**Machine Learning Using R: An Approach for Predicting COVID-19 Case Increase**

**Yitong Chen, Zhiyuan Yang**

**Contents**

[1. Introduction 1](#_Toc49966889)

[1.1 Problem Explanation and Importance 1](#_Toc49966890)

[1.2 Model Selection and Potential Evaluation Metrics 1](#_Toc49966891)

[1.3 Previous Work Done and Domain Importance 1](#_Toc49966892)

[2. Data Preparation and Exploratory Data Analysis 2](#_Toc49966893)

[2.1 Data Source 2](#_Toc49966894)

[2.2 Data Preprocessing 2](#_Toc49966895)

[2.3 Exploratory Data Analysis 3](#_Toc49966896)

[3. Feature Engineering and Selection 4](#_Toc49966897)

[3.1 Transformation 4](#_Toc49966898)

[3.2 Feature Selection 5](#_Toc49966899)

[4. Modelling 6](#_Toc49966900)

[4.1 Multiple Linear Regression Model 6](#_Toc49966901)

[4.2 Generalized Additive Models (GAMs) 8](#_Toc49966902)

[4.3 Ridge Regression and The Lasso Models 9](#_Toc49966903)

[4.4 Random Forest Model 11](#_Toc49966904)

[5. Conclusion 12](#_Toc49966905)

[5.1 Model Selection 12](#_Toc49966906)

[5.2 Lessons Learned / Policies 13](#_Toc49966907)

[5.3 Potential Next Steps 13](#_Toc49966908)

[Appendix I - Formula and Variable Definition 14](#_Toc49966909)

[Appendix II - Summary Statistics Table 16](#_Toc49966910)

[Appendix III - Reference 16](#_Toc49966911)

[Appendix IV – Code 16](#_Toc49966912)

# **1. Introduction**

## **1.1 Problem Explanation and Importance**

Novel Coronavirus 2019 (COVID-19) is a respiratory illness that can spread from person to person. The first case in the U.S. was reported on Jan. 21, 2020. As of August 2020, the COVID-19 pandemic continues to take its toll nationwide, with more than 6 million confirmed infections and more than 183,000 deaths.

As a global crisis, the coronavirus pandemic draws attention from scientists and health professionals across the world. The problem we are going to solve is to predict COVID-19 cases increasing in the U.S. by using several machine learning models based on 20 factors including cases, hospitalizations, demographic index, mobility rates and races. This purpose of the study is to explore the community transmission of coronavirus and measure the effectiveness of policy in the U.S.

## **1.2 Model Selection and Potential Evaluation Metrics**

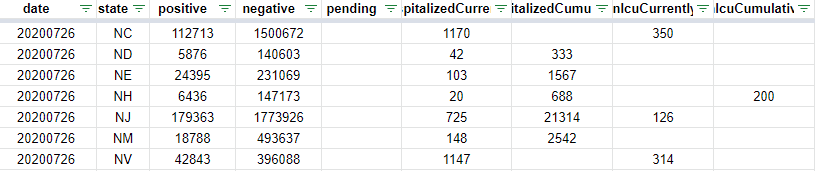
The study is aimed to solve a regression problem since the dependent variable is quantitative. We will construct multiple linear regression, GAM, ridge regression, the lasso and random forest models. The two main metrics to evaluate the models are MSE and R-squared, and MSE is our key evaluation metric.

## **1.3 Previous Work Done and Domain Importance**

Both the government and academia have done a lot of work in predicting COVID-19 by machine learning, such as [forecast deaths conducted by CDC](https://www.cdc.gov/coronavirus/2019-ncov/covid-data/forecasting-us.html), [prediction in cases and deaths via sensitive analysis](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7176069/), [predict the impact of COVID-19 with machine learning models](https://www.nature.com/articles/s41560-020-0662-1), etc. Predicting COVID-19 case increasing is significant in the larger domain space because it is one of the most intuitive data to be understood by the public.

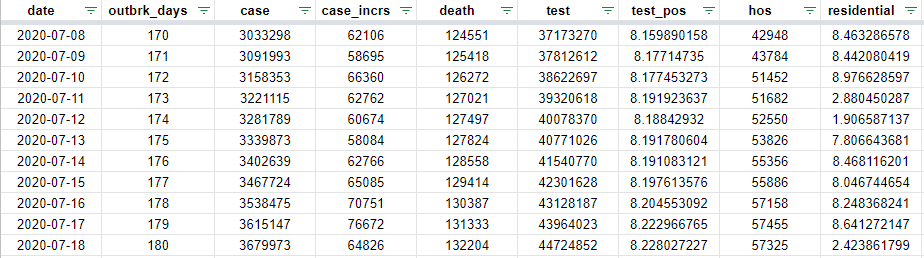
# **2. Data Preparation and Exploratory Data Analysis**

