STAT 512 Group Project: Covid Death Rate

Sahana Rayan, Hyeong Kyun Park, Gage Miller, Hannah Crouch

November 26, 2020

1. Introduction

1.1 Background

During the COVID-19 pandemic, there have been various areas throughout the United States that have been severely affected for various reasons. During this time, many people have attempted to predict which factors will have the biggest effect on how a population will react to the spread. Our goal is to analyze several factors from the nation's largest cities to see what factors truly impact the death rate of COVID-19.

We were able to find a published article with the same research question. From the published article, it was shown that population density, testing rate, airport traffic, and high age groups emerge as the most significant variables, while healthcare index, homelessness, and GDP have small impacts. Our goal is to see if there are similarities between the datasets.

1.2 Research Question

What is your best linear regression model to predict the mean death rate? Justify your model. Do a cross validation on your model. What factors contribute to the COVID death rate? Discuss the actual impact and define multiple Ho/Ha according to your hypothesis. Then verify your hypotheses with the data. $H_o : \beta_1 \neq \beta_2 \neq \cdots \neq \beta_7 \neq 0$ (Every variable contributes to the death rate) H_a : at least one $\beta_i = 0$ (There is at least one variable that does not contribute to the death rate)

1.3 Source of the data

This data set came from the census.

2. Data set characteristic

2.1 Variables

```
library(readxl)
library(tinytex)
```

```
## Warning: package 'tinytex' was built under R version 4.0.3
```

```r
covid <- read_excel("C:\\Users\\sahan\\Downloads\\CITY COVID.xlsx", sheet = "Sheet2" )
summary(covid)
```

```
##      City           City Area (mi^2) death rate in city   Median age
##  Length:46          Min.   : 26.0    Min.   :0.00935    Min.   :30.00
##  Class :character   1st Qu.:142.6    1st Qu.:0.01409    1st Qu.:33.02
##  Mode  :character   Median :225.9    Median :0.02032    Median :34.35
##                     Mean   :273.7    Mean   :0.02557    Mean   :34.33
##                     3rd Qu.:371.3    3rd Qu.:0.03192    3rd Qu.:35.75
##                     Max.   :875.0    Max.   :0.10490    Max.   :39.00
##  avg city household income # of hospitals  City population
##  Min.   : 29008            Min.   : 2.00   Min.   : 300576
##  1st Qu.: 47042            1st Qu.: 8.00   1st Qu.: 498198
##  Median : 54643            Median :13.00   Median : 681309
##  Mean   : 56568            Mean   :13.98   Mean   :1054515
##  3rd Qu.: 61215            3rd Qu.:17.75   3rd Qu.: 911026
##  Max.   :104552            Max.   :30.00   Max.   :8399000
##  % of people in poverty % without health insurance
##  Min.   : 7.50          Min.   : 3.800
##  1st Qu.:14.53          1st Qu.: 8.725
##  Median :18.75          Median :10.800
##  Mean   :18.42          Mean   :11.465
##  3rd Qu.:20.48          3rd Qu.:13.025
##  Max.   :36.40          Max.   :23.800
```

```r
library(knitr)
```

```
## Warning: package 'knitr' was built under R version 4.0.3
```

```r
tab <- read.csv("C:\\Users\\sahan\\Downloads\\tableforvars.csv")
kable(tab, caption = "sahana")
```

Table 1: sahana

ï..Variable.Name	Unit	Type	Range
City	None	Discrete	46 cities
City Area	mi2	Continuous	26 - 875
Death rate (Response Variable)	Number of deaths per cases	Continuous	0.00935 -0.10490
Median Age	years	Continuous	30 - 39
Average Median household income	Dollars	Continuous	29005 - 104552
Number of hospitals	None	Discrete/Continuous	Feb-30
City population	None	Continuous	300576 - 8399000
% in poverty	None	Continuous	7.5 - 36.4
% without health insurance	None	Continuous	3.8 - 23.8

Sample Size: There are a total of 46 rows or cities in the data set.

2.2 Data cleaning

```
covid$City <- NULL
colnames(covid) <- c('city_area', 'death_rate', 'med_age', 'avg_household', 'num_hospitals',
                          'pop', 'poverty', 'no_health_insurance')
apply(covid, 2, function(x) any(is.na(x)))
```

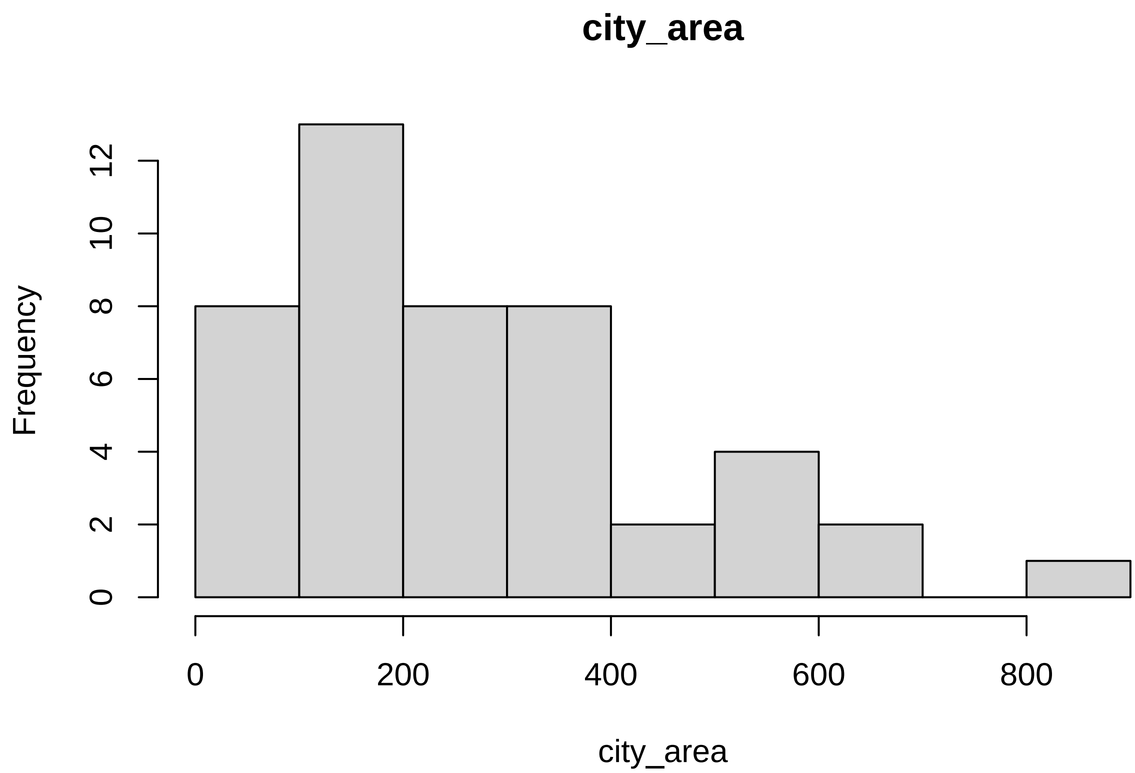
```
##           city_area          death_rate             med_age       avg_household
##               FALSE               FALSE               FALSE               FALSE
##       num_hospitals                 pop             poverty no_health_insurance
##               FALSE               FALSE               FALSE               FALSE
```

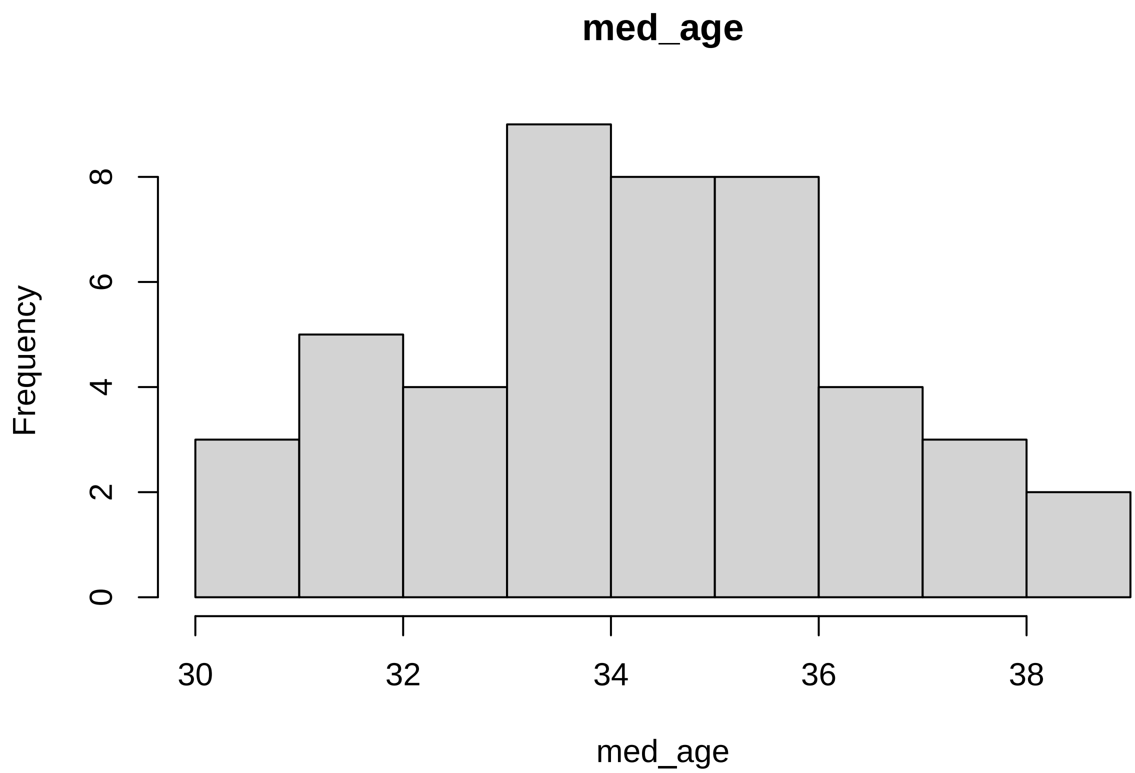
The column with city names was excluded since the values are essentially used as unique index labels and do not offer much to the prediction. No NA values were present, hence there was no need for handling NA.

