Evaluating US Socioeconomic Factors of COVID Confirmed Cases

Keywords: Web Scrapping, Linear Regression, COVID-19

Yunting Chiu, Chiyun Liu, Ana Lim

2021-04-27

Contents

Install Packages

Install the required libraries

- library(broom) # convert analysis objects from R into tidy tibbles
- library(tidyverse) # tidy data
- library(psych) # EDA

- library(countrycode) # get the continents
- library(lubridate) # adjust the date variable
- library(usmap) # find out the US states
- library(readxl) # read MS excel file
- library(corrplot) #for visualization of correlation
- library(leaps) # regsubsets
- library(car) # ncvTest
- library(corrplot) # multicollinearity plot
- library(performance) # multicollinearity table

Web Scraping COVID-19 Data

- The data source is from Johns Hopkins University. Here is the link

```
url_in <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covi
```

Data Scraping

```
+ time_series_covid19_confirmed_global.csv
+ time_series_covid19_deaths_global.csv
+ time_series_covid19_confirmed_US.csv
+ time_series_covid19_deaths_US.csv
```

```
df <- tibble(file_names = c("time_series_covid19_confirmed_global.csv",
                            "time_series_covid19_deaths_global.csv",
                            "time_series_covid19_confirmed_US.csv",
                            "time_series_covid19_deaths_US.csv")) -> df
```

Data Mapping

Using Regex for mapping

```
df %>%
  mutate(url = str_c(url_in, file_names, sep = "")) -> df
```

```
df %>%
  mutate(data = map(url, ~read_csv(., na = ""))) -> df
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   `Province/State` = col_character(),
##   `Country/Region` = col_character()
## )

## See spec(...) for full column specifications.

## Parsed with column specification:
## cols(
##   .default = col_double(),
##   `Province/State` = col_character(),
##   `Country/Region` = col_character()
## )

## See spec(...) for full column specifications.
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   iso2 = col_character(),
##   iso3 = col_character(),
##   Admin2 = col_character(),
##   Province_State = col_character(),
##   Country_Region = col_character(),
##   Combined_Key = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   iso2 = col_character(),
##   iso3 = col_character(),
##   Admin2 = col_character(),
##   Province_State = col_character(),
##   Country_Region = col_character(),
##   Combined_Key = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```
df %>%
  mutate(case_types = as.factor(str_extract(file_names, "[:alpha:]*_[gU][:alpha:]*"))) ->
  df
# alpha = Any letter, [A-Za-z]
# reference: https://www.petefreitag.com/cheatsheets/regex/character-classes/
```

```
df %>%
  dplyr::select(case_types, data) -> df
```

Clean Data

```
df %>%
  mutate(vars = map(df$data, names)) -> df
# map(df$vars, ~unlist(.)[1:15]) for checking

fix_names <- function(df, pattern, rePattern){
  stopifnot(is.data.frame(df), is.character(pattern), is.character(rePattern))
  names(df) <- str_replace_all(names(df), pattern, rePattern)
  return(df)
}

df %>%
  mutate(data = map(data, ~fix_names(., "([ey])/", "\\1_")),
         data = map(data, ~fix_names(., "Admin2", "County")),
         data = map(data, ~fix_names(., "Long_", "Long")),
         data = map_if(data, str_detect(df$case_types, "US"),
                    ~dplyr::select(., -c("UID", "iso2", "iso3",
                                          "code3", "FIPS", "Combined_Key"))),
         data = map_if(data, str_detect(df$case_types, "global"),
```

```r
                      ~mutate(., County = "NA")),
         data = map_if(data, !str_detect(df$case_types, "deaths_US"),
                      ~mutate(., Population = 0)),
         data = map(data, ~unite(., "Country_State",
                                 c("Country_Region", "Province_State"),
                                 remove = FALSE, na.rm = TRUE,
                                 sep = "_"))
         ) -> df

df %>%
  mutate(vars = map(df$data, names)) -> df # synchronize the vars correspondingly
# map(df$vars, ~unlist(.)) # for checking
```

```r
df %>%
  mutate(data = map(data, ~pivot_longer(data = ., cols = contains("/"),
                                        names_to = "Date",
                                        values_to = "dailyValues",
                                        names_transform = list(Date = mdy)))
         ) -> df
# df$data <- map(df$data, names) # synchronize the vars correspondingly
# map(df$vars, ~unlist(.)) # for checking

# crate a function to fix in type of Date
mdyDate <- function(df, varsDate){
  # stopifnot(is.data.frame(df), is.character(varsDate))
  df[[varsDate]] <- ymd(df[[varsDate]])
  return(df)
}

df %>%
  mutate(data = map(data, ~mdyDate(., "Date"))) -> df_long

# str(df_long) # check the data set
```

Add Continents and fix NAs

```r
df_long %>%
  mutate(data = map(data, ~mutate(., Continent = countrycode(Country_Region,
                                              origin = "country.name",
                                              destination = "continent")))
         ) -> df_long
```

```r
df_long %>%
  mutate(data = map(data, ~mutate(., Continent = case_when(
                                              Country_Region == "Diamond Princess" ~ "Asia",
                                              Country_Region == "Kosovo" ~ "Americas",
                                              Country_Region == "MS Zaandam" ~ "Europe",
                                              TRUE ~ Continent)
                                  ))) -> df_long

map(df_long$data, ~unique(.$Continent))
```

```
## [[1]]
## [1] "Asia"     "Europe"   "Africa"   "Americas" "Oceania"  NA
```

```
##
## [[2]]
## [1] "Asia"     "Europe"   "Africa"   "Americas" "Oceania"  NA
##
## [[3]]
## [1] "Americas"
##
## [[4]]
## [1] "Americas"
```

Unnest the Data Frames

```
# 1
df_long %>%
  unnest(cols = data) %>%
  ungroup() -> df_all

# 2
remove(df, df_long)

# 3
df_all %>%
  dplyr::select(-vars) -> df_all
```

Get World Population Data

- source: UN source

```
# 1
df_pop <- read_csv("../data/WPP2019_TotalPopulation.csv")
```

```
## Parsed with column specification:
## cols(
##   LocID = col_double(),
##   Location = col_character(),
##   PopTotal = col_double(),
##   PopDensity = col_double()
## )
```

```
# summarize(df_pop, across(everything(), ~sum(is.na(.)))) # check NAs

# 2
semi_join(df_pop, df_all, by = c("Location" = "Country_Region")) -> df_pop

# 3
df_pop %>%
  mutate(rank_p = rank(-PopTotal, na.last = TRUE),
         rank_d = rank(-PopDensity, na.last = TRUE),
         PopTotal = (PopTotal*1000)) -> df_pop
```

```
df_all %>%
  inner_join(df_pop, by = c("Country_Region" = "Location")) -> df_all
```

Tidy Data

- Because COVID-19 data is a time series data, we only focus on 2020/01/22 - 2021/01/22 for our experiment.

```
#df_all %>%
  #filter(case_types == "confirmed_US") %>%
  #select(Date, Province_State, County, dailyValues) %>%
  #tail()

# extract one year
df_all %>%
  filter(case_types == "confirmed_US" & as_date(Date) <= as_date("2021-01-22") | case_types == "deaths_U

names(covid)
```

```
##  [1] "case_types"     "Country_State"  "Province_State" "Country_Region"
##  [5] "Lat"            "Long"           "County"         "Population"
##  [9] "Date"           "dailyValues"    "Continent"      "LocID"
## [13] "PopTotal"       "PopDensity"     "rank_p"         "rank_d"
```

Find out each US state using usmap

```
state_map <- us_map(regions = "states")
state_map %>%
  distinct(full) %>%
  rename("Province_State" = "full") -> USstates
```

Obtain the number of confirmed cases for each state

```
covid %>%
  filter(case_types == "confirmed_US" & as_date(Date) == as_date("2021-01-22")) %>%
  dplyr::select(Province_State, County, dailyValues) %>%
  group_by(Province_State) %>%
  tally(dailyValues) %>%
  right_join(USstates) %>%
  rename("confirmed" = "n") -> confirmed
```

```
## Joining, by = "Province_State"
```

Obtain the number of death cases for each state

```
covid %>%
  filter(case_types == "deaths_US" & as_date(Date) == as_date("2021-01-22")) %>%
   dplyr::select(Province_State, County, dailyValues) %>%
  group_by(Province_State) %>%
  tally(dailyValues) %>%
  right_join(USstates) %>%
  rename("deaths" = "n") -> deathes
```

```
## Joining, by = "Province_State"
```

```
full_join(confirmed, deathes) -> covid
```

```
## Joining, by = "Province_State"
```