## **2.1 Data Source**

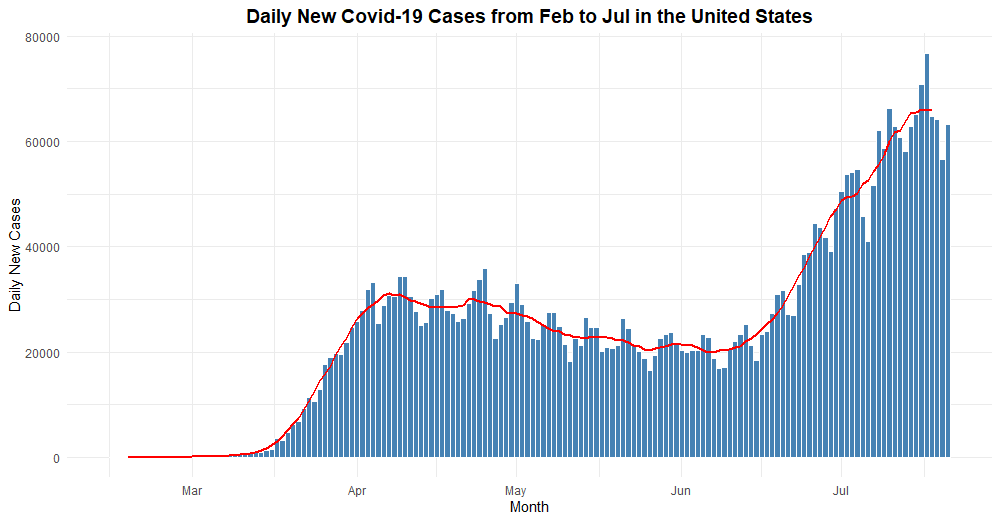
We collect four datasets from multiple resources: case and hospitalizations from the [covid-19 tracking project](https://covidtracking.com/data/download), social vulnerability index including poverty rate, uninsured rate, and other factors from [data.cdc.gov](https://data.cdc.gov/Health-Statistics/Social-Vulnerability-Index-2018-United-States-coun/48va-t53r), mobility data from [Google covid-19 mobility reports](https://www.google.com/covid19/mobility/), unemployment rate from [Investing.com](https://www.investing.com/economic-calendar/unemployment-rate-300) and ethnicity of deaths from [APM research lab](https://www.apmresearchlab.org/covid/deaths-by-race#counts). All variables use int-type except for the date and state.

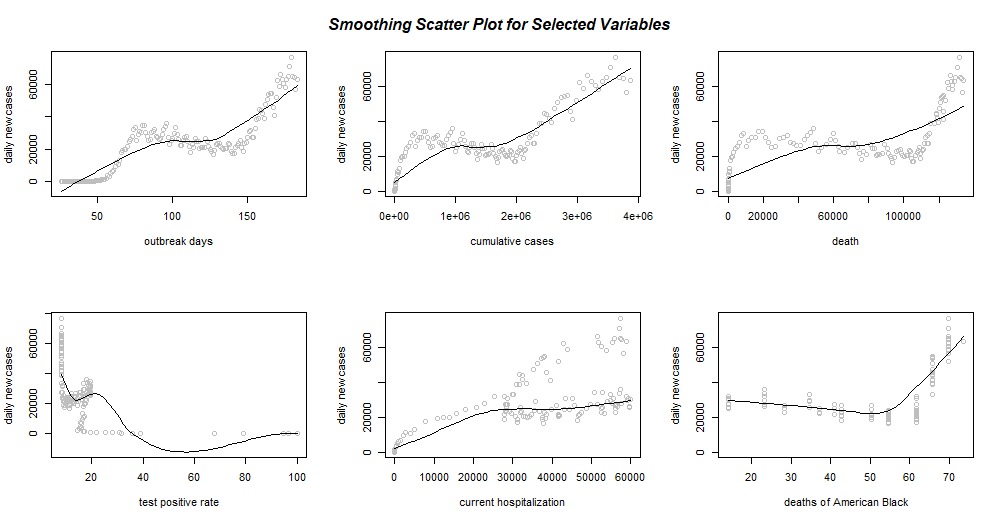
## **2.2 Data Preprocessing**

The first step is feature selection and data aggregation. We pick up five most relevant variables from the social vulnerability index dataset and aggregate raw data by state. Then we aggregate the case data and mobility data by date and merge all the datasets using R. Additionally, we create two numerical variables: outbreak days and test positive rate. The outbreak days are the number of days since the first case was reported in the U.S. The final step is data cleaning. Because the institute reported NA when no case occurred, we simply replace the missing value with 0. Since the records of racial deaths start from April, we apply multiple imputation to fill in the missing data before April. Since the mobility records range from February 15 to July 21, we apply listwise deletion for missing mobility data. Our cleaned dataset contains 22 variables and 158 observations. Please check the appendix for the full list.