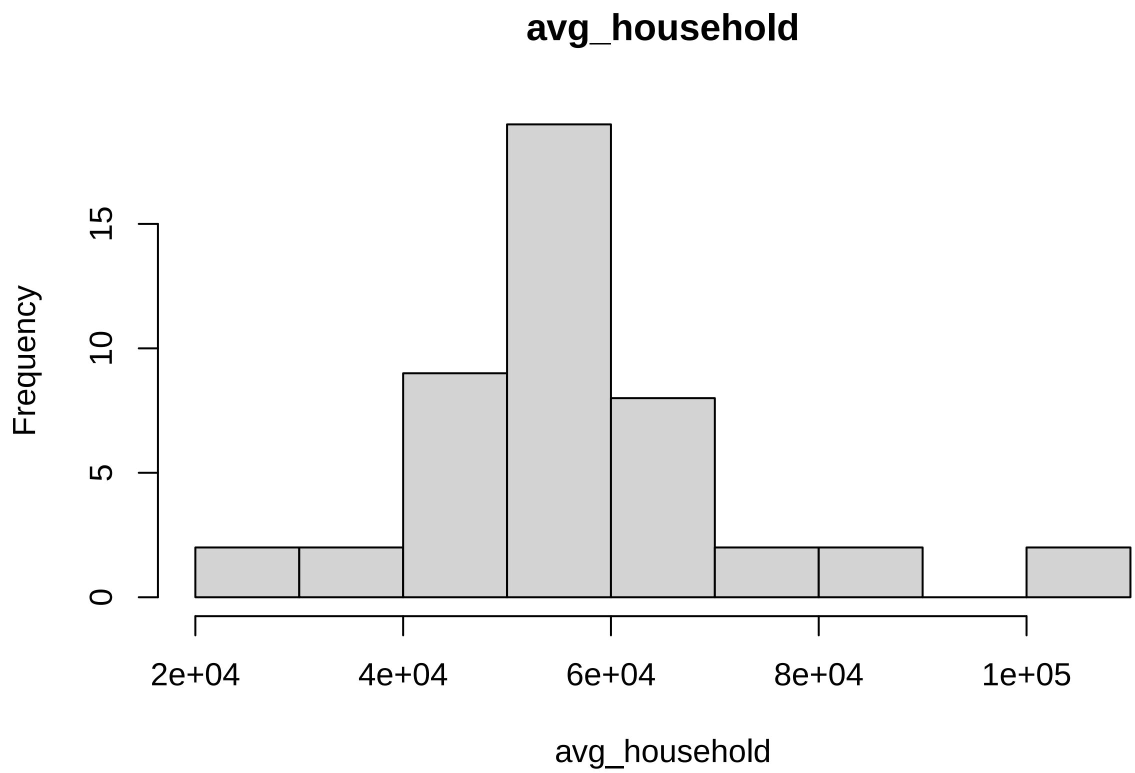
3. Preliminary Analysis

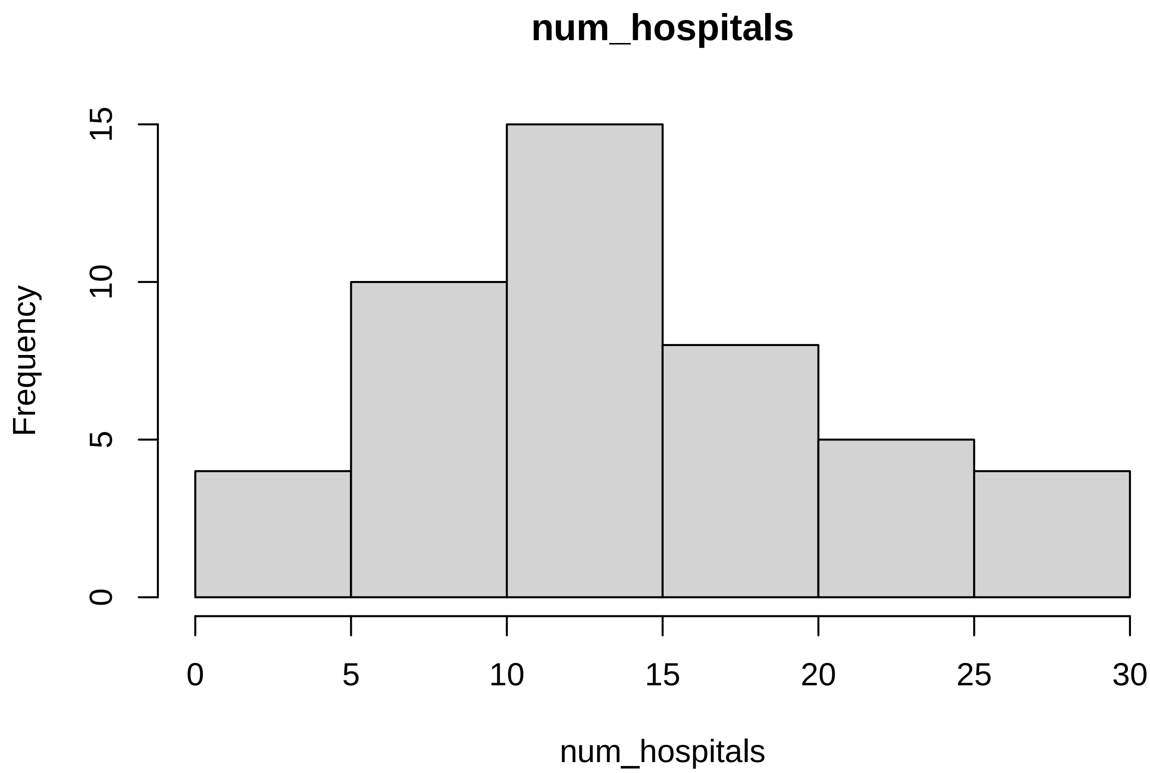
3.1 Histograms

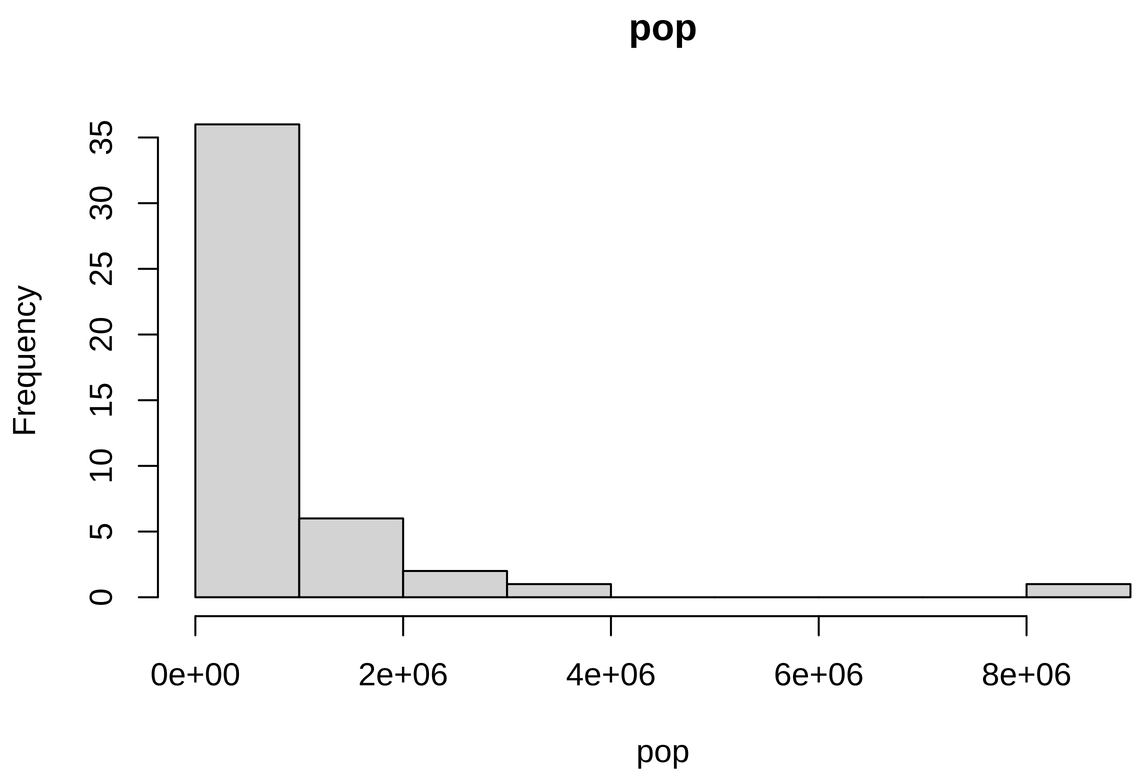
```
for (colname in colnames(covid)) {
  if (colname != 'death_rate') {
    hist(as.numeric(unlist(covid[,colname])), main=colname, xlab=colname)
  }
}
```

