01.analysis) + geom_histogram(aes(admin_per_100k), color = "black", fill = '#24d0bc', binwidth = 5) +  
  ggtitle("Vaccines Administered Per 100,000 Residents by State") +  
  labs(x = "Vaccines Administered Per 100k Residents") +  
  theme_classic()
```

Vaccines Administered Per 100,000 Residents by State



```
##Linear model
```

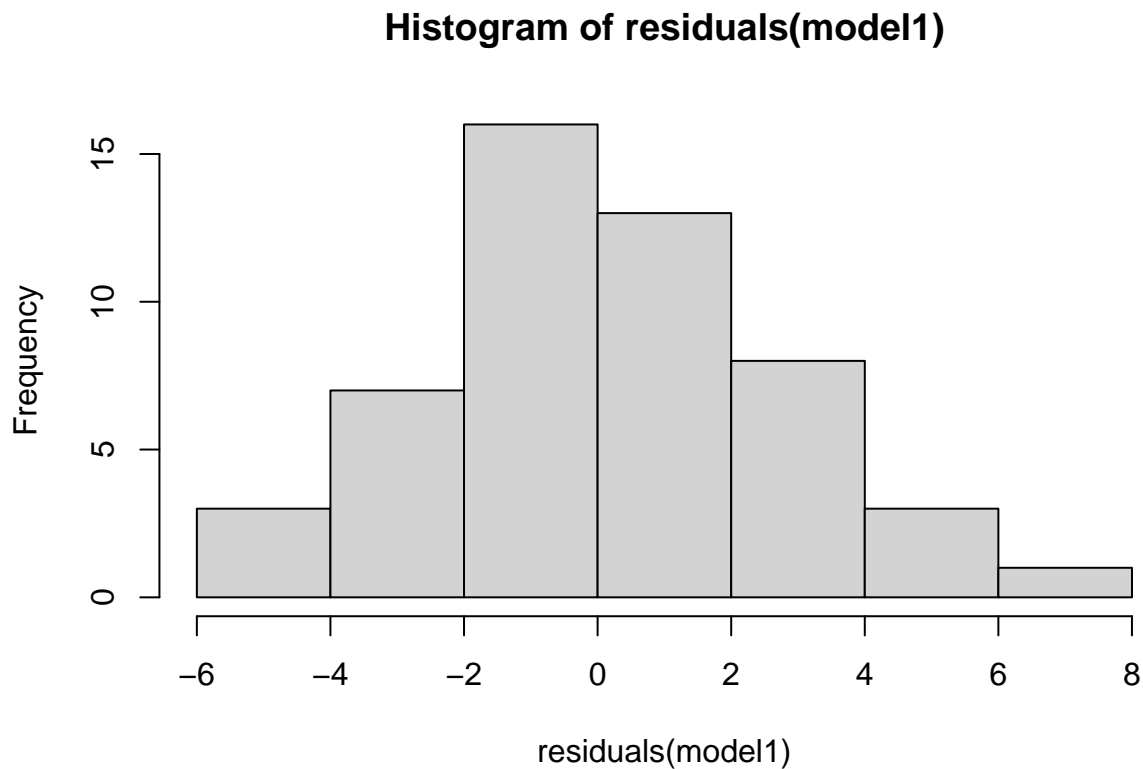
```
model1 <- lm(series_complete_pop_pct ~ pct.vote.rep + pct.white + pct.black + hits.1 + hits.2 + hits.3 + hits.4 + hits.5 + hits.6 + hits.7 + hits.8 + hits.9 + hits.10 + hits.11 + hits.12, data = Jan01.analysis)
summary(model1)
```

```
##
## Call:
## lm(formula = series_complete_pop_pct ~ pct.vote.rep + pct.white +
##     pct.black + hits.1 + hits.2 + hits.3 + hits.4 + hits.5 +
##     hits.6 + hits.7 + hits.8 + hits.9 + hits.10 + hits.11 + hits.12,
##     data = Jan01.analysis)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -5.7127 -1.6852 -0.2653  1.9016  7.6727
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.577e+01  6.716e+00   8.303 8.67e-10 ***
## pct.vote.rep -2.851e+01  9.199e+00  -3.099 0.003814 **
## pct.white    -1.523e+00  5.263e+00  -0.289 0.774004
## pct.black    -2.223e+01  6.187e+00  -3.593 0.000995 ***
## hits.1         3.474e-01  8.097e-02   4.290 0.000134 ***
## hits.2        -8.337e-02  6.730e-02  -1.239 0.223675
## hits.3        -8.722e-02  5.140e-02  -1.697 0.098609 .
##
```