## **2.3 Exploratory Data Analysis**

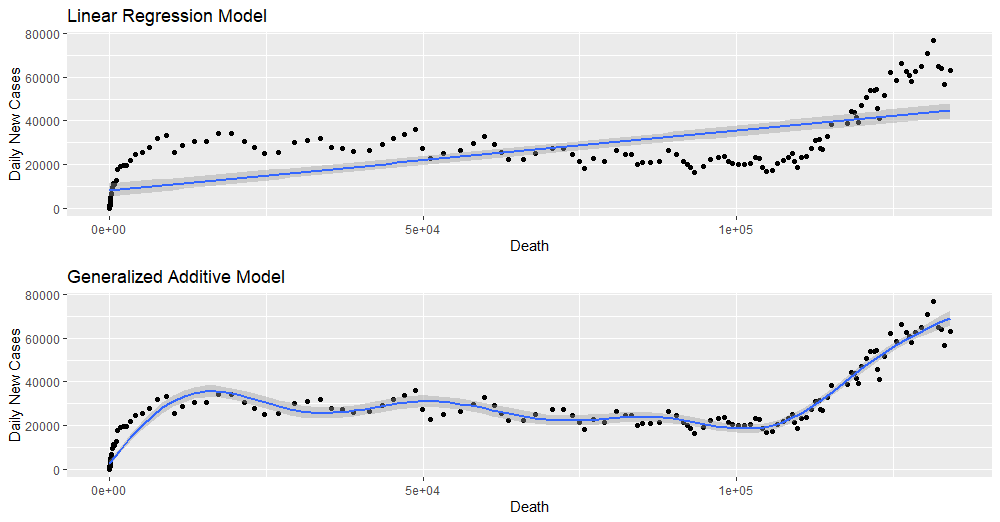
We use a 7-day moving average to visualize the daily number of new COVID-19 cases using ggplot(). The graph suggests the first peak occurs in April. At that moment, the government placed stay-at-home and social distancing orders, which effectively flattens the curve. However, due to the protest activities and reopening of many states in June, we see the second wave of outbreak come back in July. Therefore, the study will analyse what factors cause the surge of coronavirus cases. 

The scatter plots visualize the relationship between daily new cases and selected variables. There is a positive relationship between daily new case and outbreak days, cumulative cases, deaths, and death rate of Black Americans. The test positive rate is negatively related to the number of daily cases.

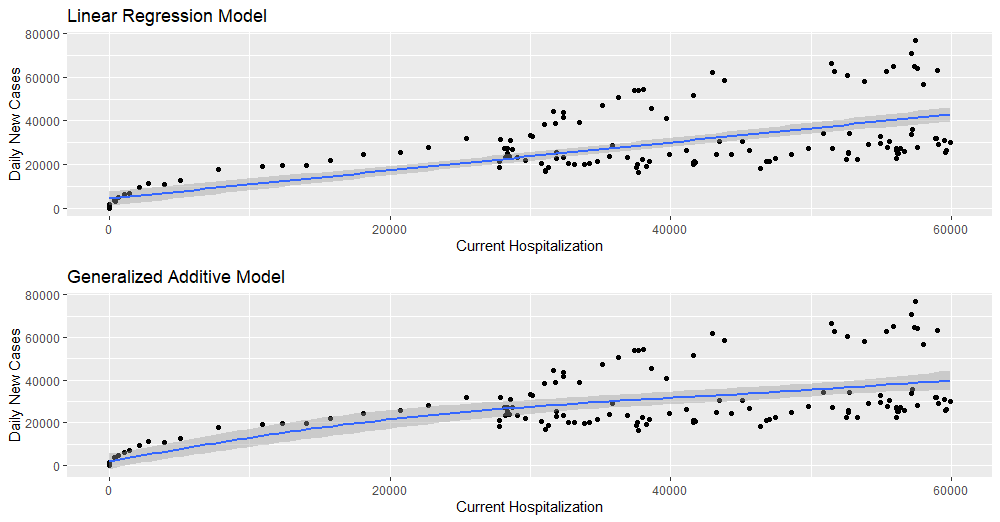
# **3. Feature Engineering and Selection**

## **3.1 Transformation**

The preceding scatter plots along with the smoothing line suggest that some variables present non-linear relationships with daily new cases. In this case, we can create smooth terms for explanatory variables in GAM. We observe the relationship between the response and predictors and choose the smoothing parameters. For example, if adding a smooth term for death better fits to the data, we will create a smooth term for death.