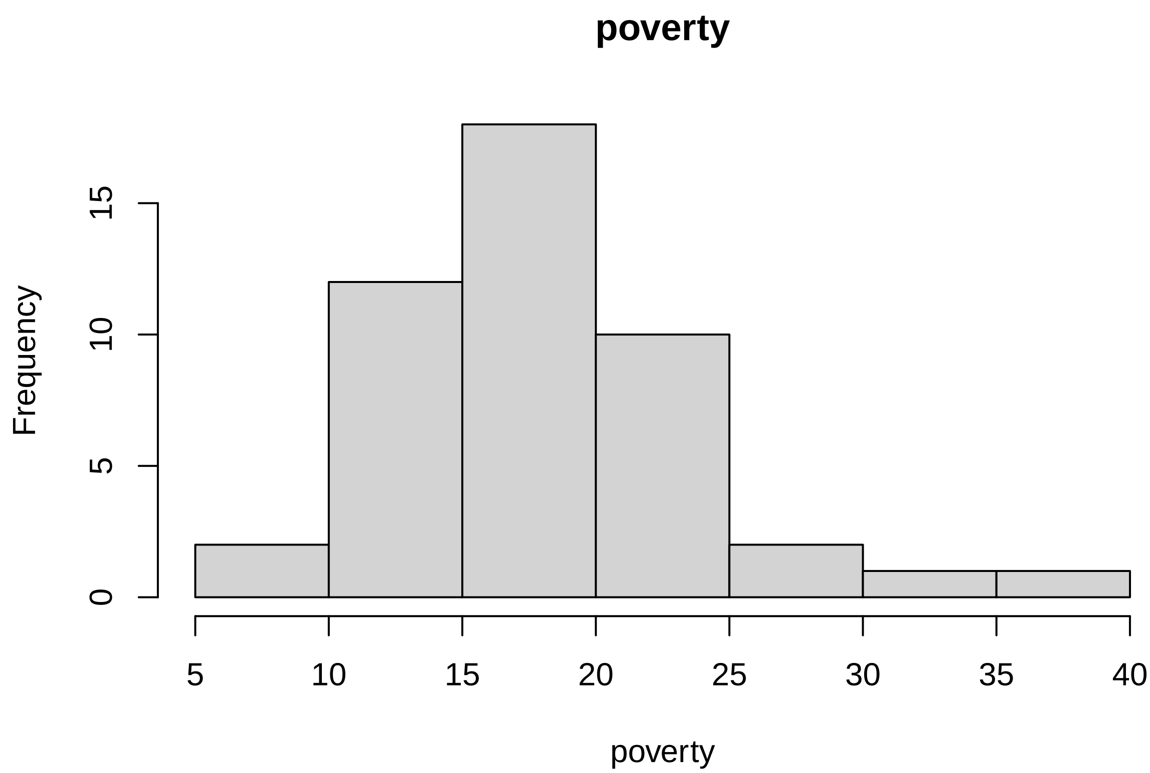

city_area
Frequency
0
2
4
6
8
10
12
0
200
400
600
800
city_area

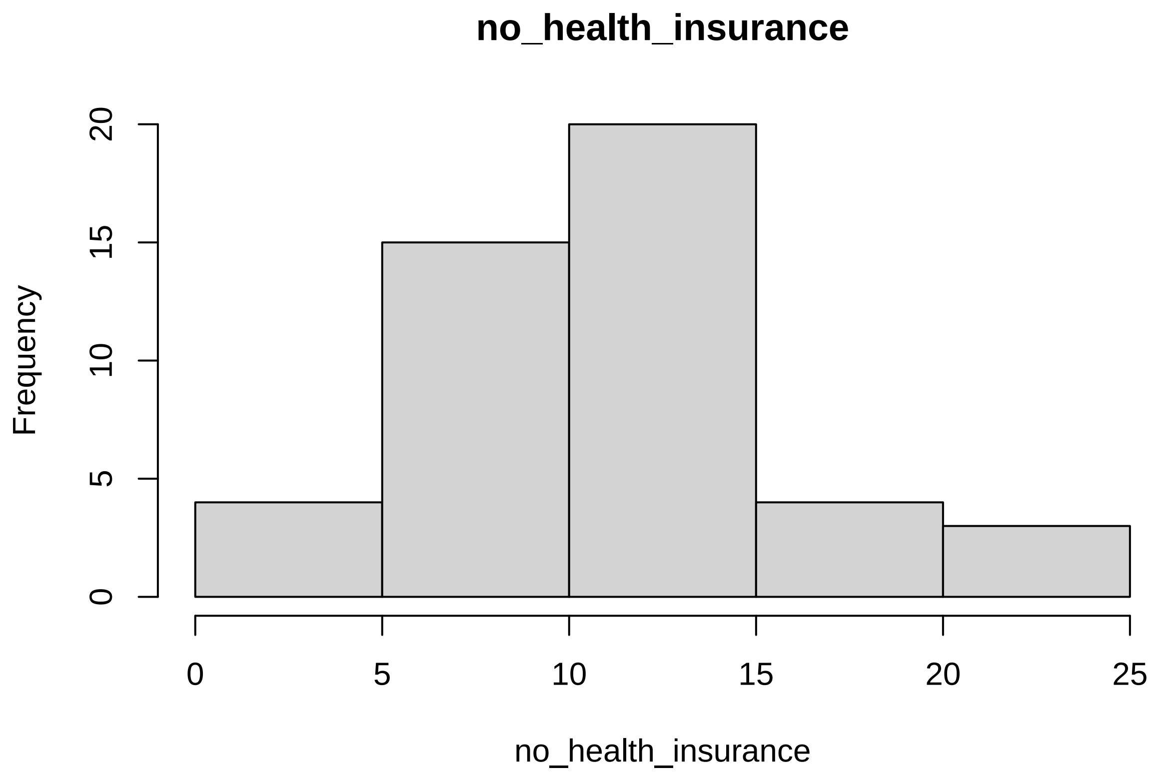

med_age
Frequency
0
2
4
6
8
30
32
34
36
38
med_age


avg_household
Frequency
0
5
10
15
2e+04
4e+04
6e+04
8e+04
1e+05
avg_household


num_hospitals
Frequency
0
5
10
15
0
5
10
15
20
25
30
num_hospitals

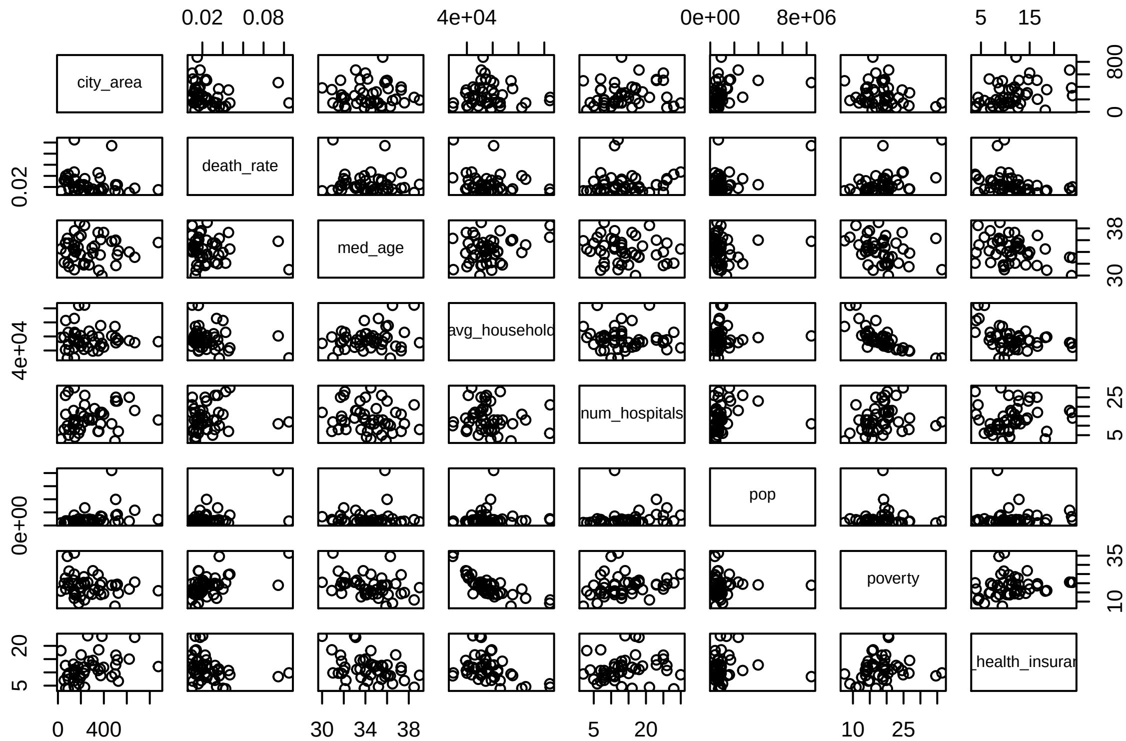

pop
Frequency
0
5
10
15
20
25
30
35
0e+00
2e+06
4e+06
6e+06
8e+06
pop


poverty
Frequency
0
5
10
15
5
10
15
20
25
30
35
40
poverty



3.2 Scatter Plots & Pairwise Correlations

```
plot(covid)
```




```
cor(covid)
```

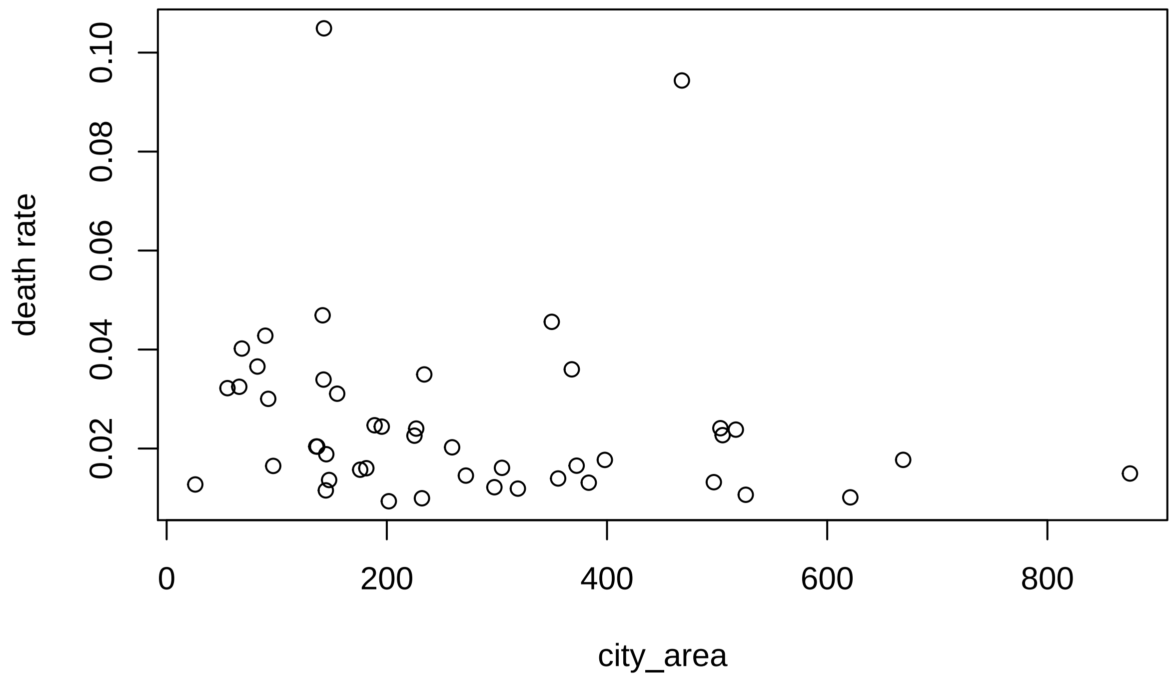
```
##                        city_area  death_rate     med_age avg_household
## city_area             1.00000000 -0.15903992  0.01572523   -0.04368357
## death_rate           -0.15903992  1.00000000 -0.11383544   -0.23374168
## med_age               0.01572523 -0.11383544  1.00000000    0.26663702
## avg_household        -0.04368357 -0.23374168  0.26663702    1.00000000
## num_hospitals         0.17180062  0.09788775 -0.16522422   -0.09733396
## pop                   0.34759003  0.47656391  0.01918488    0.07600774
## poverty              -0.18752896  0.51858615 -0.31143923   -0.77132988
## no_health_insurance   0.36832757 -0.29099952 -0.36378621   -0.42070780
##                      num_hospitals        pop     poverty no_health_insurance
## city_area               0.17180062 0.34759003 -0.18752896          0.36832757
## death_rate              0.09788775 0.47656391  0.51858615         -0.29099952
## med_age                -0.16522422 0.01918488 -0.31143923         -0.36378621
## avg_household          -0.09733396 0.07600774 -0.77132988         -0.42070780
## num_hospitals           1.00000000 0.19246652  0.21447584          0.11838818
## pop                     0.19246652 1.00000000  0.01299165          0.03683754
## poverty                 0.21447584 0.01299165  1.00000000          0.15100138
## no_health_insurance     0.11838818 0.03683754  0.15100138          1.00000000
```

- Poverty and avg_household might potentially have a multicollinearity issue, and we will find out later in the report by analyzing the VIF value between avg_household and poverty.

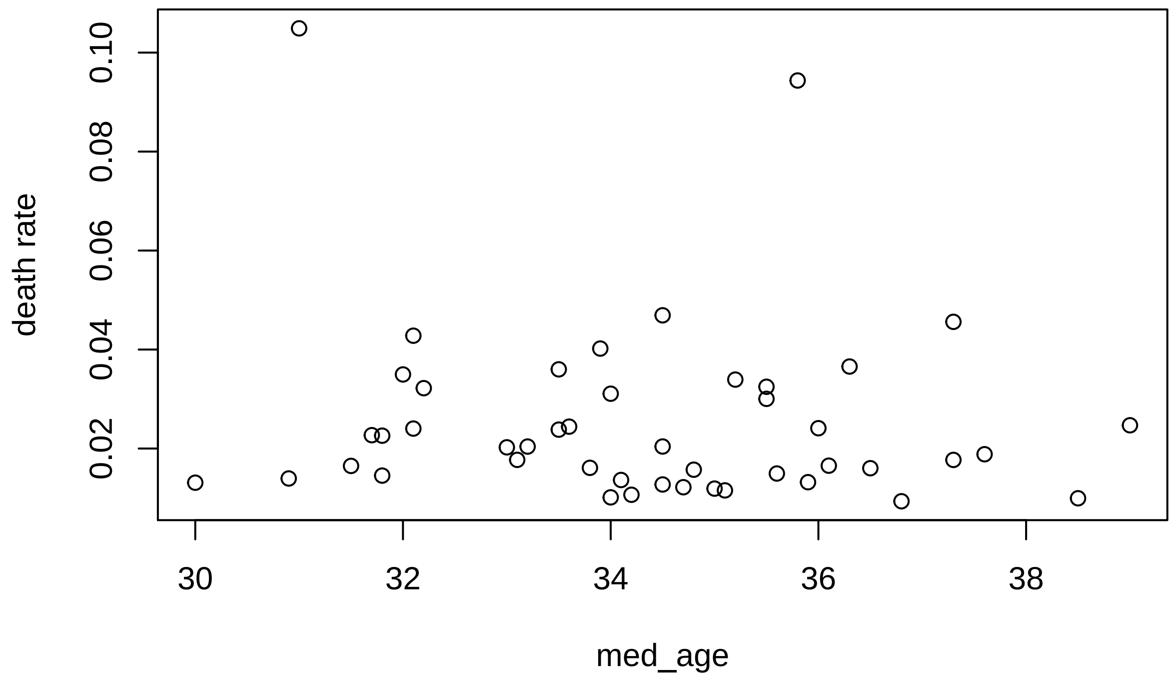
3.3 SLR Output

```
for (colname in colnames(covid)) {
  if (colname != 'death_rate') {
    citycovid.lm <- lm(as.numeric(unlist(covid[,"death_rate"]))~as.numeric(unlist(covid[,colname])))
    print(summary(citycovid.lm)) # SLR result
    plot(as.numeric(unlist(covid[,"death_rate"]))~as.numeric(unlist(covid[,colname])), xlab=colname, yla
    }
}
```

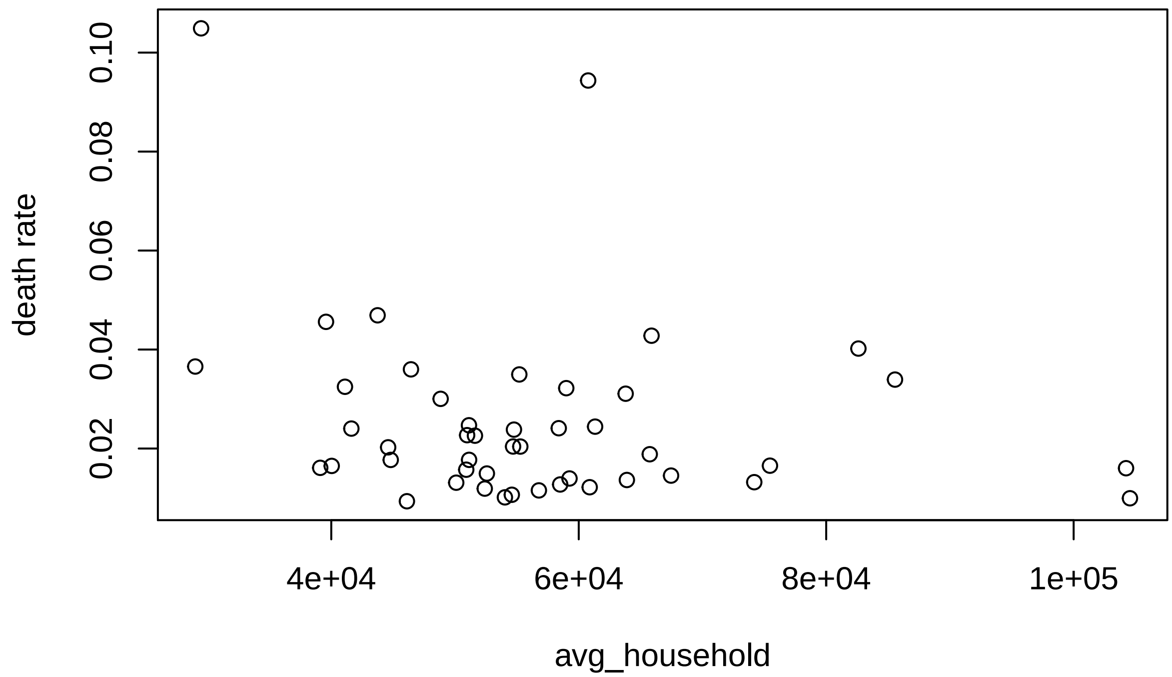
```
##
## Call:
## lm(formula = as.numeric(unlist(covid[, "death_rate"])) ~ as.numeric(unlist(covid[,
##     colname])))
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.017395 -0.010780 -0.004665  0.003411  0.077197
##
## Coefficients:
##                                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)                           3.003e-02  5.015e-03   5.988 3.52e-07 ***
## as.numeric(unlist(covid[, colname])) -1.627e-05  1.523e-05  -1.069    0.291
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01892 on 44 degrees of freedom
## Multiple R-squared:  0.02529,    Adjusted R-squared:  0.003141
## F-statistic: 1.142 on 1 and 44 DF,  p-value: 0.2911
```