Read 2019 American community survey estimate by race by state

- Source: https://www.governing.com/now/State-Population-By-Race-Ethnicity-Data.html

```
race <- read_csv("../data/2019_state_community_by_race.csv")
```

```
## Parsed with column specification:
## cols(
##   State = col_character(),
##   `American Indian and Alaska Native alone` = col_double(),
##   `Asian alone` = col_double(),
##   `Black or African American alone` = col_double(),
##   `Native Hawaiian and Other Pacific Islander alone` = col_double(),
##   `Some other race alone` = col_double(),
##   `Total Population` = col_double(),
##   `Two or more races` = col_double(),
##   `White alone` = col_double()
## )
```

```
race %>%
  rename(Province_State = State) -> race

covid %>%
  left_join(race, by = "Province_State") %>%
  rename(American_Indian_and_Alaska_Native_alone = "American Indian and Alaska Native alone",
         Asian_alone = "Asian alone",
         Black_or_African_American_alone = "Black or African American alone",
         Native_Hawaiian_and_Other_Pacific_Islander_alone = "Native Hawaiian and Other Pacific Islander
         Some_other_race_alone = "Some other race alone",
         Total_Population = "Total Population",
         Two_or_more_races = "Two or more races",
         White_alone = "White alone") -> covid

#race %>%
  #anti_join(covidForRegression, by = "Province_State") #-->Check if there are some diff value of state

# The 2020 American Community Survey (ACS) 1-year estimates will be released on September 23, 2021.
# Since the 2020 survey does not yet release, we use the 2019 survey here
```

Read 2021 Household income

- This data is tided fromWorld Population Review, and the original, and source is from US Census

```
householdIncome2021 <- read_csv("../data/MedianHouseholdIncome2021.csv")
```

```
## Parsed with column specification:
## cols(
##   State = col_character(),
##   HouseholdIncome = col_double()
## )
```

```
householdIncome2021
```

```
## # A tibble: 51 x 2
##    State          HouseholdIncome
##    <chr>                    <dbl>
##  1 Maryland                 84805
##  2 New Jersey                82545
##  3 Hawaii                    81275
##  4 Massachusetts             81215
##  5 Connecticut               78444
##  6 Alaska                    77640
##  7 New Hampshire             76768
##  8 California                75235
##  9 Virginia                  74222
## 10 Washington                73775
## # ... with 41 more rows
```

```r
householdIncome2021  %>%
  rename(Province_State = State) -> householdIncome2021

covid %>%
  left_join(householdIncome2021, by = "Province_State") -> covid

# Recheck
#covid %>%
  #dplyr::select(Province_State, HouseholdIncome) %>%
   #dplyr::arrange(desc(HouseholdIncome))
```

Exploratory Data Analysis

Data Analysis and Visualization

```r
str(covid)
```

```
## tibble [51 x 12] (S3: tbl_df/tbl/data.frame)
##  $ Province_State                                  : chr [1:51] "Alabama" "Alaska" "Arizona" "Arkansa
##  $ confirmed                                       : num [1:51] 436087 52926 708041 281382 3147207 .
##  $ deaths                                          : num [1:51] 6486 254 12001 4549 36615 ...
##  $ American_Indian_and_Alaska_Native_alone         : num [1:51] 23265 115544 332273 17216 321112 ...
##  $ Asian_alone                                     : num [1:51] 66129 43678 241721 46078 5865435 ...
##  $ Black_or_African_American_alone                 : num [1:51] 1319551 22551 343729 467468 2282144
##  $ Native_Hawaiian_and_Other_Pacific_Islander_alone: num [1:51] 1892 9923 14168 12829 155871 ...
##  $ Some_other_race_alone                           : num [1:51] 74451 12602 364442 75590 5424558 ...
##  $ Total_Population                                : num [1:51] 4903185 731545 7278717 3017804 39512
##  $ Two_or_more_races                               : num [1:51] 91522 57476 280574 83603 1978145 ...
##  $ White_alone                                     : num [1:51] 3326375 469771 5701810 2315020 23484
##  $ HouseholdIncome                                 : num [1:51] 50536 77640 58945 47597 75235 ...
```

```r
covid %>% head()
```

```
## # A tibble: 6 x 12
##   Province_State confirmed deaths American_Indian~ Asian_alone Black_or_Africa~
##   <chr>              <dbl>  <dbl>            <dbl>       <dbl>            <dbl>
## 1 Alabama           436087   6486            23265       66129          1319551
## 2 Alaska             52926    254           115544       43678            22551
## 3 Arizona           708041  12001           332273      241721           343729
## 4 Arkansas          281382   4549            17216       46078           467468
## 5 California       3147207  36615           321112     5865435          2282144
## 6 Colorado          383008   5462            57578      188461           240538
## # ... with 6 more variables:
## #   Native_Hawaiian_and_Other_Pacific_Islander_alone <dbl>,
## #   Some_other_race_alone <dbl>, Total_Population <dbl>,
## #   Two_or_more_races <dbl>, White_alone <dbl>, HouseholdIncome <dbl>
```

Top 5 Confirmed Covid-19 Cases (Jan. 22, 2020 to Jan. 22, 2021)

```
covid %>%
   dplyr::select(1, 2) %>%
  arrange(desc(confirmed)) %>%
  head(5)
```

```
## # A tibble: 5 x 2
##   Province_State confirmed
##   <chr>              <dbl>
## 1 California       3147207
## 2 Texas            2227789
## 3 Florida          1627603
## 4 New York         1309403
## 5 Illinois         1093375
```

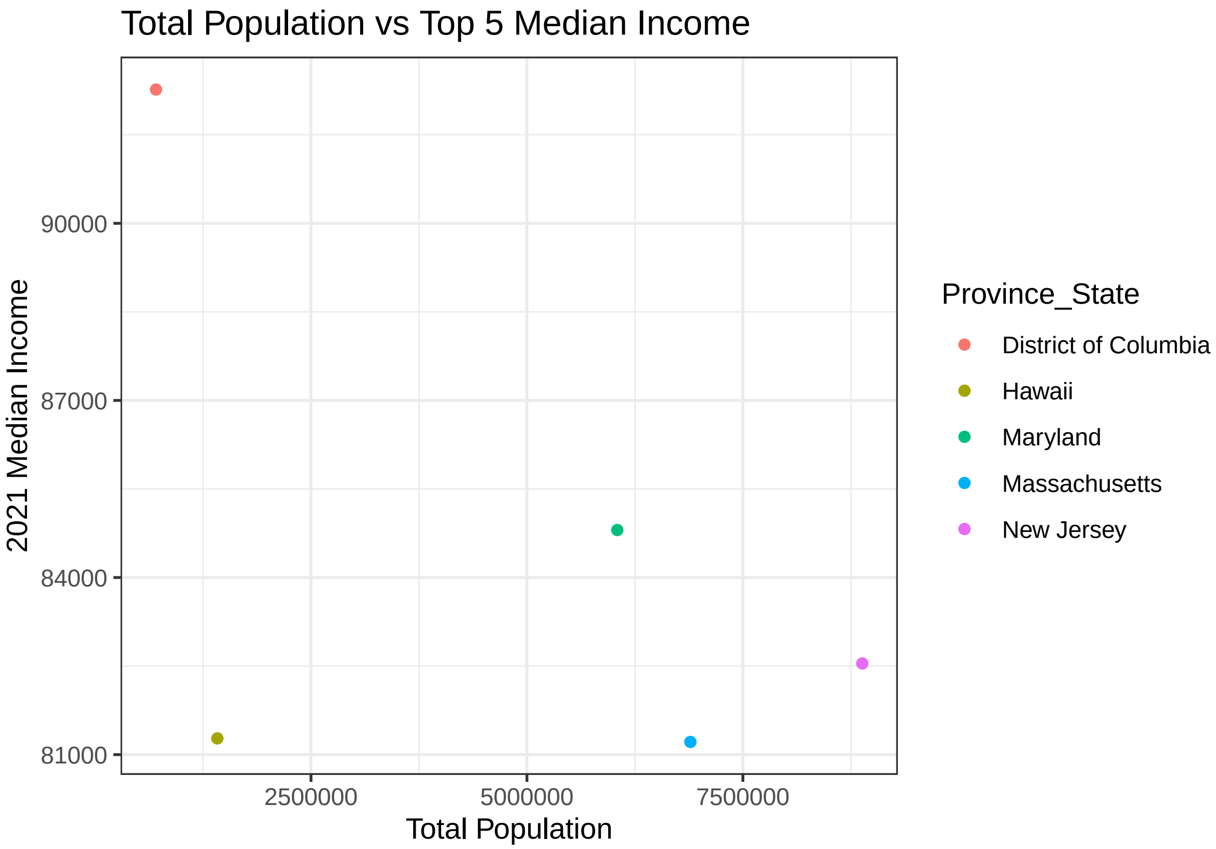
Top 5 Death Cases (Jan. 22, 2020 to Jan. 22, 2021)

```
covid %>%
   dplyr::select(1, 3) %>%
  arrange(desc(deaths)) %>%
  head(5)
```

```
## # A tibble: 5 x 2
##   Province_State deaths
##   <chr>           <dbl>
## 1 New York        41974
## 2 California      36615
## 3 Texas           34381
## 4 Florida         25011
## 5 New Jersey      20875
```