However, if adding a smooth term won’t significantly improve the fit, we will keep the variable as a linear term.



## **3.2 Feature Selection**

By analysing the scatter plots, we take the number of daily new case as our dependent variable, and we have 20 predictors including outbreak days, total cases, total deaths, total tests, test positive rate, current hospitalization, residential rate, grocery rate, park rate, retail rate, transit rate, workplace rate, poverty rate, unemployment rate, income per capita, non-vehicle household rate, uninsured rate, age over 65 rate, deaths for Black Americans, and deaths for White Americans. Additionally, we apply best subsets selection to identify the best regression model.

# **4. Modelling**

We randomly split the dataset into a training set (80% for building the model) and a test set (20% for prediction). By calculating MSE (mean squared error), we could identify the prediction accuracy of all the models.

## **4.1 Multiple Linear Regression Model**

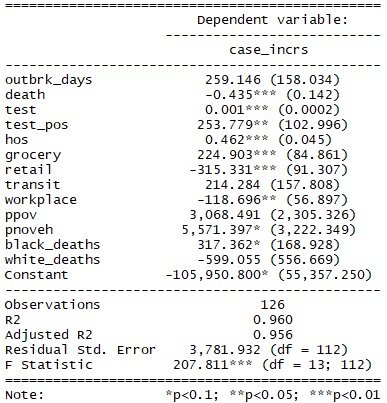
We have 20 predictors in the linear regression model while some of them might not be statistically significant. We conduct the best subsets selection to select the best set of variables for our model.

The following table tells us the size of the best model based on each criterion. The adjusted R-squared evaluates the variance explained by the predictors while the CP and BIC represents the prediction error. Adjusted R2 suggests the best regression model is the one with 13 variables.

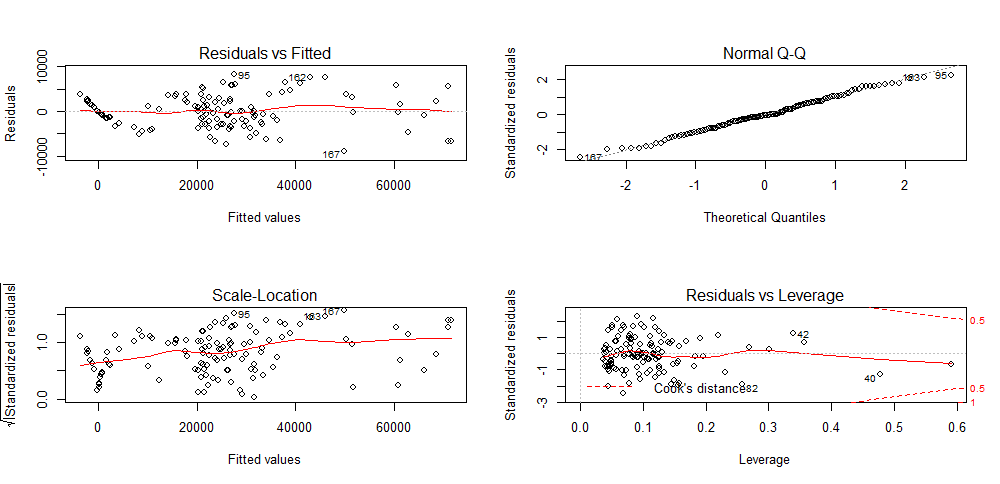
**Table 1 Metrics of Best Subset Selection**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Adjusted R-squared | CP | BIC |
| best size of model | 13 | 9 | 5 |

Here is the formula for linear regression model:



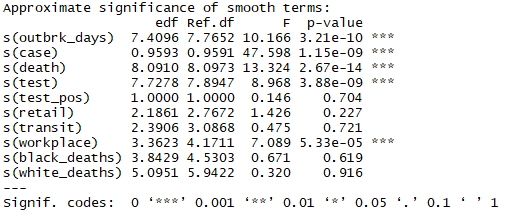
The regression output shows, excluding the outbreak days, transit and death rate of White Americans, all other variables have significant effect on the daily new cases. To be more specific, keeping all other predictors constant, 1% growth in grocery rate will lead to 224 increase in daily new cases on average. The coefficient of hos shows every 1000 increase in current hospitalizations will lead to 462 increase in daily new cases. We also note that mortality tolls of black is strongly related to the case growths, indicating potential racial disparity in the healthy system. The R2 suggests 96% of variance in daily new cases could be explained by predictors. The training MSE is 12,713,787 and the test MSE is 21,133,054. From the diagnostic plots, we can see the residuals are not evenly spread as fitted values increase, indicating there might be non-linear relationship between the response and predictors and we need consider another model.