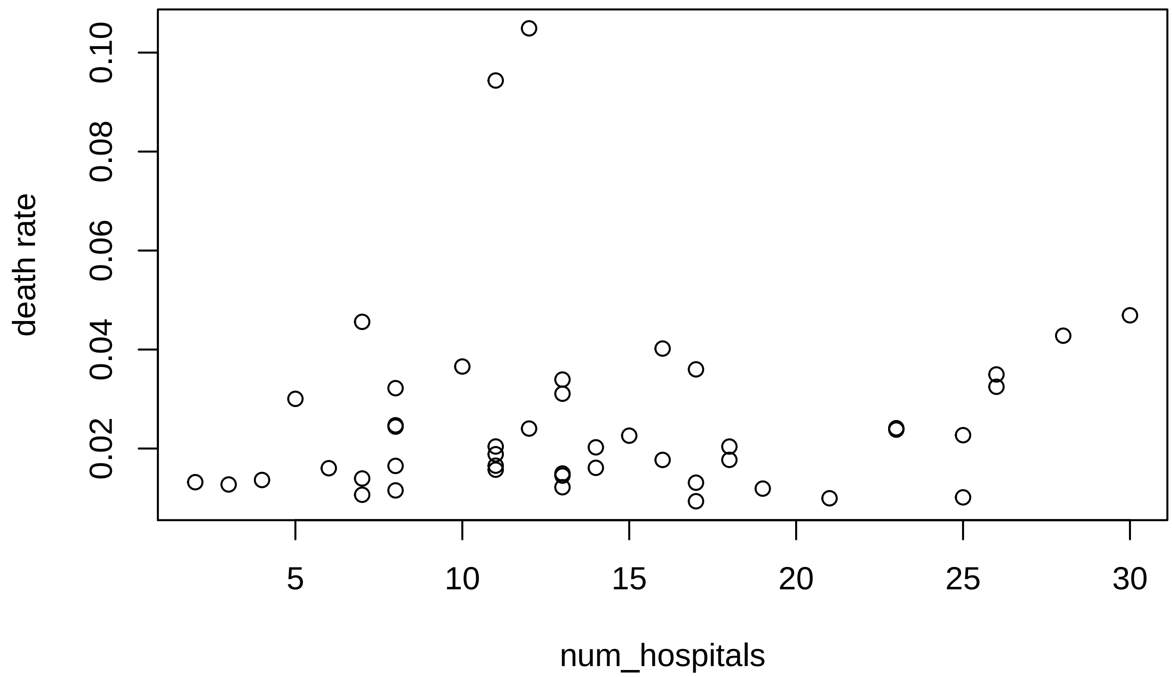

```
##
## Call:
## lm(formula = as.numeric(unlist(covid[, "death_rate"])) ~ as.numeric(unlist(covid[,
##     colname])))
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.016948 -0.011832 -0.005589  0.005551  0.075886
##
## Coefficients:
##                                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)                           0.061069   0.046785   1.305    0.199
## as.numeric(unlist(covid[, colname])) -0.001034   0.001360  -0.760    0.451
##
## Residual standard error: 0.01904 on 44 degrees of freedom
## Multiple R-squared:  0.01296,    Adjusted R-squared:  -0.009474
## F-statistic: 0.5777 on 1 and 44 DF,  p-value: 0.4513
```



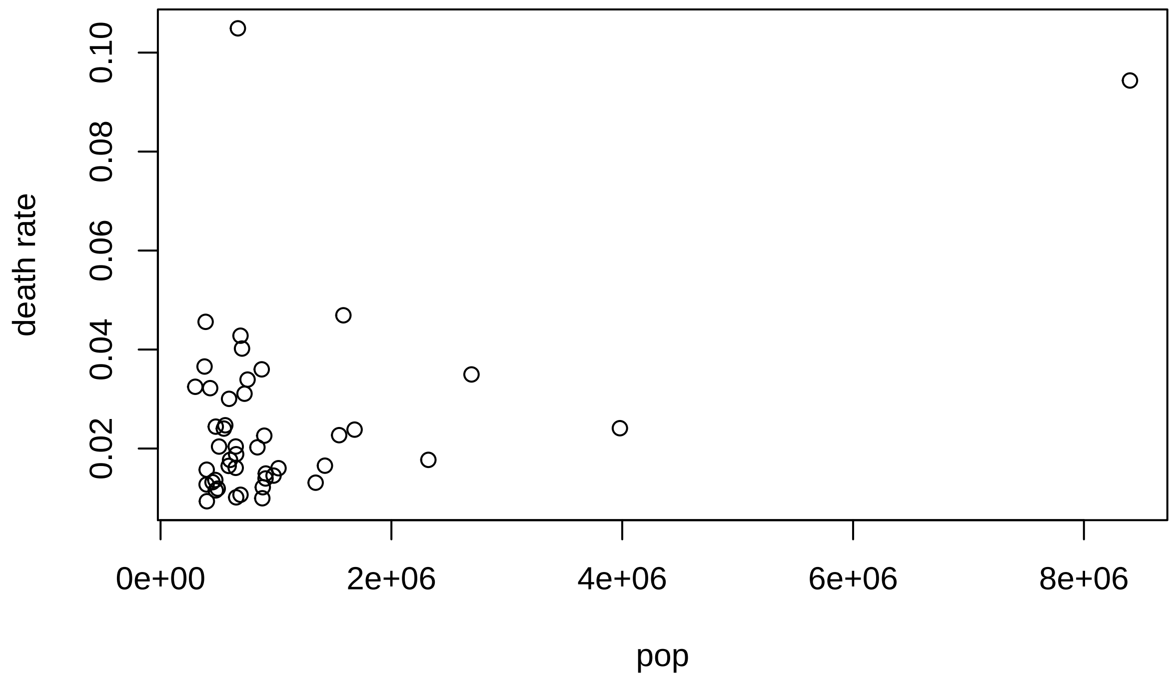

```
##
## Call:
## lm(formula = as.numeric(unlist(covid[, "death_rate"])) ~ as.numeric(unlist(covid[,
##     colname])))
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.019184 -0.011394 -0.004417  0.003755  0.071658
##
## Coefficients:
##                                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)                         4.159e-02  1.041e-02   3.995 0.000243 ***
## as.numeric(unlist(covid[, colname])) -2.831e-07  1.775e-07  -1.595 0.117952
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01863 on 44 degrees of freedom
## Multiple R-squared:  0.05464,    Adjusted R-squared:  0.03315
## F-statistic: 2.543 on 1 and 44 DF,  p-value: 0.118
```



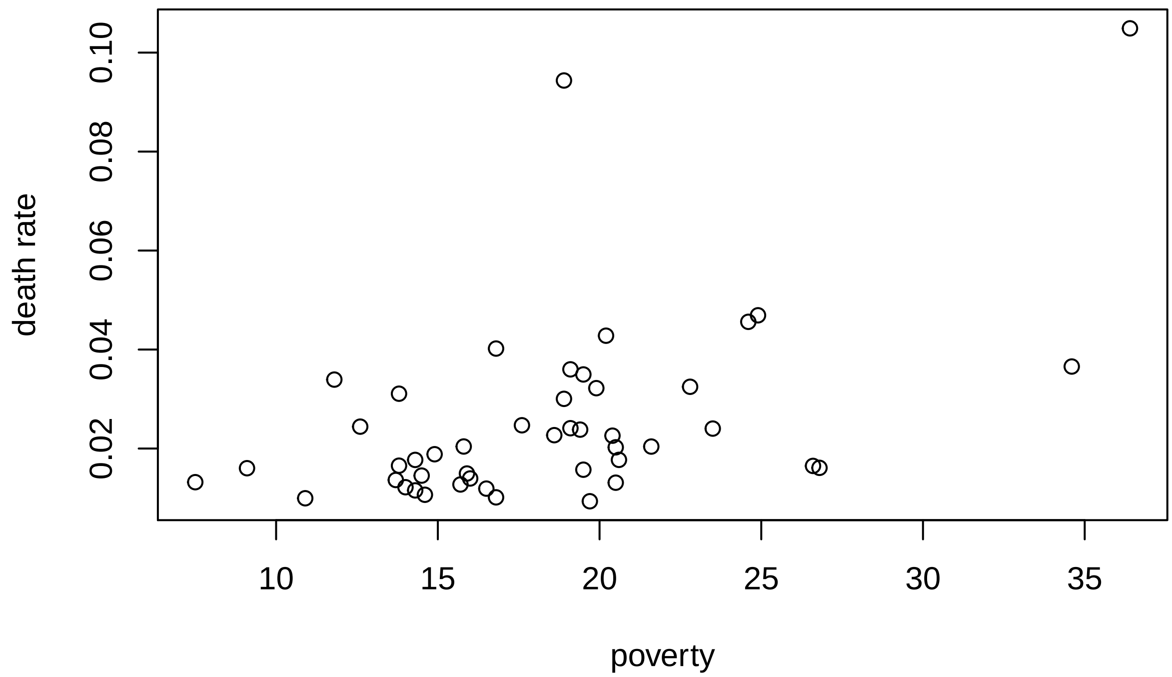

```
##
## Call:
## lm(formula = as.numeric(unlist(covid[, "death_rate"])) ~ as.numeric(unlist(covid[,
##     colname])))
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.018367 -0.009700 -0.005868  0.006094  0.079850
##
## Coefficients:
##                                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)                         0.0218641  0.0063446   3.446  0.00126 **
## as.numeric(unlist(covid[, colname])) 0.0002655  0.0004069   0.652  0.51751
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01907 on 44 degrees of freedom
## Multiple R-squared:  0.009582,   Adjusted R-squared:  -0.01293
## F-statistic: 0.4257 on 1 and 44 DF,  p-value: 0.5175
```