Total Population vs Top 5 Median Income

```
covid %>%
  dplyr::arrange(desc(HouseholdIncome)) %>%
  head(5) %>%
  ggplot(aes(x = Total_Population, y = HouseholdIncome, color = Province_State)) +
  geom_point() +
  theme_bw() +
  labs(x = "Total Population", y = "2021 Median Income",
       title = "Total Population vs Top 5 Median Income")
```



In order to run the linear regression model smoothly, we deleted the `Province State`variable from our covid data and saved it as `covidForRegression` dataframe. The reason of removing these two is :

- The `Province State` variable has 51 different categorical types, and each observation has its own type, making it difficult to run a regression model.
- We refer to the existing paper1 & 2 and select some similar independent variables as predictors.

```
covid %>%
  dplyr::select(-Province_State) -> covidForRegression
names(covidForRegression)
```

```
##  [1] "confirmed"
##  [2] "deaths"
##  [3] "American_Indian_and_Alaska_Native_alone"
##  [4] "Asian_alone"
##  [5] "Black_or_African_American_alone"
##  [6] "Native_Hawaiian_and_Other_Pacific_Islander_alone"
##  [7] "Some_other_race_alone"
##  [8] "Total_Population"
##  [9] "Two_or_more_races"
## [10] "White_alone"
## [11] "HouseholdIncome"
```

Descriptive Statistics

```
describe(covidForRegression)
```

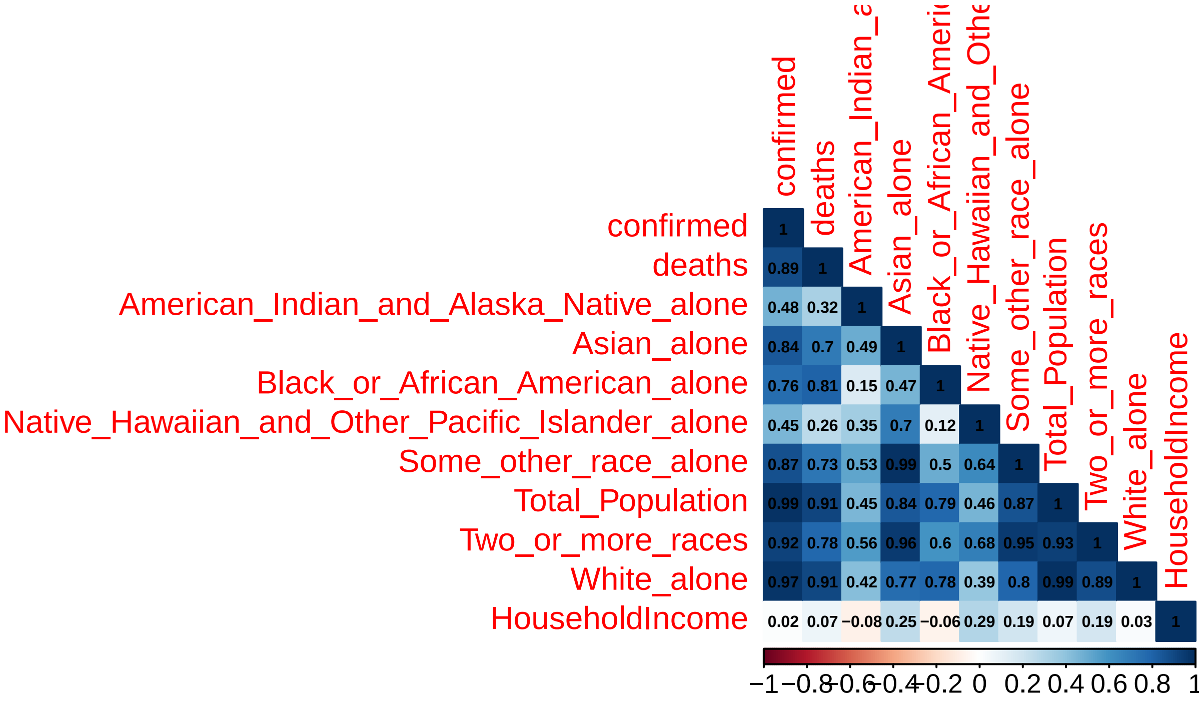
```
##                                                  vars  n       mean         sd
## confirmed                                           1 51  486333.45  569046.28
## deaths                                              2 51    8233.76    9596.98
## American_Indian_and_Alaska_Native_alone             3 51   55830.12   78344.82
## Asian_alone                                         4 51  365431.06  856969.06
## Black_or_African_American_alone                     5 51  823326.88  992012.64
## Native_Hawaiian_and_Other_Pacific_Islander_alone    6 51   12327.12   30362.26
## Some_other_race_alone                               7 51  320638.29  806711.25
## Total_Population                                    8 51 6436069.08 7360660.47
## Two_or_more_races                                   9 51  221743.04  305472.96
## White_alone                                        10 51 4636772.57 4879687.14
## HouseholdIncome                                    11 51   63212.49   10997.18
##                                                   median    trimmed        mad
## confirmed                                         336915  371693.90  341293.04
## deaths                                              5462    6321.95    5601.26
## American_Indian_and_Alaska_Native_alone            23860   37053.78   26287.98
## Asian_alone                                       125742  195764.44  163707.21
## Black_or_African_American_alone                   363167  638520.95  505434.65
## Native_Hawaiian_and_Other_Pacific_Islander_alone    3254    5152.68    3767.29
## Some_other_race_alone                             109373  148097.27  135321.35
## Total_Population                                 4467673 4941366.54 4167653.83
## Two_or_more_races                                 137034  162344.49  155352.76
## White_alone                                      3296909 3682867.05 3022032.51
## HouseholdIncome                                    61439   62515.10   10152.84
##                                                     min      max    range skew
## confirmed                                         10759  3147207  3136448 2.70
## deaths                                              169    41974    41805 1.86
## American_Indian_and_Alaska_Native_alone            1727   332273   330546 2.44
## Asian_alone                                        4633  5865435  5860802 5.27
## Black_or_African_American_alone                    7116  3553922  3546806 1.34
## Native_Hawaiian_and_Other_Pacific_Islander_alone    175   155871   155696 4.02
## Some_other_race_alone                              2153  5424558  5422405 5.14
## Total_Population                                 578759 39512223 38933464 2.53
## Two_or_more_races                                 15367  1978145  1962778 3.92
## White_alone                                      300058 23484958 23184900 2.14
## HouseholdIncome                                   45081    92266    47185 0.57
##                                                  kurtosis         se
## confirmed                                            8.62   79682.42
## deaths                                               3.05    1343.85
## American_Indian_and_Alaska_Native_alone              5.36   10970.47
## Asian_alone                                         30.44  119999.67
## Black_or_African_American_alone                      0.88  138909.55
## Native_Hawaiian_and_Other_Pacific_Islander_alone    15.79    4251.57
## Some_other_race_alone                               28.75  112962.17
## Total_Population                                     7.38 1030698.63
## Two_or_more_races                                   18.95   42774.77
## White_alone                                          4.90  683292.87
## HouseholdIncome                                     -0.46    1539.91
```

Linear Model Assumptions

Multicollinearity

Plot

```
correlation <- cor(covidForRegression)
corrplot(correlation, method="color", addCoef.col = "black", number.cex = 0.5, type = "lower")
```



Table

Check the inversely related values of Tolerance and VIF. **Tolerance has to be > 0.10 and VIF < 10.** If these stipulated are not fulfilled, multicollinearity is at hand.

```
reg <- lm(confirmed ~ ., data = covidForRegression)
check_collinearity(reg)
```

```
## Warning: Model matrix is rank deficient. VIFs may not be sensible.
```

```
## # Check for Multicollinearity
##
## Low Correlation
##
##                                               Term  VIF Increased SE Tolerance
##            American_Indian_and_Alaska_Native_alone 3.05         1.75      0.33
##   Native_Hawaiian_and_Other_Pacific_Islander_alone 4.43         2.11      0.23
##                                    HouseholdIncome 1.38         1.18      0.72
##
## Moderate Correlation
##
##                               Term  VIF Increased SE Tolerance
##                             deaths 9.23         3.04      0.11
##   Black_or_African_American_alone 5.05         2.25      0.20
```

```
##
## High Correlation
##
##                    Term    VIF Increased SE Tolerance
##             Asian_alone 136.39        11.68      0.01
##  Some_other_race_alone  87.55         9.36      0.01
##        Total_Population  84.30         9.18      0.01
##       Two_or_more_races 133.56        11.56      0.01
```