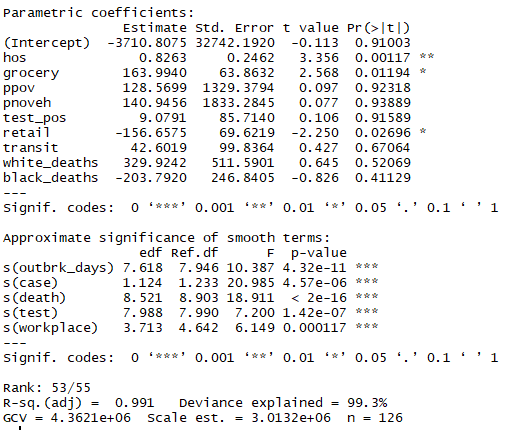
## **4.2 Generalized Additive Models (GAMs)**

We repeat the step mentioned in section 3.1 and create some smooth terms for variables in the previous model, plus case variable, and construct a new generalized additive model.

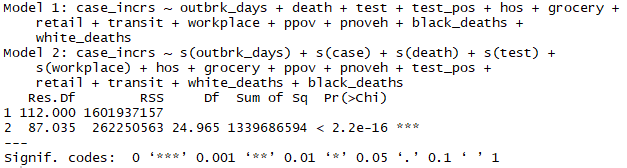
The output below shows the significance of smooth terms. There is no evidence for the significance of test\_pos, retail, transit, black\_deaths and white\_deaths. Therefore, we remove the smooth terms for these variables and add them back to linear terms.



The updated model shows current hospitalization, grocery rate, retail rate, outbreak days, cases, death, test and workplace rate are significantly associated with the number of daily new cases.

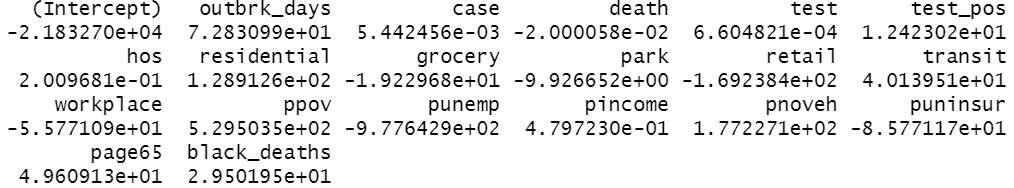


The R-squared is 99.1%, suggesting approximately 99.1% variance in response could be explained by the smoothing predictors. The training MSE is 2,081,354 and the test MSE is 8,314,604, which is better than the linear regression model. We can compare the GAM with the prior linear regression model with anova analysis. The result suggests introducing non-linear terms will improve the model fit.



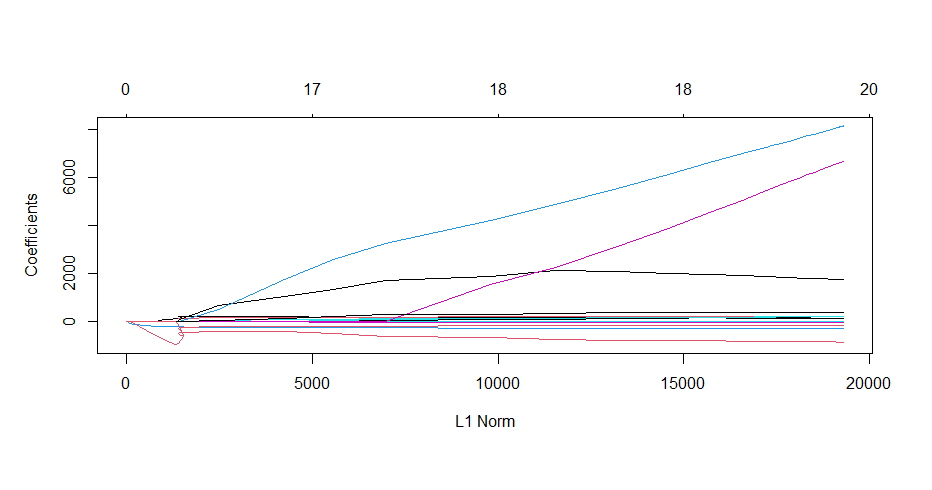
## **4.3 Ridge Regression and The Lasso Models**