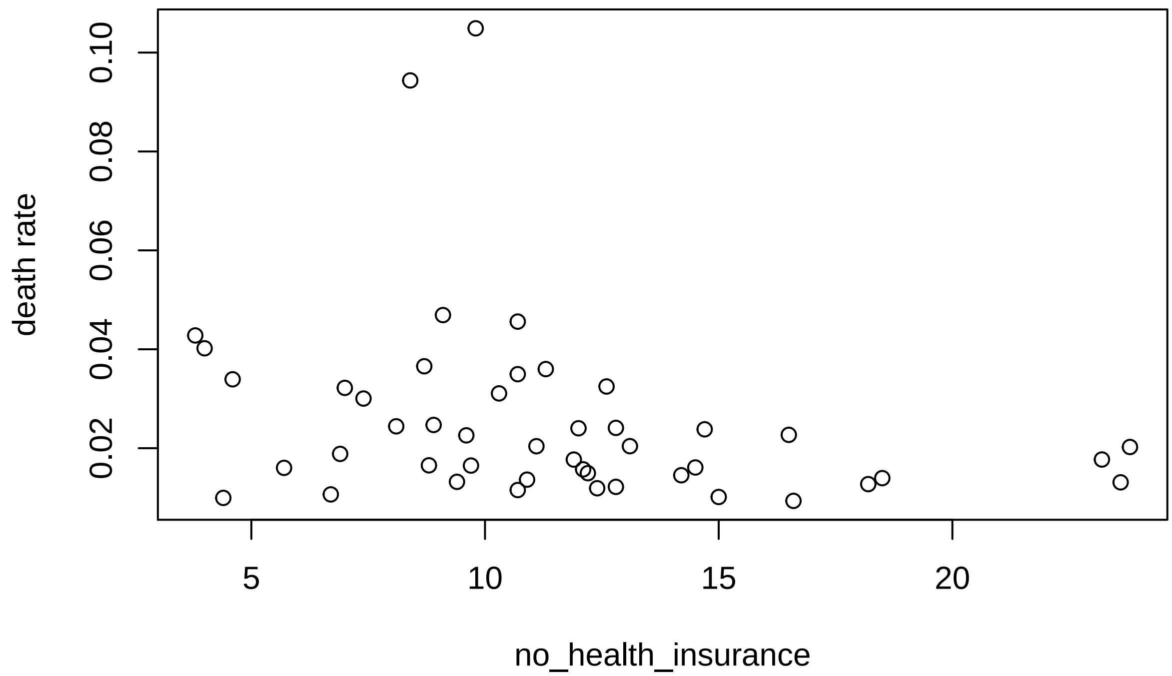

```
##
## Call:
## lm(formula = as.numeric(unlist(covid[, "death_rate"])) ~ as.numeric(unlist(covid[,
##     colname])))
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.021825 -0.009964 -0.005011  0.007754  0.082001
##
## Coefficients:
##                                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)                        1.824e-02  3.215e-03   5.672 1.02e-06 ***
## as.numeric(unlist(covid[, colname])) 6.960e-09  1.936e-09   3.596 0.000814 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01685 on 44 degrees of freedom
## Multiple R-squared:  0.2271, Adjusted R-squared:  0.2095
## F-statistic: 12.93 on 1 and 44 DF,  p-value: 0.0008135
```




```
##
## Call:
## lm(formula = as.numeric(unlist(covid[, "death_rate"])) ~ as.numeric(unlist(covid[,
##     colname])))
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.024108 -0.008776 -0.002915  0.006731  0.067953
##
## Coefficients:
##                                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)                         -0.0066120  0.0083572  -0.791 0.433087
## as.numeric(unlist(covid[, colname]))  0.0017470  0.0004342   4.023 0.000222 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01639 on 44 degrees of freedom
## Multiple R-squared:  0.2689, Adjusted R-squared:  0.2523
## F-statistic: 16.19 on 1 and 44 DF,  p-value: 0.0002224
```




```
##
## Call:
## lm(formula = as.numeric(unlist(covid[, "death_rate"])) ~ as.numeric(unlist(covid[,
##     colname])))
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.023794 -0.010916 -0.003666  0.005322  0.077399
##
## Coefficients:
##                                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)                         0.0388359  0.0071069   5.465 2.05e-06 ***
## as.numeric(unlist(covid[, colname])) -0.0011566  0.0005733  -2.018   0.0498 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01833 on 44 degrees of freedom
## Multiple R-squared:  0.08468,    Adjusted R-squared:  0.06388
## F-statistic: 4.071 on 1 and 44 DF,  p-value: 0.04976
```



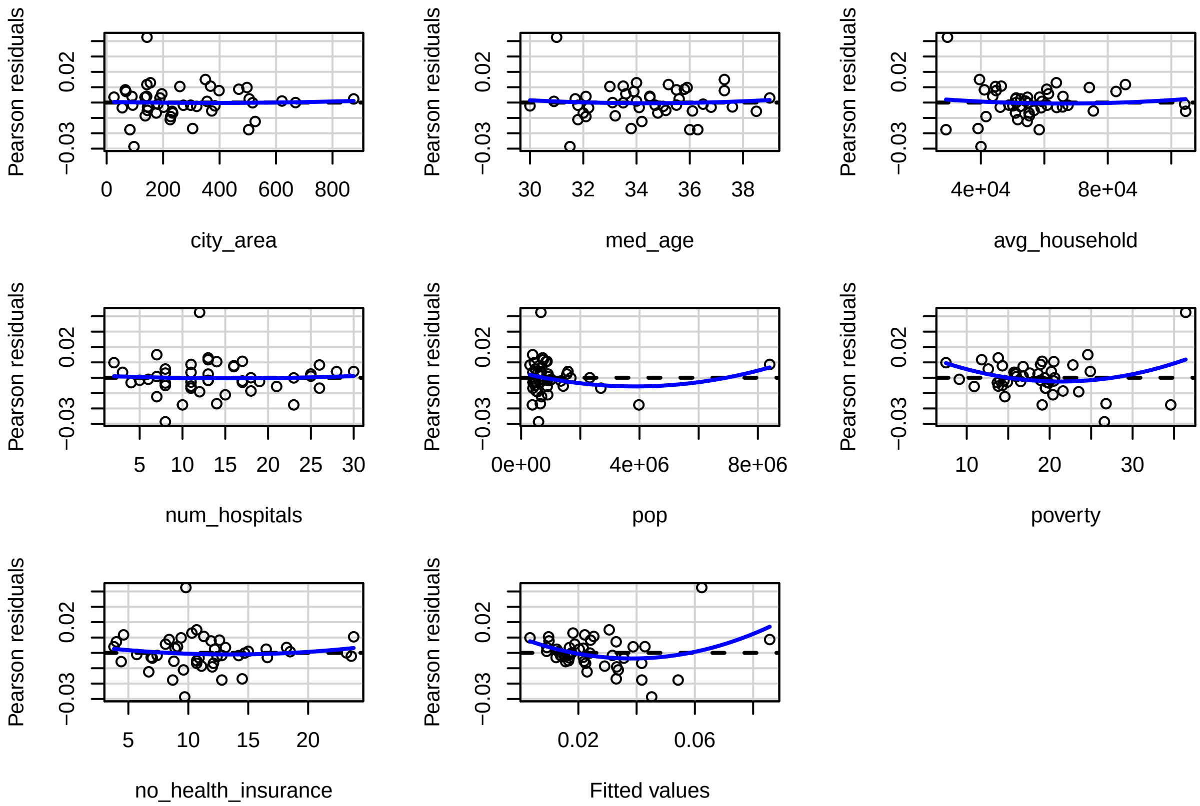
3.4 Residual Plots

```
library(car)
```

```
## Warning: package 'car' was built under R version 4.0.3
```

```
## Loading required package: carData
```

```
library(carData)
citycovid.lm <- lm(death_rate~city_area+med_age+avg_household+num_hospitals+
                   pop+poverty+no_health_insurance, data=covid)
residualPlots(citycovid.lm)
```




```
##                     Test stat Pr(>|Test stat|)
## city_area              0.1336        0.8944814
## med_age                0.2982        0.7671901
## avg_household          0.6872        0.4962144
## num_hospitals          0.2219        0.8255798
## pop                    1.4774        0.1480333
## poverty                2.5976        0.0133942 *
## no_health_insurance    0.8546        0.3982701
## Tukey test             3.3651        0.0007653 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

4. Model Building, Diagnosis, and Validation

4.1 Best Subset Analysis

```r
library(ALSM)
```

```
## Warning: package 'ALSM' was built under R version 4.0.2
```

```
## Loading required package: leaps
```

```
## Loading required package: SuppDists
```

```
## Warning: package 'SuppDists' was built under R version 4.0.2
```

```r
bs <- BestSub(covid[,c(1,3:8)],
              covid$death_rate, num=7)
bs
```

```
##   p 1 2 3 4 5 6 7        SSEp          r2       r2.adj        Cp      AICp
## 1 2 0 0 0 0 0 1 0 0.011812632 0.268931595  0.252316404 40.614401 -376.2924
## 1 2 0 0 0 0 1 0 0 0.012488336 0.227113164  0.209547554 45.340094 -373.7337
## 1 2 0 0 0 0 0 0 1 0.014789765 0.084680719  0.063878008 61.435676 -365.9532
## 1 2 0 0 1 0 0 0 0 0.015275243 0.054635174  0.033149610 64.830973 -364.4675
## 1 2 1 0 0 0 0 0 0 0.015749343 0.025293696  0.003141280 68.146708 -363.0615
## 1 2 0 1 0 0 0 0 0 0.015948656 0.012958507 -0.009474254 69.540646 -362.4830
## 1 2 0 0 0 1 0 0 0 0.016003213 0.009582011 -0.012927489 69.922207 -362.3259
## 2 3 0 0 0 0 1 1 0 0.008245352 0.489705912  0.465971304 17.665795 -390.8304
## 2 3 0 0 0 0 0 1 1 0.009557457 0.408501452  0.380989892 26.842307 -384.0374
## 2 3 1 0 0 0 1 0 0 0.010550822 0.347023383  0.316652378 33.789637 -379.4889
## 2 3 0 0 1 0 0 1 0 0.010709947 0.337175379  0.306346327 34.902511 -378.8003
## 2 3 0 0 0 0 1 0 1 0.010947901 0.322448696  0.290934682 36.566700 -377.7895
## 2 3 0 0 1 0 1 0 0 0.011303885 0.300417339  0.267878611 39.056354 -376.3175
## 2 3 1 0 0 0 0 1 0 0.011748692 0.272888761  0.239069634 42.167222 -374.5421
## 3 4 0 0 0 0 1 1 1 0.005782519 0.642127454  0.616565129  2.441395 -405.1516
## 3 4 1 0 0 0 1 1 0 0.007263204 0.550489772  0.518381898 12.796913 -394.6645
## 3 4 0 0 1 0 1 1 0 0.007601945 0.529525517  0.495920196 15.165979 -392.5677
## 3 4 0 0 1 0 1 0 1 0.007761311 0.519662583  0.485352768 16.280540 -391.6133
## 3 4 0 0 0 1 1 1 0 0.008060546 0.501143352  0.465510734 18.373307 -389.8731
## 3 4 0 1 0 0 1 1 0 0.008221163 0.491202946  0.454860299 19.496622 -388.9655
## 3 4 1 0 1 0 1 0 0 0.009120256 0.435559285  0.395242091 25.784636 -384.1913
## 4 5 0 1 0 0 1 1 1 0.005623285 0.651982204  0.618029248  3.327759 -404.4361
## 4 5 1 0 0 0 1 1 1 0.005653246 0.650127985  0.615994130  3.537295 -404.1917
## 4 5 0 0 0 1 1 1 1 0.005692921 0.647672527  0.613299116  3.814774 -403.8700
## 4 5 0 0 1 0 1 1 1 0.005781471 0.642192318  0.607284252  4.434065 -403.1600
## 4 5 1 0 1 0 1 1 0 0.007049530 0.563713761  0.521149250 13.302536 -394.0380
## 4 5 1 0 1 0 1 0 1 0.007095621 0.560861271  0.518018468 13.624882 -393.7383
## 4 5 0 1 1 0 1 0 1 0.007138481 0.558208739  0.515107152 13.924631 -393.4612
## 5 6 0 1 0 1 1 1 1 0.005509916 0.658998498  0.616373311  4.534882 -403.3730
## 5 6 1 1 0 0 1 1 1 0.005520075 0.658369783  0.615666005  4.605930 -403.2882
## 5 6 1 0 0 1 1 1 1 0.005590680 0.654000100  0.610750112  5.099726 -402.7036
## 5 6 0 1 1 0 1 1 1 0.005622418 0.652035907  0.608540395  5.321690 -402.4432
## 5 6 1 0 1 0 1 1 1 0.005651175 0.650256126  0.606538142  5.522814 -402.2085
## 5 6 0 0 1 1 1 1 1 0.005686980 0.648040226  0.604045254  5.773222 -401.9180
## 5 6 1 1 1 0 1 0 1 0.006650949 0.588381443  0.536929123 12.514964 -394.7153
## 6 7 1 1 0 1 1 1 1 0.005435653 0.663594542  0.611839856  6.015506 -401.9972
## 6 7 0 1 1 1 1 1 1 0.005509657 0.659014512  0.606555206  6.533073 -401.3751
## 6 7 1 1 1 0 1 1 1 0.005510815 0.658942854  0.606472524  6.541170 -401.3655
## 6 7 1 0 1 1 1 1 1 0.005590680 0.654000109  0.600769357  7.099725 -400.7036
## 6 7 1 1 1 1 1 0 1 0.006650907 0.588384042  0.525058510 14.514671 -392.7156
## 6 7 1 1 1 1 1 1 0 0.006925073 0.571416269  0.505480310 16.432114 -390.8574
## 6 7 1 1 1 1 0 1 1 0.009108315 0.436298293  0.349574953 31.701124 -378.2516
## 7 8 1 1 1 1 1 1 1 0.005433436 0.663731760  0.601787610  8.000000 -400.0160
##        SBCp     PRESSp
```

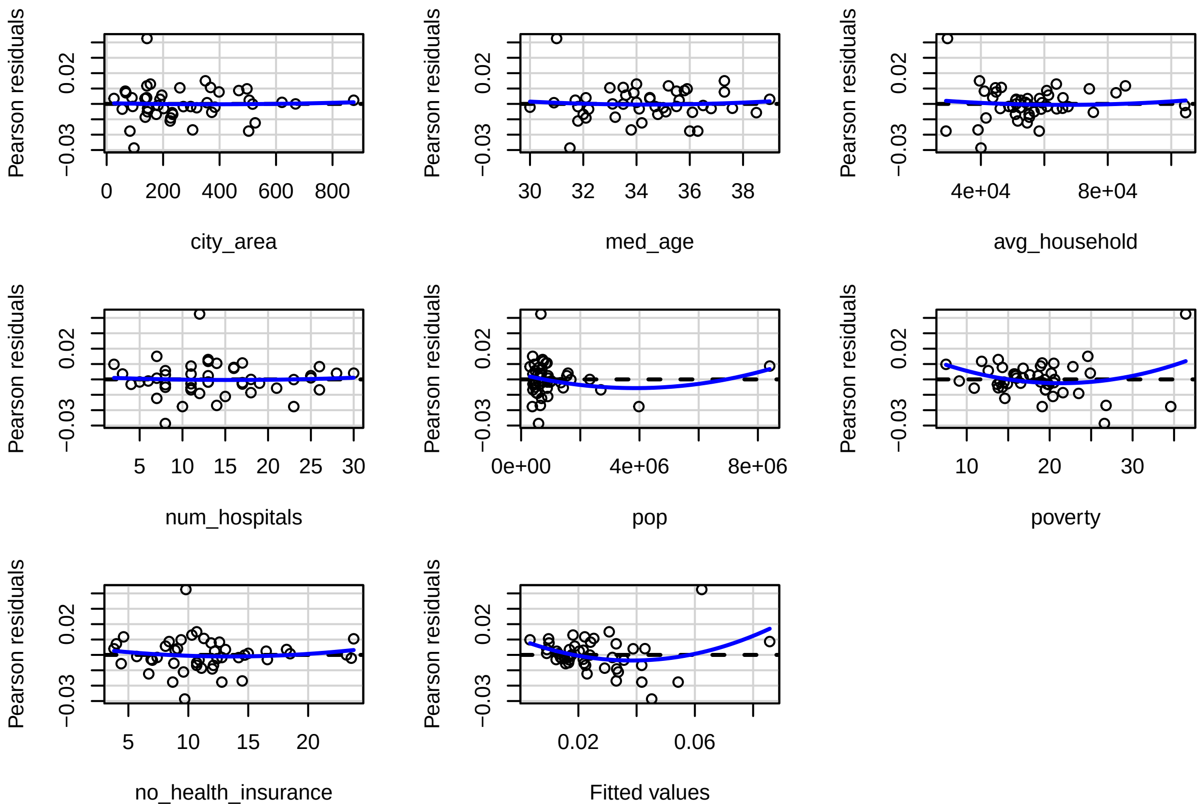
```
## 1 -372.6352 0.014385935
## 1 -370.0764 0.017337856
## 1 -362.2959 0.015772220
## 1 -360.8102 0.017009729
## 1 -359.4042 0.017015018
## 1 -358.8257 0.017826050
## 1 -358.6686 0.017097967
## 2 -385.3444 0.015046803
## 2 -378.5515 0.012182221
## 2 -374.0029 0.013551677
## 2 -373.3144 0.013709472
## 2 -372.3035 0.014251943
## 2 -370.8316 0.016863381
## 2 -369.0562 0.014768350
## 3 -397.8371 0.010267488
## 3 -387.3499 0.012876421
## 3 -385.2531 0.015281800
## 3 -384.2987 0.010593611
## 3 -382.5585 0.015664265
## 3 -381.6509 0.015566875
## 3 -376.8768 0.012618241
## 4 -395.2929 0.010666868
## 4 -395.0485 0.010063796
## 4 -394.7268 0.010599170
## 4 -394.0168 0.010926533
## 4 -384.8948 0.014100885
## 4 -384.5951 0.009466575
## 4 -384.3180 0.010994474
## 5 -392.4011 0.010948125
## 5 -392.3164 0.010601195
## 5 -391.7318 0.010486519
## 5 -391.4714 0.011205510
## 5 -391.2367 0.010755547
## 5 -390.9461 0.011276188
## 5 -383.7435 0.010046504
## 6 -389.1967 0.010996642
## 6 -388.5746 0.011511964
## 6 -388.5650 0.011165283
## 6 -387.9031 0.011264869
## 6 -379.9151 0.010810330
## 6 -378.0569 0.015502472
## 6 -365.4511 0.013949089
## 7 -385.3868 0.011655493
```

The subset analysis suggests that we should drop average household income, however we feel that it is a very important variable that affects the COVID death rate so we are opting to leave it in.

4.2 Diagnosis Analysis

4.2.1 Linearity Assumption