```
## hits.4      -7.262e-03  3.543e-02  -0.205  0.838762
## hits.5       5.730e-02  4.619e-02   1.241  0.222954
## hits.6      -4.797e-02  4.233e-02  -1.133  0.264723
## hits.7       4.173e-04  8.533e-02   0.005  0.996126
## hits.8       5.368e-03  2.026e-02   0.265  0.792645
## hits.9       1.086e-02  3.036e-02   0.358  0.722698
## hits.10      -4.966e-03  2.417e-02  -0.205  0.838416
## hits.11       3.135e-02  4.718e-02   0.664  0.510760
## hits.12       2.003e-02  2.701e-02   0.742  0.463288
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.261 on 35 degrees of freedom
## Multiple R-squared:  0.8867, Adjusted R-squared:  0.8381
## F-statistic: 18.26 on 15 and 35 DF,  p-value: 2.981e-12
```

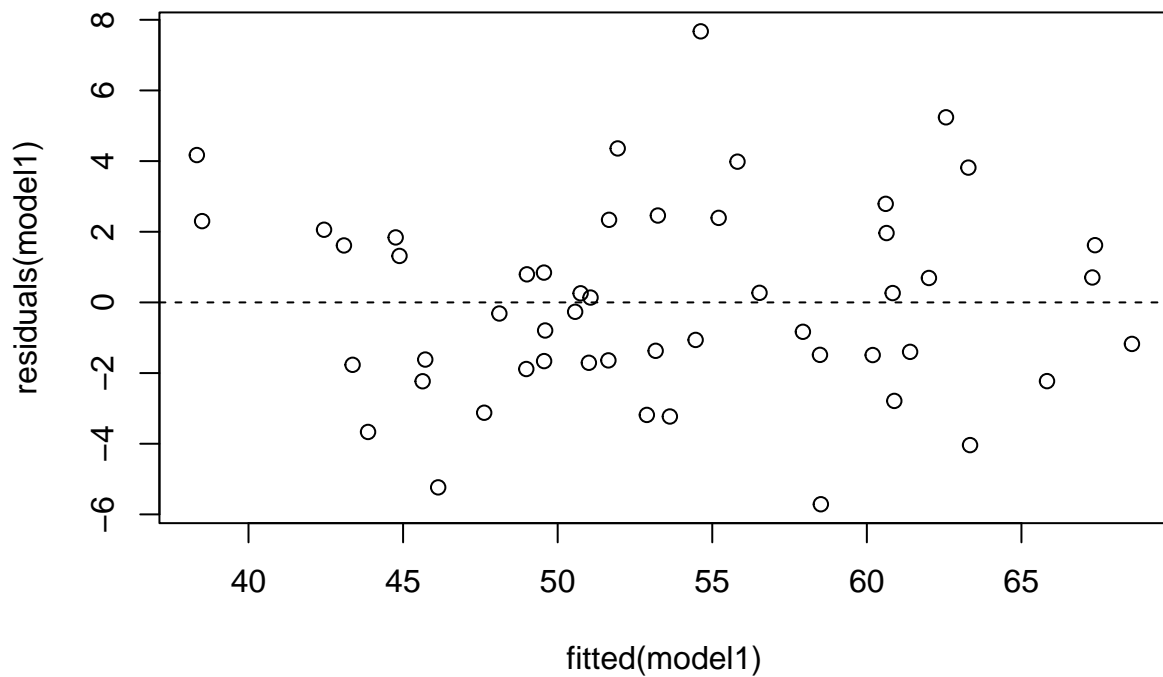
```
#Check that residuals are normally distributed
```

```
hist(residuals(model1))
```



```
#Check for homoskedasticity in residual variances (looks ok)
```

```
plot(fitted(model1), residuals(model1))
abline(h = 0, lty = 2)
```



```
#Linear model with interaction
#When adding interaction between hits.1 and % who voted republican, the main effects and the interaction

model2 <- lm(series_complete_pop_pct ~ pct.vote.rep + pct.black + hits.1 + hits.1*pct.vote.rep, data = 
summary(model2)

##
## Call:
## lm(formula = series_complete_pop_pct ~ pct.vote.rep + pct.black +
##     hits.1 + hits.1 * pct.vote.rep, data = Jan01.analysis)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -6.5878 -2.1744 -0.2679  2.5307  7.9335
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    52.3502    12.8896   4.061 0.000188 ***
## pct.vote.rep   -25.7923    23.4115  -1.102 0.276326
## pct.black      -25.3249     4.7632  -5.317 3.01e-06 ***
## hits.1          0.3237     0.1778   1.821 0.075133 .
## pct.vote.rep:hits.1 -0.1366     0.3496  -0.391 0.697849
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 3.381 on 46 degrees of freedom
## Multiple R-squared:  0.84, Adjusted R-squared:  0.826
## F-statistic: 60.36 on 4 and 46 DF,  p-value: < 2.2e-16
```

```
save.image(file = "shared_work_space.RData")
```

## Conclusion

## References

- Anon. 2021a. "COVID-19 Vaccine Myths Debunked." *Mayo Clinic Health System*.
- Anon. 2021b. "List of U.S. States by Population." *Wikipedia, the Free Encyclopedia*.
- Anon. 2021c. "Population Distribution by Race/Ethnicity (CPS)." *Kaiser Family Foundation*.
- Anon. n.d.a. "2020 Popular Vote Tracker." *Cook Political Report*.
- Anon. n.d.b. "Adult Population by Age Group." *KIDS COUNT Data Center*.
- CDC. 2021. "COVID-19 Vaccine Facts." *Centers for Disease Control and Prevention*.
- Stephens-Davidowitz, Seth. 2014. "The Cost of Racial Animus on a Black Candidate: Evidence Using Google Search Data." *Journal of Public Economics* 118:26–40. doi: [10.1016/j.jpubeco.2014.04.010](https://doi.org/10.1016/j.jpubeco.2014.04.010).
- United States Census Bureau. n.d. *Current Population Survey 2019 Annual Social and Economic (ASEC) Supplement*.