Remediation - Removing Highly Correlated Predictors

- Remove highly correlated predictors from the model. If you have two or more factors with a high VIF, remove one from the model.
- According to the result, we firstly remove the high correlation variables: `Some_other_race_alone`, `Total_Population`, and `Two_or_more_races`, and keep the majority of race variables.
- After we removed, now the independent variables have very low correlation with each other.

```
covidForRegression %>%
  dplyr::select(-Total_Population, -Two_or_more_races, -Some_other_race_alone) -> covidForRegression2
names(covidForRegression2)
```

```
## [1] "confirmed"
## [2] "deaths"
## [3] "American_Indian_and_Alaska_Native_alone"
## [4] "Asian_alone"
## [5] "Black_or_African_American_alone"
## [6] "Native_Hawaiian_and_Other_Pacific_Islander_alone"
## [7] "White_alone"
## [8] "HouseholdIncome"
```

```
reg <- lm(confirmed ~ ., data = covidForRegression2)
check_collinearity(reg)
```

```
## # Check for Multicollinearity
##
## Low Correlation
##
##                                               Term  VIF Increased SE Tolerance
##            American_Indian_and_Alaska_Native_alone 1.50         1.22      0.67
##                    Black_or_African_American_alone 3.64         1.91      0.27
##  Native_Hawaiian_and_Other_Pacific_Islander_alone 2.50         1.58      0.40
##                                    HouseholdIncome 1.26         1.12      0.80
##
## Moderate Correlation
##
##         Term  VIF Increased SE Tolerance
##       deaths 7.88         2.81      0.13
##  Asian_alone 5.55         2.36      0.18
##  White_alone 8.88         2.98      0.11
```

Independence

We would need to know more from the data providers to really assess this. We will assume it holds.

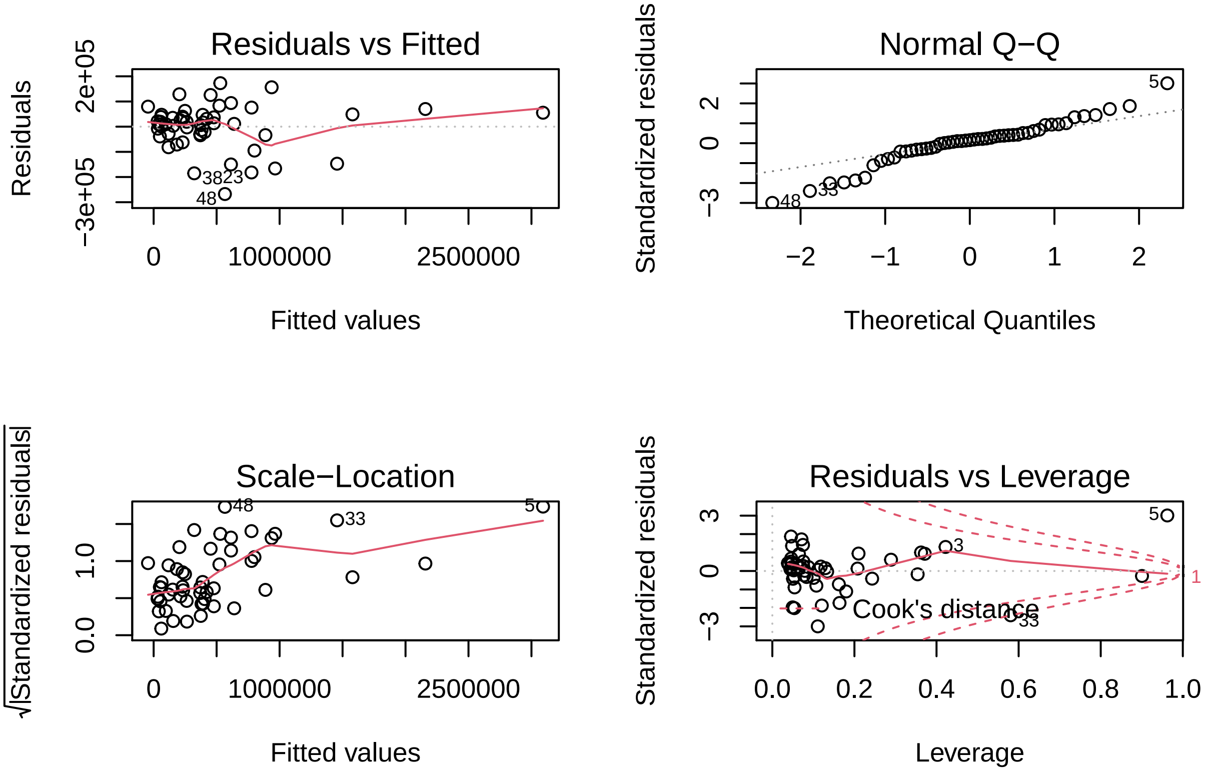
Linearity

The lack of fit F test works only with **simple linear regression** so we see the residual plots. As for the residuals versus fitted plot below, there may be no pattern indicating non-linearity in the data, but we attempt to remove some potential outliers.

```
par(mfrow = c(2,2))
# summary(reg)
plot(reg)
```

```
## Warning in sqrt(crit * p * (1 - hh)/hh): NaNs produced

## Warning in sqrt(crit * p * (1 - hh)/hh): NaNs produced
```



Remediation - Removing Influential Outliers

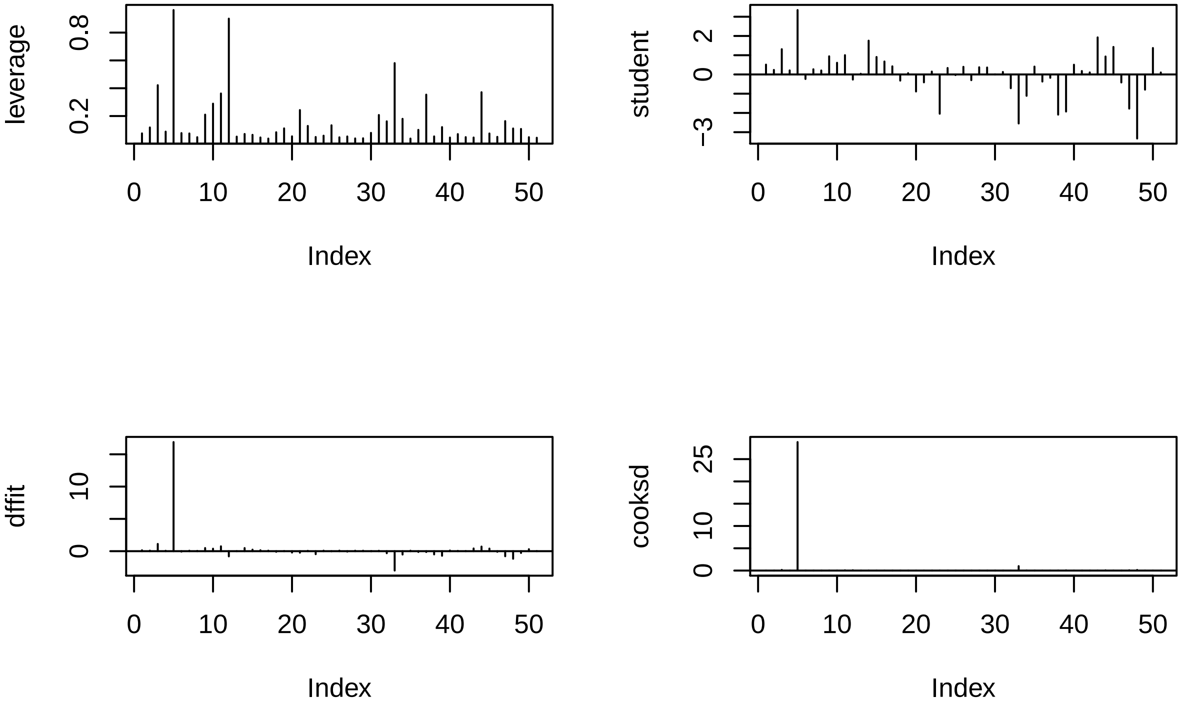
- Get other diagnostics measures

Take away the largest values of **cooks** and **leverage**. That is, observations 5 (California), is no longer present.

```
leverage <- hatvalues(reg)
student <- rstudent(reg)
dfs <- dffits(reg)
cooksd <- cooks.distance(reg)
data.frame(confirmed = covidForRegression2$confirmed, fitted = reg$fitted,
           residual = reg$residual, leverage, student, dffits = dfs, cooksd) -> diag

par(mfrow=c(2,2))
plot(leverage,type='h')
abline(h=0)
```

```
plot(student,type='h')
abline(h=0)
plot(dfs,type='h',ylab='dffit')
abline(h=0)
plot(cooksd,type='h')
abline(h=0)
```



```
diag %>%
  arrange(desc(leverage)) %>%
  head(3)
```