```r
residualPlots(citycovid.lm)
```

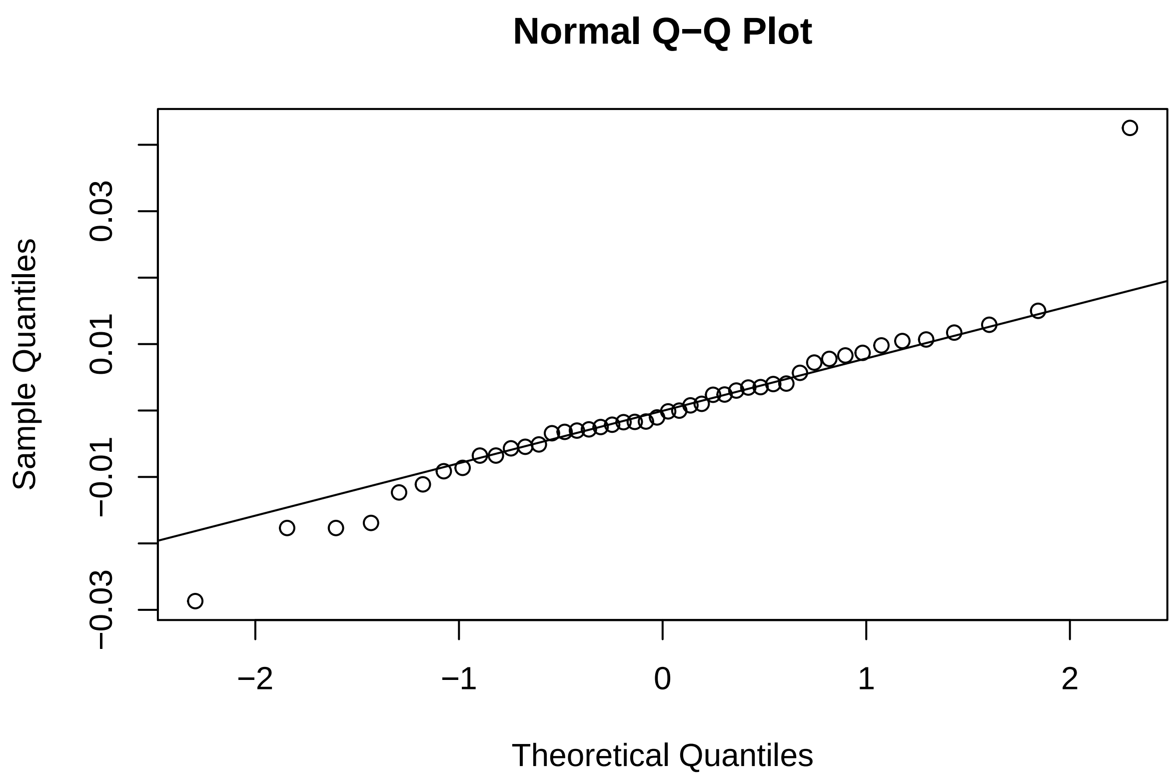



```
##                     Test stat Pr(>|Test stat|)
## city_area              0.1336        0.8944814
## med_age                0.2982        0.7671901
## avg_household          0.6872        0.4962144
## num_hospitals          0.2219        0.8255798
## pop                    1.4774        0.1480333
## poverty                2.5976        0.0133942 *
## no_health_insurance    0.8546        0.3982701
## Tukey test             3.3651        0.0007653 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The blue line at approximately y = 0 in the residual plot for poverty does not seem to be horizontal, which indicates that they may be violating the linearity assumption. For population we cannot draw any conclusion about linearity as the observations are concentrated in the left, so a linear transformation will be made later for only poverty.

4.2.2 Normality Assumption

```r
qqnorm(citycovid.lm$residuals)
qqline(citycovid.lm$residuals)
```




```r
shapiro.test(citycovid.lm$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  citycovid.lm$residuals
## W = 0.92235, p-value = 0.004527
```

The QQ plot becomes less stable as the theoretical quantities deviate from 0. The assumption of normality might be violated. The shapiro test is conducted and since the p-value is less than the 0.05. So we say the data doesn't follow a normal distribution at a significant level of 0.05.

4.2.3 Constant Variance Assumption

```r
library(onewaytests)
```

```
## Warning: package 'onewaytests' was built under R version 4.0.2
```

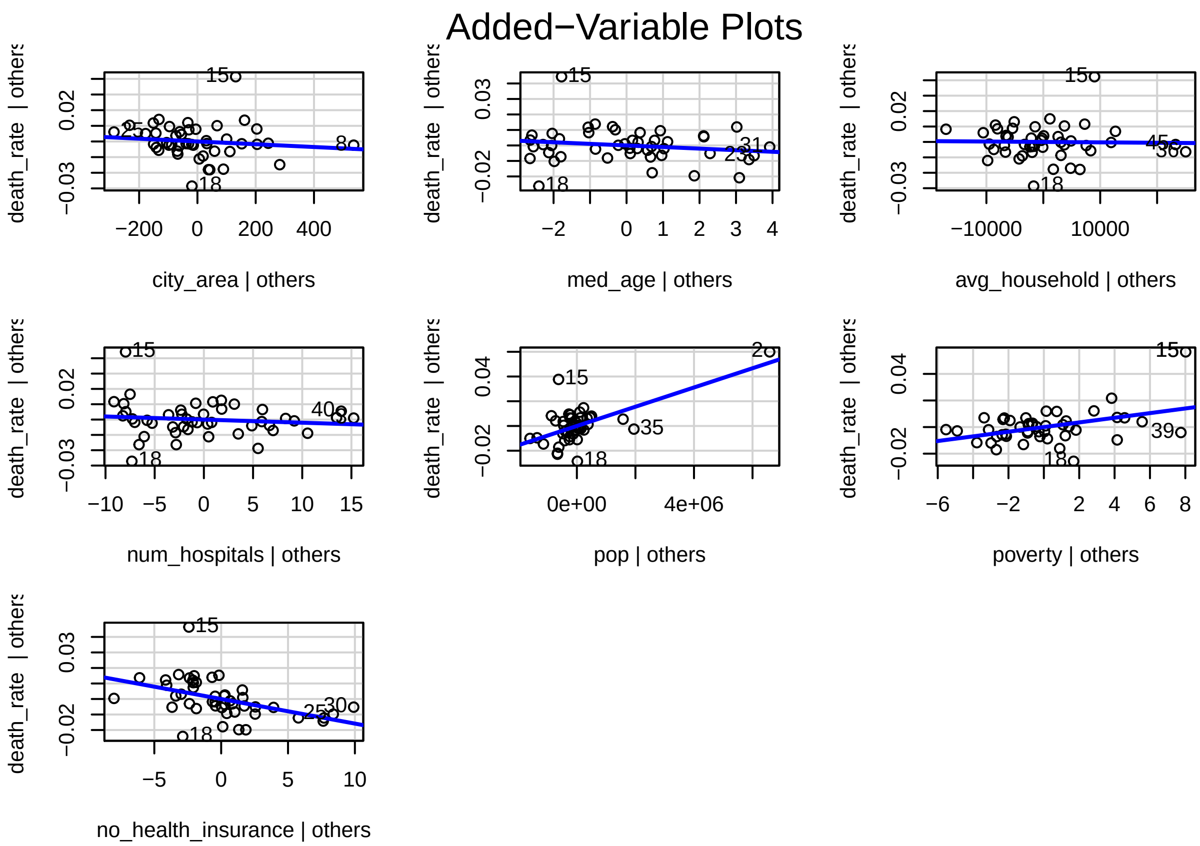
```r
covid$fit <- citycovid.lm$fitted.values
covid$resid <- citycovid.lm$residuals
covid$group <- cut(covid$fit, 3)
bf.test(resid ~ group, covid)
```

```
##
##   Brown-Forsythe Test (alpha = 0.05)
## -------------------------------------------------------------
##   data : resid and group
##
##   statistic  : 3.35649
##   num df     : 2
##   denom df   : 1.396221
##   p.value    : 0.2928377
##
##   Result     : Difference is not statistically significant.
## -------------------------------------------------------------
```

From the Brown-Forsythe test result, we could conclude that there is no violation for constant variance assumption.