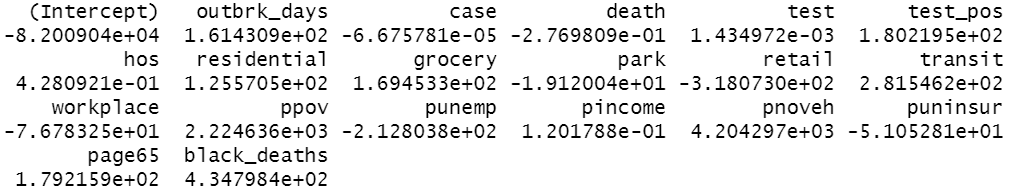
We use the glmnet package in order to perform ridge regression and the lasso. We split the covidbydate dataset into a training set and a test set in order to estimate the test error of ridge regression and the lasso. We fit a ridge regression model on the training set and evaluate its MSE on the test set using λ = 4. Then we get predictions for a test set with the new argument. The test MSE is 21,749,845. However, instead of arbitrarily choosing λ = 4, it would be better to use cross-validation to choose the tuning parameter λ. We do this using the built-in cross-validation function, cv.glmnet(). Therefore, we see that the value of λ that results in the smallest cross-validation error is 1476.059, and the training MSE 23,919,102 with test MSE 20,474,454. The R2 is 94.3%, meaning 94.3% of variance could be explained by predictors. We then refit our ridge regression model on the full data set, using the value of λ chosen by cross-validation, and examine the coefficient estimates.



As expected, none of the 20 coefficients are zero, meaning ridge regression does not perform variable selection.

Now we want to know whether the lasso can yield either a more accurate or a more interpretable model than ridge regression. In order to fit a lasso model, we once again use the glmnet() function; however, this time we use the argument alpha = 1.