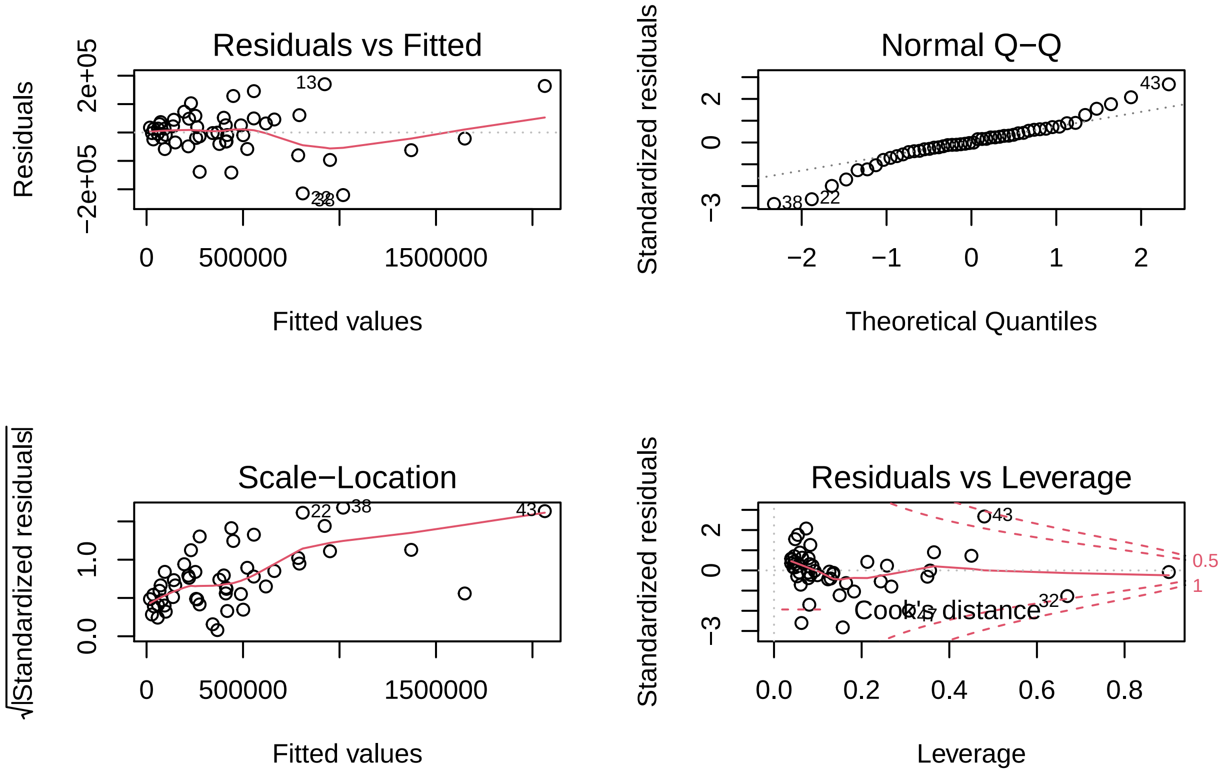
```
##    confirmed     fitted    residual  leverage    student     dffits      cooksd
## 5    3147207 3091857.30   55349.699 0.9622216  3.3473969 16.8936369 28.83180263
## 12     25658   33724.59   -8066.588 0.9005303 -0.2673471 -0.8044132  0.08267023
## 33   1309403 1456505.11 -147102.113 0.5805405 -2.5490024 -2.9987585  0.99665047
```

```
diag %>%
  arrange(desc(cooksd)) %>%
  head(3)
```

```
##    confirmed    fitted   residual  leverage   student    dffits    cooksd
## 5    3147207 3091857.3   55349.70 0.9622216  3.347397 16.893637 28.8318026
## 33   1309403 1456505.1 -147102.11 0.5805405 -2.549002 -2.998758  0.9966505
## 3     708041  614260.7   93780.28 0.4220373  1.314555  1.123322  0.1551054
```

- Look at the Residual vs Fitted plot; linearity has occurred.

```
reg1 <- lm(confirmed ~ ., data = covidForRegression2[-5,])
par(mfrow=c(2,2))
plot(reg1)
```



Homoscedasticity

With a small p-value, we have evidence that the variances are non-constant

```
# test homoscedasticity
ncvTest(reg1)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 18.38094, Df = 1, p = 1.8086e-05
```

Remediation - Square Root of Y

With a high p-value of 0.84229, there is no evidence of non-constant variance.

```
covidForRegression2 %>%
  mutate(confirmed = sqrt(confirmed)) -> covidForRegressionSQRT

reg2 <- lm(confirmed ~ ., data = covidForRegressionSQRT[-5,])
ncvTest(reg2)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 0.03958722, Df = 1, p = 0.84229
```

Normality

- The p-value of the Shapiro-Wilk Test 0.1822 is greater than α 0.05 so the data is follow a normal distribution.

```
# test normality
shapiro.test(rstudent(reg2))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  rstudent(reg2)
## W = 0.96747, p-value = 0.1822
```

Linear Model Selection

Our final mission is to select the **fewest** predictors and the determine by the **lowest** mean squared error in the linear model.

Adding all variables as full model first. We can see that only three variables in the linear model are significant at the beginning, which is `deaths`, `Black_or_African_American_alone`, and `White_alone`.

```
summary(reg2)
```

```
##
## Call:
## lm(formula = confirmed ~ ., data = covidForRegressionSQRT[-5,
##     ])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -222.18  -76.64   10.25   68.97  164.51
##
## Coefficients:
##                                             Estimate Std. Error t value
## (Intercept)                                3.624e+02  1.103e+02   3.285
## deaths                                     1.192e-02  5.486e-03   2.173
## American_Indian_and_Alaska_Native_alone    3.498e-04  2.184e-04   1.602
## Asian_alone                               -1.807e-04  1.219e-04  -1.483
## Black_or_African_American_alone            5.840e-05  2.727e-05   2.142
## Native_Hawaiian_and_Other_Pacific_Islander_alone  1.095e-05  7.942e-04   0.014
## White_alone                                4.288e-05  8.548e-06   5.017
## HouseholdIncome                           -1.027e-03  1.640e-03  -0.626
##                                           Pr(>|t|)
## (Intercept)                                0.00206 **
## deaths                                     0.03547 *
## American_Indian_and_Alaska_Native_alone    0.11674
## Asian_alone                                0.14562
## Black_or_African_American_alone            0.03807 *
## Native_Hawaiian_and_Other_Pacific_Islander_alone  0.98906
## White_alone                               1.01e-05 ***
## HouseholdIncome                            0.53469
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 98.64 on 42 degrees of freedom
## Multiple R-squared:  0.9041, Adjusted R-squared:  0.8882
## F-statistic: 56.58 on 7 and 42 DF,  p-value: < 2.2e-16
```

Exhaustive Search

Using algorithm to select the best model exhaustively. Also, to use all X-variables available, change the `nvmax option`. Because I am too lazy to count variables, I entered a much larger number, such as my favorite number `69` to do it.

```r
reg_fitExhaustive <- regsubsets(confirmed ~ ., data = covidForRegressionSQRT[-5,], nvmax = 69)
summary(reg_fitExhaustive)
```

```
## Subset selection object
## Call: regsubsets.formula(confirmed ~ ., data = covidForRegressionSQRT[-5,
##     ], nvmax = 69)
## 7 Variables  (and intercept)
##                                                   Forced in Forced out
## deaths                                                FALSE      FALSE
## American_Indian_and_Alaska_Native_alone               FALSE      FALSE
## Asian_alone                                           FALSE      FALSE
## Black_or_African_American_alone                       FALSE      FALSE
## Native_Hawaiian_and_Other_Pacific_Islander_alone      FALSE      FALSE
## White_alone                                           FALSE      FALSE
## HouseholdIncome                                       FALSE      FALSE
## 1 subsets of each size up to 7
## Selection Algorithm: exhaustive
##          deaths American_Indian_and_Alaska_Native_alone Asian_alone
## 1  ( 1 ) " "    " "                                     " "        
## 2  ( 1 ) " "    " "                                     " "        
## 3  ( 1 ) "*"    " "                                     "*"        
## 4  ( 1 ) "*"    " "                                     "*"        
## 5  ( 1 ) "*"    "*"                                     "*"        
## 6  ( 1 ) "*"    "*"                                     "*"        
## 7  ( 1 ) "*"    "*"                                     "*"        
##          Black_or_African_American_alone
## 1  ( 1 ) " "                            
## 2  ( 1 ) "*"                            
## 3  ( 1 ) " "                            
## 4  ( 1 ) "*"                            
## 5  ( 1 ) "*"                            
## 6  ( 1 ) "*"                            
## 7  ( 1 ) "*"                            
##          Native_Hawaiian_and_Other_Pacific_Islander_alone White_alone
## 1  ( 1 ) " "                                              "*"        
## 2  ( 1 ) " "                                              "*"        
## 3  ( 1 ) " "                                              "*"        
## 4  ( 1 ) " "                                              "*"        
## 5  ( 1 ) " "                                              "*"        
## 6  ( 1 ) " "                                              "*"        
## 7  ( 1 ) "*"                                              "*"        
##          HouseholdIncome
## 1  ( 1 ) " "            
## 2  ( 1 ) " "            
## 3  ( 1 ) " "            
## 4  ( 1 ) " "            
## 5  ( 1 ) " "            
## 6  ( 1 ) "*"            
## 7  ( 1 ) "*"
```