4.2.4 Added Variable Plots

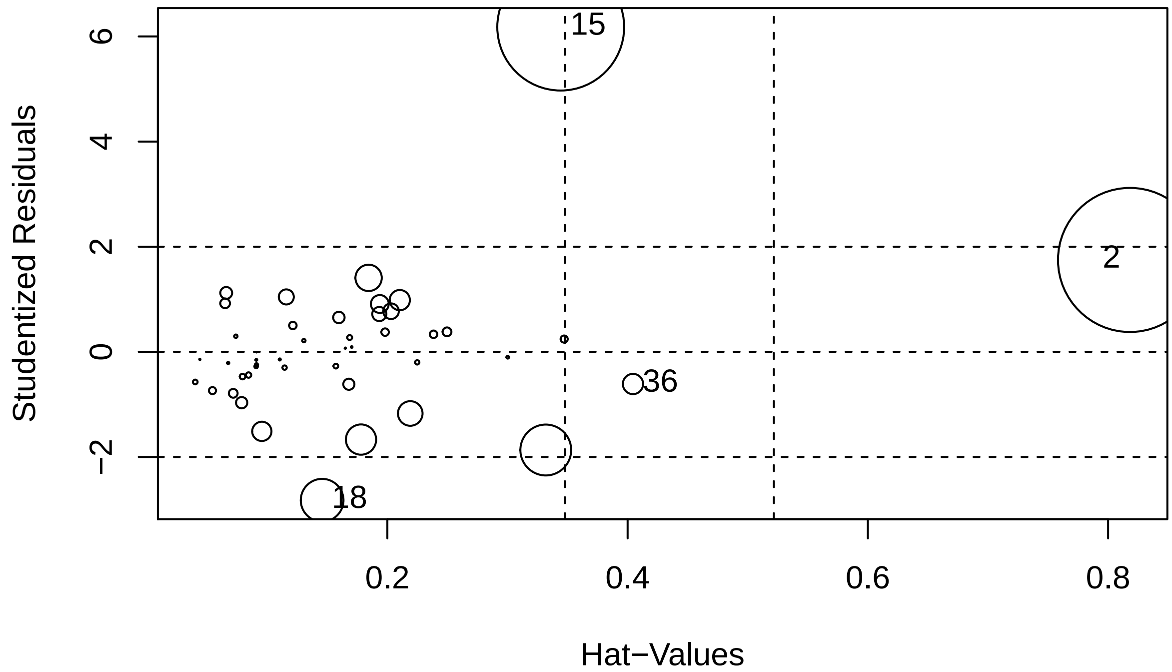
```r
avPlots(citycovid.lm)
```



The variables in this plot suggests that a lot of variables most variables need to be removed except population, poverty, and health insurance. However, this may change after the necessary transformations happen.

4.2.5 Identify Y outliers

```
influencePlot(citycovid.lm)
```




```
##       StudRes       Hat      CookD
## 2   1.7479916 0.8181951 1.63065358
## 15  6.1789091 0.3443144 1.26672003
## 18 -2.8233293 0.1456934 0.14358478
## 36 -0.6115722 0.4044568 0.03228331
```

```
n <- nrow(covid)
qt(1-(0.05/n*2),n-1-8)
```

```
## [1] 3.038246
```

```
sort(rstudent(citycovid.lm))
```

```
##           18           39           35           11           12           14
## -2.823329345 -1.867412115 -1.668890221 -1.512592730 -1.172392030 -0.967143552
```

```
##          43          46          17          36           9          41 
## -0.788780079 -0.737691482 -0.616531740 -0.611572206 -0.573204125 -0.470521737 
##          20          19           7          23          37          21 
## -0.441307995 -0.300044173 -0.278340070 -0.271585791 -0.245594475 -0.212299143 
##          24          29           4          42          45          33 
## -0.200226921 -0.151820815 -0.145076971 -0.144383639 -0.102383665 -0.010811438 
##          27          26          22          28           8          31 
## -0.001276771  0.070133761  0.091607496  0.213246429  0.242501713  0.272360831 
##          34          25           3           1          44           5 
##  0.297078139  0.333590129  0.374587013  0.380280031  0.501246683  0.653459362 
##          13          40           6          16          30          38 
##  0.719324804  0.772696382  0.910592128  0.923043614  0.983863992  1.044400581 
##          32          10           2          15 
##  1.120839658  1.407258560  1.747991613  6.178909067
```

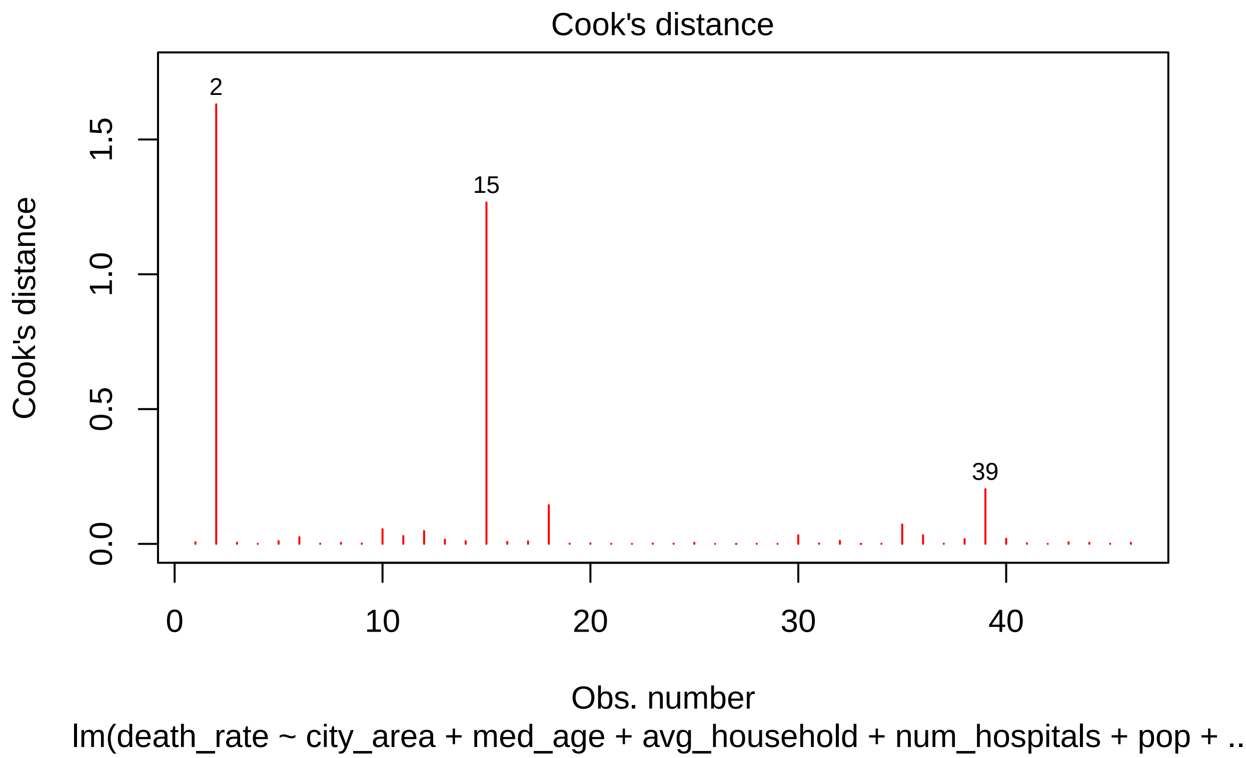
Case 15 is an outlying Y observation because 6.1789 > 3.038246

4.2.6 Cook's Distance or Identifying Influential points

```r
sort(cooks.distance(citycovid.lm))
```

```
##           27           33           42           26           22           29 
## 6.974581e-08 1.503306e-06 1.229399e-04 1.247370e-04 2.210946e-04 2.956716e-04 
##            4           21           45           37           28           34 
## 3.352460e-04 4.182341e-04 5.771239e-04 7.742293e-04 8.752885e-04 9.014262e-04 
##            7           19           24            9           23           31 
## 9.905477e-04 1.489975e-03 1.491324e-03 1.745432e-03 1.761506e-03 1.927451e-03 
##           20           41           46            8           44            3 
## 2.293704e-03 2.436977e-03 3.961268e-03 4.008680e-03 4.424102e-03 4.433796e-03 
##           25           43            1           16           17           14 
## 4.459379e-03 6.065158e-03 6.152513e-03 7.424042e-03 9.748784e-03 1.000662e-02 
##            5           32           13           38           40            6 
## 1.029440e-02 1.099426e-02 1.569751e-02 1.782428e-02 1.921792e-02 2.501360e-02 
##           11           30           36           12           10           35 
## 2.920777e-02 3.225641e-02 3.228331e-02 4.772322e-02 5.455355e-02 7.203701e-02 
##           18           39           15            2 
## 1.435848e-01 2.032294e-01 1.266720e+00 1.630654e+00
```

```r
plot(citycovid.lm, pch = 18, col="red", which=c(4))
```




```
minor <- qf(0.2,8,n-8)
major <- qf(0.5,8,n-8)
```

Case 2 and Case 15 are influential points because those points are above the threshold (0.9344).

4.2.7 Identify X outliers

```
sort(lm.influence(citycovid.lm)$hat)
```

```
##          9         42         46         16         32         21         43
## 0.04007462 0.04394336 0.05440502 0.06492861 0.06584277 0.06748621 0.07167697
##         34         14         41         20          7         29         37
## 0.07385983 0.07871380 0.07940698 0.08443652 0.09074565 0.09089493 0.09103247
##         33         11          4         19         38         44         28
## 0.09105952 0.09550502 0.11043336 0.11444290 0.11585801 0.12133662 0.13052307
##         18         23          5         26         17         31         22
## 0.14569339 0.15711310 0.15963358 0.16497122 0.16793583 0.16860786 0.17030950
##         35         10         13          6          3         40         30
## 0.17806060 0.18438058 0.19330012 0.19371032 0.19812120 0.20304145 0.21033549
##         12         24         25          1         27         45         39
## 0.21905517 0.22484933 0.23843387 0.24964138 0.24996605 0.30020102 0.33188206
##         15          8         36          2
## 0.34431437 0.34718438 0.40445677 0.81819510
```

$\frac{2 \times p}{n} = \frac{18}{46} \approx 0.39$

Case 36 and case 2 are outlying X observations because they are both greater than 0.39.

4.2.8 Variation Inflation Factor

```
library(fmsb)
library(olsrr)
#VIF(citycovid.lm)
ols_vif_tol(citycovid.lm)
```

```
##             Variables Tolerance      VIF
## 1           city_area 0.6331427 1.579423
## 2             med_age 0.7710011 1.297015
## 3       avg_household 0.2735454 3.655701
## 4       num_hospitals 0.8565866 1.167424
## 5                 pop 0.8013099 1.247957
## 6             poverty 0.2792373 3.581183
## 7 no_health_insurance 0.5807057 1.722043
```

Since the VIF < 10, multicollinearity is not much of an issue. However, these VIF values reflect the correlation between the i^{th} variable and the rest of the variables. To examine the relationship between poverty and average household income, separate VIF values must be calculated.

```
VIF(lm(death_rate~avg_household, data = covid))
```

```
## [1] 1.057793
```

```
VIF(lm(death_rate~poverty , data = covid))
```

```
## [1] 1.367861
```

```
VIF(lm(death_rate~avg_household+poverty, data = covid))
```

```
## [1] 1.508695
```

These VIF values show that there isn't any concerning issues of multicollinearity between poverty and average household income.

4.3 Remedial Measures

4.3.1 X transformation on Poverty

```
covid$Tpov <- (covid$poverty)^2
covidmodel2 <- lm(death_rate~city_area+med_age+avg_household+num_hospitals+
                      pop+Tpov+no_health_insurance, data=covid)
```

The relationship between poverty and death rate seemed to be non-linear and so, a transformation was necessary. The residual graph appears concave, and so, an the poverty variable was squared and regression was remodeled.