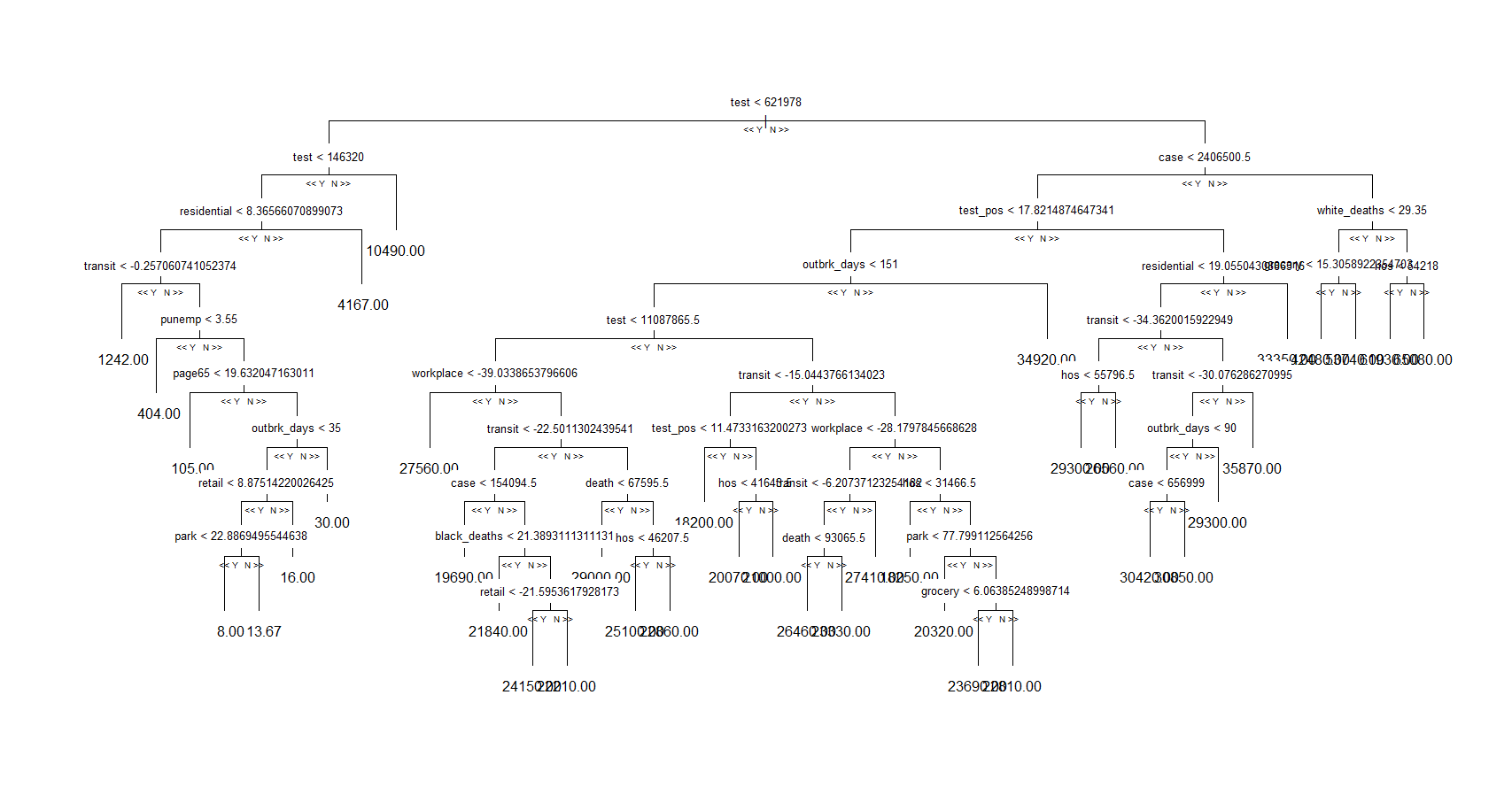


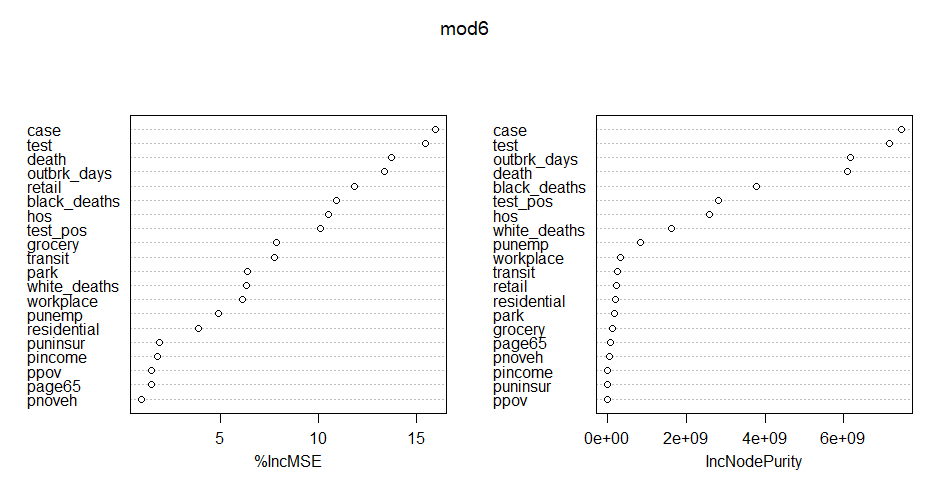
We can see from the coefficient plot above that depending on the choice of tuning parameter, some of the coefficients will be exactly equal to zero. We now perform cross-validation and compute the associated test error. The training MSE of the lasso is 12,694,600 with the test MSE 22,199,203. The R2 is 94.9%, meaning 94.9% of variance could be explained by predictors.

The lasso has a substantial advantage over ridge regression in that the resulting coefficient estimates are sparse. However, as none of the 20 coefficient estimates are zero, the lasso model with λ chosen by cross-validation contains all 20 variables.

## **4.4 Random Forest Model**

Here we apply the random forest model to the covidbydate dataset. By default, randomForest() uses p/3 variables when building a random forest of regression trees, so here we use mtry = 6, and the training MSE is 2,166,339 with the test MSE 11,727,616. The R2 is 97.4%, meaning 97.4% of variance could be explained by predictors. We then use the reprtree() and varImpPlot() functions to view the importance of each variable.





The resulting plots above indicate that across all of the trees considered in the random forest, case and test are by far the two most important variables.

# **5. Conclusion**

## **5.1 Model Selection**

**Table 2 Evaluation Metrics of Models**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Linear regression | GAMs | Ridge regression | The lasso | Random forest |
| Training MSE | 12,713,787 | 2,081,354 | 23,919,102 | 12,694,600 | 2,166,339 |
| Test MSE | 21,133,054 | 8,314,604 | 20,474,454 | 22,199,203 | 11,727,616 |
| R2 | 96% | 98.8% | 94.3% | 94.9% | 97.4% |

We use MSE as our key metric to select the best model. The smaller the MSE the better the prediction accuracy of the model. Consequently, we select the GAM as our final model due to the smallest test and training scores among all models.

## **5.2 Lessons Learned / Policies**

Through analysis from the models, we find tests and cases are strongly related to the daily case increases. Only through that expanding diagnostic and antibody testing will we have the data and information necessary for public health officials to determine when it is safe to resume a semi-normal way of life. Additionally, current hospitalization plays a critical role in the growth of new cases. Overwhelmed hospitals would force patients to wait outside, leaving more people infected and even died. More makeshift hospitals could be built to treat increasing patients. The significance of grocery rate, retail rate and workplace rate indicate the government should carefully monitor if people practice social distancing rules in such areas.