Select the variables and choose the optimal model

```
summary(reg_fitExhaustive)$adjr2
```

```
## [1] 0.8574573 0.8711181 0.8792470 0.8866123 0.8922247 0.8907509 0.8881502
```

```
summary(reg_fitExhaustive)$cp
```

```
## [1] 15.171783 10.157020  7.661584  5.618763  4.397164  6.000190  8.000000
```

```
summary(reg_fitExhaustive)$bic
```

```
## [1] -90.61260 -92.79050 -93.21127 -93.54488 -93.29474 -89.85308 -85.94128
```

```
which.max(summary(reg_fitExhaustive)$adjr2)
```

```
## [1] 5
```

```
which.min(summary(reg_fitExhaustive)$cp)
```

```
## [1] 5
```

```
which.min(summary(reg_fitExhaustive)$bic)
```

```
## [1] 4
```

Sequential Search

We can also choose the best model by means of a stepwise procedure, starting with one model and ending with another

Forward Method

Forward addition can be used to perform variable selection.

```
reg_fitForward <- regsubsets(confirmed ~ .,
                             data = covidForRegressionSQRT[-c(5),],
                             method = "forward")
summary(reg_fitForward)
```

```
## Subset selection object
## Call: regsubsets.formula(confirmed ~ ., data = covidForRegressionSQRT[-c(5),
##     ], method = "forward")
## 7 Variables  (and intercept)
##                                                  Forced in Forced out
## deaths                                               FALSE      FALSE
## American_Indian_and_Alaska_Native_alone              FALSE      FALSE
## Asian_alone                                          FALSE      FALSE
## Black_or_African_American_alone                      FALSE      FALSE
## Native_Hawaiian_and_Other_Pacific_Islander_alone     FALSE      FALSE
## White_alone                                          FALSE      FALSE
## HouseholdIncome                                      FALSE      FALSE
## 1 subsets of each size up to 7
## Selection Algorithm: forward
##          deaths American_Indian_and_Alaska_Native_alone Asian_alone
## 1  ( 1 ) " "    " "                                     " "
## 2  ( 1 ) " "    " "                                     " "
## 3  ( 1 ) " "    " "                                     " "
## 4  ( 1 ) "*"    " "                                     " "
## 5  ( 1 ) "*"    " "                                     "*"
## 6  ( 1 ) "*"    "*"                                     "*"
## 7  ( 1 ) "*"    "*"                                     "*"
##          Black_or_African_American_alone
## 1  ( 1 ) " "
## 2  ( 1 ) "*"
## 3  ( 1 ) "*"
## 4  ( 1 ) "*"
## 5  ( 1 ) "*"
## 6  ( 1 ) "*"
## 7  ( 1 ) "*"
##          Native_Hawaiian_and_Other_Pacific_Islander_alone White_alone
## 1  ( 1 ) " "                                              "*"
## 2  ( 1 ) " "                                              "*"
## 3  ( 1 ) " "                                              "*"
## 4  ( 1 ) " "                                              "*"
## 5  ( 1 ) " "                                              "*"
## 6  ( 1 ) " "                                              "*"
## 7  ( 1 ) "*"                                              "*"
##          HouseholdIncome
## 1  ( 1 ) " "
## 2  ( 1 ) " "
## 3  ( 1 ) "*"
## 4  ( 1 ) "*"
## 5  ( 1 ) "*"
## 6  ( 1 ) "*"
## 7  ( 1 ) "*"
```

Select the variables and choose the optimal model

```
summary(reg_fitForward)$adjr2
```

```
## [1] 0.8574573 0.8711181 0.8780890 0.8831499 0.8867090 0.8907509 0.8881502
```

```
summary(reg_fitForward)$cp
```

```
## [1] 15.171783 10.157020  8.137825  7.011779  6.566962  6.000190  8.000000
```

```
summary(reg_fitForward)$bic
```

```
## [1] -90.61260 -92.79050 -92.73407 -92.04092 -90.79917 -89.85308 -85.94128
```

```
which.max(summary(reg_fitForward)$adjr2)
```

```
## [1] 6
```

```
which.min(summary(reg_fitForward)$cp)
```

```
## [1] 6
```

```
which.min(summary(reg_fitForward)$bic)
```

```
## [1] 2
```

Backward Method

Backward elimination can be used to perform variable selection.

```
reg_fitBackward <- regsubsets(confirmed ~ .,
                              data = covidForRegressionSQRT[-c(5),], method = "backward")
summary(reg_fitBackward)
```

```
## Subset selection object
## Call: regsubsets.formula(confirmed ~ ., data = covidForRegressionSQRT[-c(5),
##     ], method = "backward")
## 7 Variables  (and intercept)
##                                                   Forced in Forced out
## deaths                                               FALSE      FALSE
## American_Indian_and_Alaska_Native_alone              FALSE      FALSE
## Asian_alone                                          FALSE      FALSE
## Black_or_African_American_alone                      FALSE      FALSE
## Native_Hawaiian_and_Other_Pacific_Islander_alone     FALSE      FALSE
## White_alone                                          FALSE      FALSE
## HouseholdIncome                                      FALSE      FALSE
## 1 subsets of each size up to 7
## Selection Algorithm: backward
##          deaths American_Indian_and_Alaska_Native_alone Asian_alone
## 1  ( 1 ) " "    " "                                     " "
## 2  ( 1 ) "*"    " "                                     " "
## 3  ( 1 ) "*"    " "                                     "*"
## 4  ( 1 ) "*"    " "                                     "*"
## 5  ( 1 ) "*"    "*"                                     "*"
## 6  ( 1 ) "*"    "*"                                     "*"
## 7  ( 1 ) "*"    "*"                                     "*"
##          Black_or_African_American_alone
## 1  ( 1 ) " "
## 2  ( 1 ) " "
## 3  ( 1 ) " "
## 4  ( 1 ) "*"
## 5  ( 1 ) "*"
## 6  ( 1 ) "*"
## 7  ( 1 ) "*"
##          Native_Hawaiian_and_Other_Pacific_Islander_alone White_alone
## 1  ( 1 ) " "                                              "*"
## 2  ( 1 ) " "                                              "*"
## 3  ( 1 ) " "                                              "*"
## 4  ( 1 ) " "                                              "*"
## 5  ( 1 ) " "                                              "*"
## 6  ( 1 ) " "                                              "*"
## 7  ( 1 ) "*"                                              "*"
##          HouseholdIncome
## 1  ( 1 ) " "
## 2  ( 1 ) " "
## 3  ( 1 ) " "
## 4  ( 1 ) " "
## 5  ( 1 ) " "
## 6  ( 1 ) "*"
## 7  ( 1 ) "*"
```

Select the variables and choose the optimal model

```r
summary(reg_fitBackward)$adjr2
```

```
## [1] 0.8574573 0.8651883 0.8792470 0.8866123 0.8922247 0.8907509 0.8881502
```

```r
summary(reg_fitBackward)$cp
```

```
## [1] 15.171783 12.648768  7.661584  5.618763  4.397164  6.000190  8.000000
```

```r
summary(reg_fitBackward)$bic
```

```
## [1] -90.61260 -90.54137 -93.21127 -93.54488 -93.29474 -89.85308 -85.94128
```

```r
which.max(summary(reg_fitBackward)$adjr2)
```

```
## [1] 5
```

```r
which.min(summary(reg_fitBackward)$cp)
```

```
## [1] 5
```

```r
which.min(summary(reg_fitBackward)$bic)
```

```
## [1] 4
```