4.3.2 Y transformation on death rate

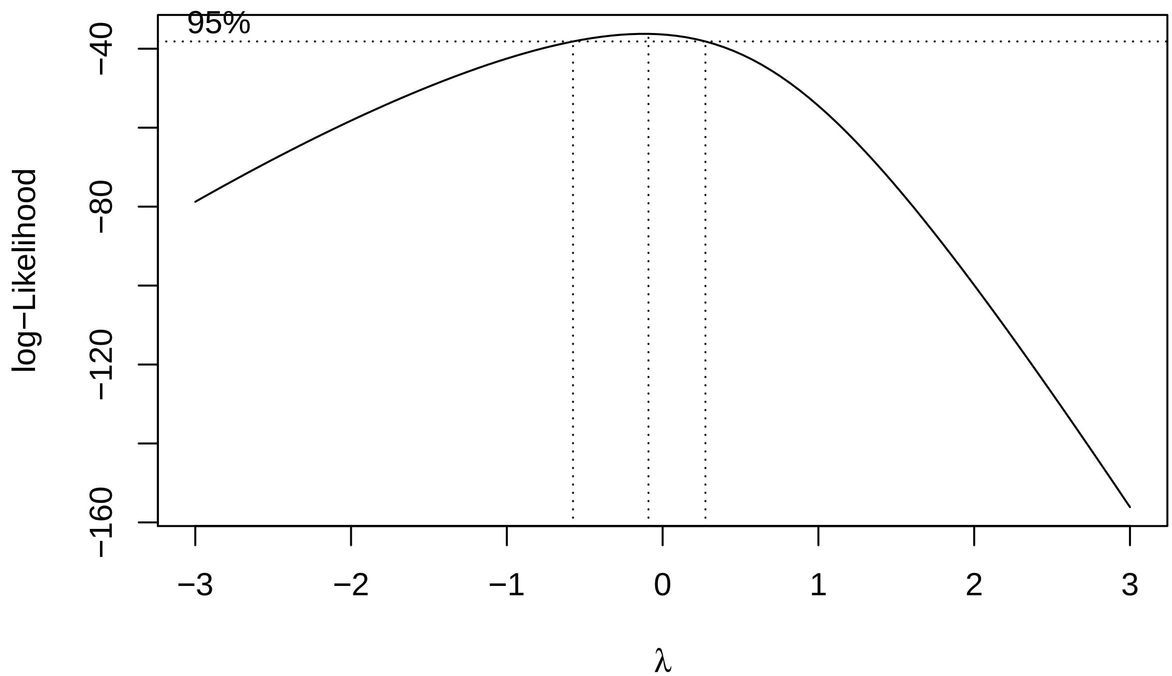
```r
library(MASS)
```

```
## Warning: package 'MASS' was built under R version 4.0.3

##
## Attaching package: 'MASS'

## The following object is masked from 'package:olsrr':
##
##     cement
```

```r
bcmle <- boxcox(covidmodel2, lambda= seq(-3, 3, by=0.1))
```




```r
lambda<-bcmle$x[which.max(bcmle$y)]
lambda
```

```
## [1] -0.09090909
```

```r
covid$Tdeath_rate <- (covid$death_rate)^lambda
covidmodel2 <- lm(Tdeath_rate~city_area+med_age+avg_household+num_hospitals+
                      pop+Tpov+no_health_insurance, data=covid)
```

This transformation was later done to fix the normality issue that the model was facing. A box cox transformation was done to identify an optimal lambda using Maximum likelihood. The death rate was transformed according to this lambda and regression was remodeled.

The Constant variance test and normality tests are done on the model again.

```
shapiro.test(covidmodel2$residuals)
```

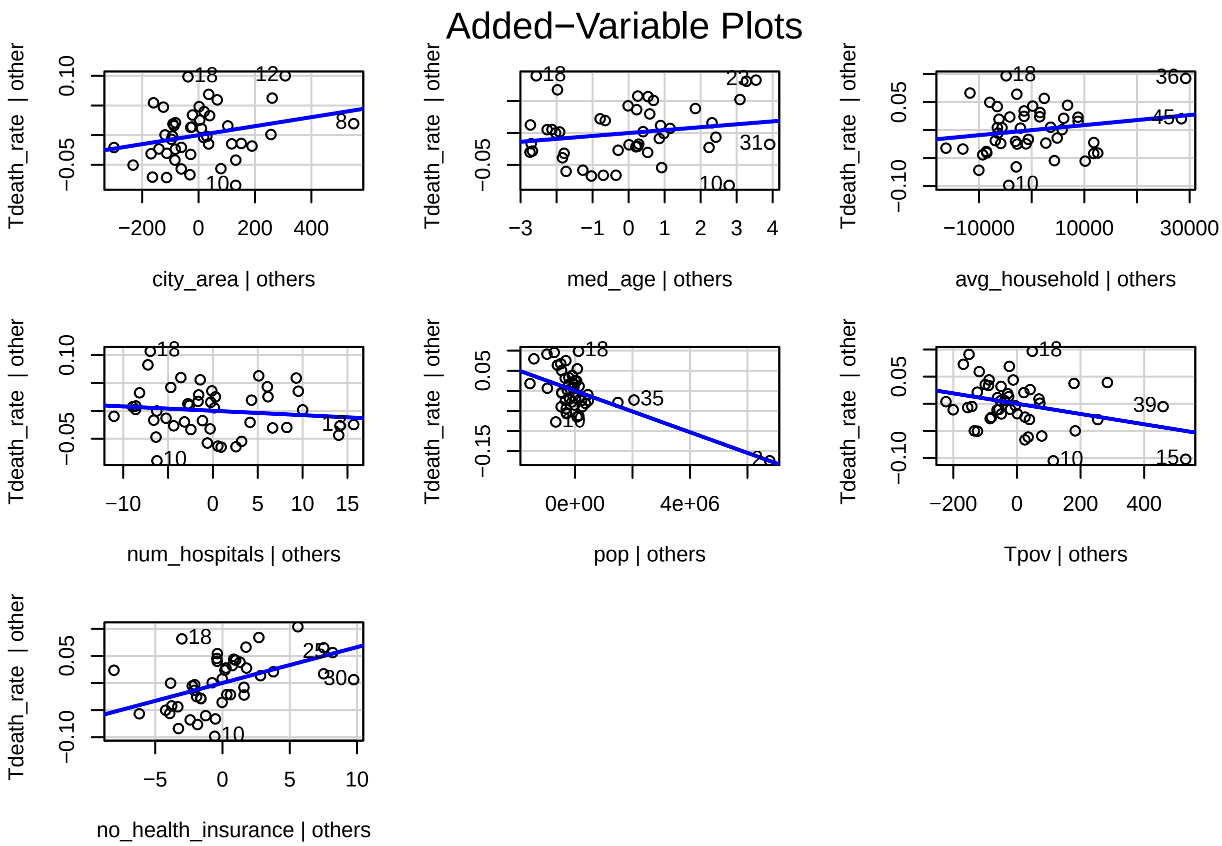
```
##
##  Shapiro-Wilk normality test
##
## data:  covidmodel2$residuals
## W = 0.99099, p-value = 0.9751
```

```
covid$fit <- covidmodel2$fitted.values
covid$resid <- covidmodel2$residuals
bf.test(resid ~ group, covid)
```

```
##
##   Brown-Forsythe Test (alpha = 0.05)
## -------------------------------------------------------------
##   data : resid and group
##
##   statistic  : 1.113229
##   num df     : 2
##   denom df   : 5.172442
##   p.value    : 0.3962016
##
##   Result     : Difference is not statistically significant.
## -------------------------------------------------------------
```

Normality and Constant variance is maintained in this new models with transformed variables. The Added-Variable Plots are produced again to see if these issues of multicollinearity (evidenced in 4.2.4) still persist.

```
avPlots(covidmodel2)
```



All the plots now have a significant slope which provides evidence of lack of multicollinearity. However, this issue can be further explored using Type II Anova

4.4 Type II Anova

```
Anova(covidmodel2, type="II")
```

```
## Anova Table (Type II tests)
##
## Response: Tdeath_rate
##                       Sum Sq Df F value    Pr(>F)
## city_area           0.005942  1  2.8128  0.101722
## med_age             0.003202  1  1.5155  0.225866
## avg_household       0.003143  1  1.4879  0.230058
## num_hospitals       0.001210  1  0.5730  0.453743
## pop                 0.041534  1 19.6607 7.638e-05 ***
## Tpov                0.009504  1  4.4990  0.040497 *
## no_health_insurance 0.025513  1 12.0771  0.001292 **
## Residuals           0.080276 38
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Based on the anova table, we conclude that the contribution of population, transformed poverty and percentage of population without health insurance doesn't contribute much to the variance of Y. Thus, this will reject the Null hypothesis stated in the beginning.

4.5 Validation

4.5.1 Full Model

```
set.seed(123)
library(caret)
```

```
## Loading required package: lattice

##
## Attaching package: 'lattice'

## The following object is masked from 'package:ALSM':
##
##     oneway

## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 4.0.3

## Registered S3 methods overwritten by 'pROC':
##   method    from
##   print.roc fmsb
##   plot.roc  fmsb
```

```
train.control <- trainControl(method = "cv", number = 10)

step.model1 <- train(Tdeath_rate~city_area+med_age+avg_household+num_hospitals+
                     pop+Tpov+no_health_insurance, data=covid, method = "leapBackward",
                     tuneGrid = data.frame(nvmax=8),
                     trControl=train.control)

step.model1$results
```

```
##   nvmax       RMSE  Rsquared        MAE     RMSESD RsquaredSD      MAESD
## 1     8 0.04810414 0.6054523 0.04132268 0.01457145  0.2925331 0.01403206
```

The full Model, after 10-fold cross validation, appears to have an RMSE of 0.048 and R^2 of 0.6.

4.5.2 Model without population

Population, based on the Type II Anova, appeared to be insignificant in the model and can be removed.

```
step.model2 <- train(Tdeath_rate~city_area+med_age+avg_household+num_hospitals+
                     Tpov+no_health_insurance, data=covid, method = "leapBackward",
                     tuneGrid = data.frame(nvmax=8),
                     trControl=train.control)

step.model2$results
```

```
##   nvmax       RMSE  Rsquared        MAE     RMSESD RsquaredSD      MAESD
## 1     8 0.05911325 0.4675216 0.04750748 0.02053167  0.3044544 0.01489546
```

After dropping the variable population, the R2 decreased by 0.06.

4.5.2 Model without population, poverty, and percentage with health insurance

```
step.model3 <- train(Tdeath_rate~city_area+med_age+avg_household+num_hospitals, data=covid, method = "l
                     tuneGrid = data.frame(nvmax=8),
                     trControl=train.control)

step.model3$results
```

```
##   nvmax       RMSE Rsquared        MAE     RMSESD RsquaredSD      MAESD
## 1     8 0.06529534  0.38002 0.05550963 0.03250252  0.3463777 0.02659781
```

We then again tried dropping all the variables that seemed insignificant in the Anova analysis however, the Rsquared decreased by 0.3 because of it.

Based on our Type II Anova and the articles we researched online, we believed that percentage with health insurance, poverty, and population can be removed from the model because of its insignificance. However, the preliminary analysis, Variation Inflation Factors, Added- Vairable Plots, and Cross validation prove otherwise, implying that all variables should be kept in the model. Therefore, the final model contains all variables with the transformed poverty and transformed death rate.

5. Conclusion

The preliminary analysis identified some multicollinearity issues between poverty and average household income. However, the Variation Inflation Factor appeared to show that both variables aren't that correlated with one and other. The Diagnostics showed some issues with the linearity assumption for the poverty variable and an X transformation (squared) was done to fix it. The normality assumption was also violated by the first order model so a box cox transformation was used to transform Y. These transformations didn't violate the normality, linearity and constant variance assumption. The Added Variable plots for the new model also showed that all variables in the model were significant.

The research question was answered by the Type II analysis that showed that some of the variables like population, poverty and health insurance were not contributing much to the explanation of Y variance. Previous research also supported this notion stating that poverty and healthcare index were not very important. However, our final model managed to keep all the variables because the exclusion of those variables led to significant decrease in the R Squared of the model.

The reason that there are factors that do not contribute to the death rate based off of our analysis while still being ones we consider important or ones that were not removed in the best subset analysis is because the model as a whole does not have an extremely high R2 value. There is not a perfect correlation between all the variables so the addition or removal of one doesn't always cause a meaningful change to the regression.

We were able to compare our results to a published article to compare conclusions. From the published article, it was shown that population density, testing rate, airport traffic, and high age groups emerge as the most significant variables, while healthcare index, homelessness, and GDP have small impacts. This is consistent with our own data as the healthcare index is the percent without health insurance and the GDP/homelessness is very comparable to the poverty rate.

Some useful changes that we could have done to our data to get a better prediction for our data would be to consolidate the city area variable into population and number of hospitals. We believe that being able to see population and hospital density would give a better prediction towards the death rate.