Stepwise Method

We use algorithm to considers either adding or removing variables at each step to final the best model. Lower AIC (Akaike information criterion) values indicate a better-fit model.

```r
null = lm(confirmed ~ 1, data = covidForRegressionSQRT[-c(5),])
full = lm(confirmed ~ ., data = covidForRegressionSQRT[-c(5),])
step(null, scope = list(lower = null, upper = full), direction = "both")
```

```
## Start:  AIC=569.67
## confirmed ~ 1
## 
##                                                      Df Sum of Sq     RSS    AIC
## + White_alone                                         1   3667200  595170 473.23
## + deaths                                              1   3265900  996470 499.00
## + Black_or_African_American_alone                     1   2944490 1317880 512.98
## + Asian_alone                                         1   2115961 2146409 537.36
## + American_Indian_and_Alaska_Native_alone             1    265309 3997061 568.45
## <none>                                                            4262370 569.67
## + HouseholdIncome                                     1    121852 4140518 570.22
## + Native_Hawaiian_and_Other_Pacific_Islander_alone    1     19189 4243181 571.44
## 
## Step:  AIC=473.23
## confirmed ~ White_alone
## 
##                                                      Df Sum of Sq     RSS    AIC
## + Black_or_African_American_alone                     1     68250  526920 469.14
## + HouseholdIncome                                     1     46407  548764 471.17
## + deaths                                              1     44007  551164 471.39
## + Native_Hawaiian_and_Other_Pacific_Islander_alone    1     35741  559429 472.13
## <none>                                                             595170 473.23
## + American_Indian_and_Alaska_Native_alone             1     12741  582429 474.15
## + Asian_alone                                         1      6779  588392 474.66
## - White_alone                                         1   3667200 4262370 569.67
## 
## Step:  AIC=469.14
## confirmed ~ White_alone + Black_or_African_American_alone
```

```
##
##                                                    Df Sum of Sq     RSS    AIC
## + HouseholdIncome                                   1     39105  487816 467.28
## + American_Indian_and_Alaska_Native_alone           1     28034  498887 468.41
## + Native_Hawaiian_and_Other_Pacific_Islander_alone  1     26719  500201 468.54
## <none>                                                           526920 469.14
## + deaths                                            1     17914  509006 469.41
## + Asian_alone                                       1     16544  510377 469.54
## - Black_or_African_American_alone                   1     68250  595170 473.23
## - White_alone                                       1    790959 1317880 512.98
##
## Step:  AIC=467.28
## confirmed ~ White_alone + Black_or_African_American_alone + HouseholdIncome
##
##                                                    Df Sum of Sq     RSS    AIC
## + deaths                                            1     30415  457401 466.06
## <none>                                                           487816 467.28
## + American_Indian_and_Alaska_Native_alone           1     17844  469972 467.42
## + Native_Hawaiian_and_Other_Pacific_Islander_alone  1     13595  474221 467.87
## - HouseholdIncome                                   1     39105  526920 469.14
## + Asian_alone                                       1       482  487334 469.23
## - Black_or_African_American_alone                   1     60948  548764 471.17
## - White_alone                                       1    796518 1284334 513.69
##
## Step:  AIC=466.06
## confirmed ~ White_alone + Black_or_African_American_alone + HouseholdIncome +
##     deaths
##
##                                                    Df Sum of Sq    RSS    AIC
## + Asian_alone                                       1     23787 433614 465.39
## <none>                                                          457401 466.06
## + American_Indian_and_Alaska_Native_alone           1     17851 439550 466.07
## + Native_Hawaiian_and_Other_Pacific_Islander_alone  1      8137 449263 467.17
## - deaths                                            1     30415 487816 467.28
## - Black_or_African_American_alone                   1     30531 487932 467.30
## - HouseholdIncome                                   1     51606 509006 469.41
## - White_alone                                       1    274482 731883 487.57
##
## Step:  AIC=465.39
## confirmed ~ White_alone + Black_or_African_American_alone + HouseholdIncome +
##     deaths + Asian_alone
##
##                                                    Df Sum of Sq    RSS    AIC
## + American_Indian_and_Alaska_Native_alone           1     24973 408640 464.43
## - HouseholdIncome                                   1     10233 443847 464.56
## <none>                                                          433614 465.39
## - Asian_alone                                       1     23787 457401 466.06
## - Black_or_African_American_alone                   1     33415 467029 467.11
## + Native_Hawaiian_and_Other_Pacific_Islander_alone  1        17 433596 467.39
## - deaths                                            1     53720 487334 469.23
## - White_alone                                       1    291418 725032 489.10
##
## Step:  AIC=464.43
## confirmed ~ White_alone + Black_or_African_American_alone + HouseholdIncome +
##     deaths + Asian_alone + American_Indian_and_Alaska_Native_alone
##
##                                                     Df Sum of Sq    RSS    AIC
## - HouseholdIncome                                    1      3862 412503 462.90
## <none>                                                           408640 464.43
## - American_Indian_and_Alaska_Native_alone            1     24973 433614 465.39
## - Asian_alone                                        1     30909 439550 466.07
## + Native_Hawaiian_and_Other_Pacific_Islander_alone  1         2 408639 466.43
## - Black_or_African_American_alone                    1     44852 453493 467.64
## - deaths                                             1     59879 468520 469.27
## - White_alone                                        1    249276 657917 486.24
##
## Step:  AIC=462.9
## confirmed ~ White_alone + Black_or_African_American_alone + deaths +
##     Asian_alone + American_Indian_and_Alaska_Native_alone
##
##                                                     Df Sum of Sq    RSS    AIC
## <none>                                                           412503 462.90
## + HouseholdIncome                                    1      3862 408640 464.43
## - American_Indian_and_Alaska_Native_alone            1     31345 443847 464.56
## + Native_Hawaiian_and_Other_Pacific_Islander_alone  1        51 412452 464.89
## - Black_or_African_American_alone                    1     51235 463738 466.75
## - Asian_alone                                        1     66859 479361 468.41
## - deaths                                             1     70052 482555 468.74
## - White_alone                                        1    260642 673144 485.38
```

```
##
## Call:
## lm(formula = confirmed ~ White_alone + Black_or_African_American_alone +
##     deaths + Asian_alone + American_Indian_and_Alaska_Native_alone,
##     data = covidForRegressionSQRT[-c(5), ])
##
## Coefficients:
##                             (Intercept)
##                               2.954e+02
##                             White_alone
##                               4.354e-05
##         Black_or_African_American_alone
##                               6.140e-05
##                                  deaths
##                               1.254e-02
##                             Asian_alone
##                              -2.163e-04
## American_Indian_and_Alaska_Native_alone
##                               3.813e-04
```

Comparison

We will select the fewest variable for each set, compare their MSE, and finally select the one with the local minimum MSE.

1. From exhaustive search, the fewest predictors is 4 in smallest BIC (Bayesian Information Criterion).

```
regExhaustive <- lm(confirmed ~ deaths + Asian_alone + Black_or_African_American_alone + White_alone
                    , data = covidForRegressionSQRT[-c(5),])
```

```r
summary(regExhaustive)
```

```
##
## Call:
## lm(formula = confirmed ~ deaths + Asian_alone + Black_or_African_American_alone +
##     White_alone, data = covidForRegressionSQRT[-c(5), ])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -232.97  -70.40   13.42   69.35  159.52
##
## Coefficients:
##                                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      3.085e+02  2.043e+01  15.100  < 2e-16 ***
## deaths                           1.221e-02  4.703e-03   2.597   0.0127 *
## Asian_alone                     -2.135e-04  8.305e-05  -2.570   0.0135 *
## Black_or_African_American_alone  5.296e-05  2.652e-05   1.997   0.0519 .
## White_alone                      4.699e-05  8.247e-06   5.698 8.76e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 99.31 on 45 degrees of freedom
## Multiple R-squared:  0.8959, Adjusted R-squared:  0.8866
## F-statistic: 96.79 on 4 and 45 DF,  p-value: < 2.2e-16
```

2. For forward method in sequential search, the fewest predictors is 2 in smallest BIC (Bayesian Information Criterion).

```r
regForward <- lm(confirmed ~ Black_or_African_American_alone + White_alone,
                 data = covidForRegressionSQRT[-c(5),])
summary(regForward)
```

```
##
## Call:
## lm(formula = confirmed ~ Black_or_African_American_alone + White_alone,
##     data = covidForRegressionSQRT[-c(5), ])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -237.00  -63.71   10.56   80.07  204.06
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     3.087e+02  2.170e+01  14.228  < 2e-16 ***
## Black_or_African_American_alone 6.614e-05  2.681e-05   2.467   0.0173 *
## White_alone                     5.366e-05  6.388e-06   8.400   6.5e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 105.9 on 47 degrees of freedom
## Multiple R-squared:  0.8764, Adjusted R-squared:  0.8711
## F-statistic: 166.6 on 2 and 47 DF,  p-value: < 2.2e-16
```

3. For backward method in sequential search, the fewest predictors is 4 in smallest BIC (Bayesian Information Criterion).

```
regBackward <-  lm(confirmed ~ deaths + Asian_alone + Black_or_African_American_alone + White_alone
                    , data = covidForRegressionSQRT[-c(5),])
summary(regBackward)
```

```
##
## Call:
## lm(formula = confirmed ~ deaths + Asian_alone + Black_or_African_American_alone +
##     White_alone, data = covidForRegressionSQRT[-c(5), ])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -232.97  -70.40   13.42   69.35  159.52
##
## Coefficients:
##                                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      3.085e+02  2.043e+01  15.100  < 2e-16 ***
## deaths                           1.221e-02  4.703e-03   2.597   0.0127 *
## Asian_alone                     -2.135e-04  8.305e-05  -2.570   0.0135 *
## Black_or_African_American_alone  5.296e-05  2.652e-05   1.997   0.0519 .
## White_alone                      4.699e-05  8.247e-06   5.698 8.76e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 99.31 on 45 degrees of freedom
## Multiple R-squared:  0.8959, Adjusted R-squared:  0.8866
## F-statistic: 96.79 on 4 and 45 DF,  p-value: < 2.2e-16
```

4. For stepwise method, the lowest AIC value is **462.9**, reporting the best model with 5 variables is: `lm(formula = confirmed ~ White_alone + Black_or_African_American_alone + deaths + Asian_alone + American_Indian_and_Alaska_Native_alone, data = covidForRegressionSQRT[-c(5), ])`.

```
regStepwise <- lm(formula = confirmed ~ White_alone + Black_or_African_American_alone +
                    deaths + Asian_alone + American_Indian_and_Alaska_Native_alone,
                  data = covidForRegressionSQRT[-c(5), ])
summary(regStepwise)
```

```
##
## Call:
## lm(formula = confirmed ~ White_alone + Black_or_African_American_alone +
##     deaths + Asian_alone + American_Indian_and_Alaska_Native_alone,
##     data = covidForRegressionSQRT[-c(5), ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -219.089  -71.294    8.229   68.185  164.196
##
## Coefficients:
##                                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)                              2.954e+02  2.116e+01  13.956  < 2e-16
## White_alone                              4.354e-05  8.258e-06   5.273  3.9e-06
## Black_or_African_American_alone          6.140e-05  2.627e-05   2.338  0.02401
## deaths                                   1.254e-02  4.588e-03   2.734  0.00899
## Asian_alone                             -2.163e-04  8.098e-05  -2.670  0.01057
## American_Indian_and_Alaska_Native_alone  3.813e-04  2.085e-04   1.828  0.07426
```

```
##
## (Intercept)                             ***
## White_alone                             ***
## Black_or_African_American_alone         *
## deaths                                  **
## Asian_alone                             *
## American_Indian_and_Alaska_Native_alone .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 96.82 on 44 degrees of freedom
## Multiple R-squared:  0.9032, Adjusted R-squared:  0.8922
## F-statistic: 82.13 on 5 and 44 DF,  p-value: < 2.2e-16
```

5. Let us start to find the minimum MSE

```
#calculate MSE
anova(regExhaustive) %>% tidy() # 9863.275
```

```
## # A tibble: 5 x 6
##   term                               df    sumsq   meansq statistic   p.value
##   <chr>                           <int>    <dbl>    <dbl>     <dbl>     <dbl>
## 1 deaths                              1 3265900. 3265900.    331.    2.25e-22
## 2 Asian_alone                         1   55370.   55370.      5.61  2.22e- 2
## 3 Black_or_African_American_alone     1  177072.  177072.     18.0   1.11e- 4
## 4 White_alone                         1  320182.  320182.     32.5   8.76e- 7
## 5 Residuals                          45  443847.    9863.     NA    NA
```

```
anova(regForward) %>% tidy() # 11211.07
```

```
## # A tibble: 3 x 6
##   term                               df    sumsq   meansq statistic   p.value
##   <chr>                           <int>    <dbl>    <dbl>     <dbl>     <dbl>
## 1 Black_or_African_American_alone     1 2944490. 2944490.     263.   7.19e-21
## 2 White_alone                         1  790959.  790959.      70.6  6.50e-11
## 3 Residuals                          47  526920.   11211.      NA   NA
```

```
anova(regBackward) %>% tidy() # 9863.275
```

```
## # A tibble: 5 x 6
##   term                               df    sumsq   meansq statistic   p.value
##   <chr>                           <int>    <dbl>    <dbl>     <dbl>     <dbl>
## 1 deaths                              1 3265900. 3265900.    331.    2.25e-22
## 2 Asian_alone                         1   55370.   55370.      5.61  2.22e- 2
## 3 Black_or_African_American_alone     1  177072.  177072.     18.0   1.11e- 4
## 4 White_alone                         1  320182.  320182.     32.5   8.76e- 7
## 5 Residuals                          45  443847.    9863.     NA    NA
```

```
anova(regStepwise) %>% tidy() # 9375.065
```

```
## # A tibble: 6 x 6
##   term                                   df    sumsq  meansq statistic   p.value
##   <chr>                               <int>    <dbl>   <dbl>     <dbl>     <dbl>
## 1 White_alone                             1 3667200.  3.67e6    391.    1.60e-23
## 2 Black_or_African_American_alone         1   68250.  6.83e4      7.28  9.85e- 3
## 3 deaths                                  1   17914.  1.79e4      1.91  1.74e- 1
## 4 Asian_alone                             1   65159.  6.52e4      6.95  1.15e- 2
## 5 American_Indian_and_Alaska_Native_~     1   31345.  3.13e4      3.34  7.43e- 2
## 6 Residuals                              44  412503.  9.38e3     NA    NA
```

Final Selection

`regStepwise` has the smallest MSE value (9375.065) so we select it as the best model. Therefore, our final predictors of square root of COVID confirmed cases by US States is `deaths`, `White_alone`,`Black_or_African_American_alone`, `Asian_alone` and `American_Indian_and_Alaska_Native_alone`.

```
finalReg <- lm(formula = confirmed ~ White_alone + Black_or_African_American_alone +
    deaths + Asian_alone + American_Indian_and_Alaska_Native_alone,
    data = covidForRegressionSQRT[-c(5), ])
summary(finalReg)
```

```
##
## Call:
## lm(formula = confirmed ~ White_alone + Black_or_African_American_alone +
##     deaths + Asian_alone + American_Indian_and_Alaska_Native_alone,
##     data = covidForRegressionSQRT[-c(5), ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -219.089  -71.294    8.229   68.185  164.196
##
## Coefficients:
##                                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)                             2.954e+02  2.116e+01  13.956  < 2e-16
## White_alone                             4.354e-05  8.258e-06   5.273  3.9e-06
## Black_or_African_American_alone         6.140e-05  2.627e-05   2.338  0.02401
## deaths                                  1.254e-02  4.588e-03   2.734  0.00899
## Asian_alone                            -2.163e-04  8.098e-05  -2.670  0.01057
## American_Indian_and_Alaska_Native_alone  3.813e-04  2.085e-04   1.828  0.07426
##
## (Intercept)                             ***
## White_alone                             ***
## Black_or_African_American_alone         *
## deaths                                  **
## Asian_alone                             *
## American_Indian_and_Alaska_Native_alone .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 96.82 on 44 degrees of freedom
## Multiple R-squared:  0.9032, Adjusted R-squared:  0.8922
## F-statistic: 82.13 on 5 and 44 DF,  p-value: < 2.2e-16
```

Model Equation

Interpretation

With the large p-value 0.07426 of b_5, the prodictor `American_Indian_and_Alaska_Native_alone` is not in the significant level. In other words, we fail to reject the null (H0: $\beta_5 = 0$ cannot be rejected) so we can conclude b_5 is 0. Therefore, b_5 can be dropped from the model.

Linear Equation

Thus, the expected value of square root of confirmed cases is:

$$\sqrt{Con\hat{firmed}} = 295.4 + 0.00004354White + 0.0000614Black + 0.01254Deaths - 0.0002163Asian$$

References

(1) Sehra, S., Fundin, S., Lavery, C., & Baker, J. (2020). Differences in race and other state-level characteristics and associations with mortality from COVID-19 infection. *Journal of Medical Virology, 92*(11), 2406–2408. https://doi.org/10.1002/jmv.26095

(2) Sa, Filipa G., Socioeconomic Determinants of Covid-19 Infections and Mortality: Evidence from England and Wales (May 2020). CEPR Discussion Paper No. DP14781, Available at SSRN: https://ssrn.com/abstract=3612850