## **5.3 Potential Next Steps**

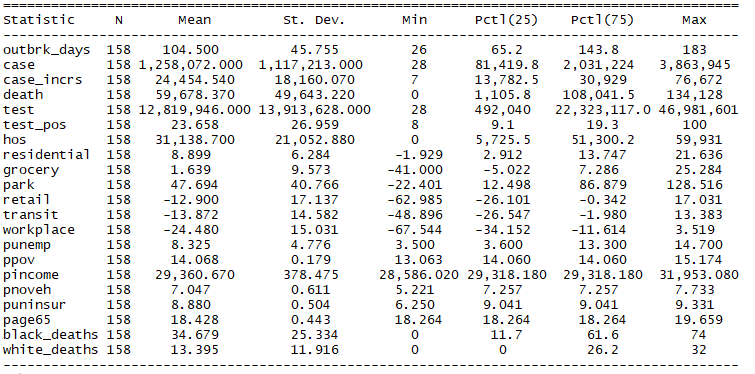
If switching the view to the global coronavirus cases, we will see different countries present different growth curves. In the next step, we could collect data outside the United States and figure out a way to measure the effectiveness of social distancing/lockdown policies across different countries. While we have researched on race and poverty, we’re looking for more factors including sex and age to get additional understanding for determining future actions or policies.

# **Appendix I - Formula and Variable Definition**

Formula:

|  |  |  |  |
| --- | --- | --- | --- |
| Variable | Definition | Type | Original variables |
| Date | Date of the day | Date | Yes |
| case\_incrs | COVID-19 daily new cases | int | Yes |
| outbrk\_days | Outbreak days since the first day of covid-19 occurrence | int | No |
| case | Total number of confirmed cases | int | Yes |
| death | COVID-19 cumulative death number by state in the U.S. | int | Yes |
| test | Total number of viral tests | int | Yes |
| test\_pos | Total number of completed antigen tests that return positive | int | No |
| hos | Individuals who are currently hospitalized with COVID-19 | int | Yes |
| residential | Mobility trends for places of residence. | int | Yes |
| grocery | Mobility trends for places like grocery markets and drug stores. | int | Yes |
| park | Mobility trends for places like local parks, national parks. | int | Yes |
| retail | Mobility trends for places like restaurants, cafes, shopping centers. | int | Yes |
| transit | Mobility trends for places like public transport hubs such as subway, bus, and train stations. | int | Yes |
| workplace | Mobility trends for places of work. | int | Yes |
| ppov | Percentage of persons below poverty | int | Yes |
| punemp | Percentage of unemployment | int | Yes |
| pincome | Per capita income estimate | int | Yes |
| pnoveh | Percentage of households with no vehicle | int | Yes |
| puninsur | Percentage of uninsured persons | int | Yes |
| page65 | Percentage of persons aged 65 and older | int | Yes |
| black\_deaths | Death tolls per 100K Black Americans | int | Yes |
| white\_deaths | Death tolls per 100K Black Americans | int | Yes |

# **Appendix II - Summary Statistics Table**



# **Appendix III - Reference**

1. COVID-19 Forecasts: Deaths. Retrieved from: https://www.cdc.gov/coronavirus/2019-ncov/covid-data/forecasting-us.html
2. Yuan, X., Xu, J., Hussain, S., Wang, H., Gao, N., & Zhang, L. (2020). Trends and Prediction in Daily New Cases and Deaths of COVID-19 in the United States: An Internet Search-Interest Based Model. *Exploratory research and hypothesis in medicine*, *5*(2), 1.

Retrieved from: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7176069/

1. Ou, S., He, X., Ji, W., Chen, W., Sui, L., Gan, Y., ... & Bouchard, J. (2020). Machine learning model to project the impact of COVID-19 on US motor gasoline demand. *Nature Energy*, 1-8.

Retrieved from: https://www.nature.com/articles/s41560-020-0662-1

# **Appendix IV - Code**

library(readxl)

library(ggplot2)

library(tidyverse)

library(dplyr)

library(forecast)

library(glmnet)

library(leaps)

library(caret)

library(stargazer)

library(gridExtra)

library(gam)

library(mgcv)

library(randomForest)

RNGkind(sample.kind = "Rounding")

################Part I: Data Cleaning################

covid <- read.csv("case.csv",header=T)

demographic <- read.csv("Social\_Vulnerability\_Index\_2018\_-\_\_United\_States\_\_county.csv", header = T)

mobility <- read\_excel("mobility\_us.xlsx")

unemp\_race <- read\_excel("unemployment\_racegroup.xlsx")

covid[is.na(covid)] <- 0 # replace NA with 0

covid<-subset(covid, state!="AS" & state!="VI" & state!="MP"& state!="GU" & state!="PR") # we drop the rows because insular area are out of our research scope

# aggregate by state

demo <- demographic %>%

group\_by(ST\_ABBR) %>%

summarise(ppov = mean(EP\_POV), pincome = mean(EP\_PCI), pnoveh = mean(EP\_NOVEH), puninsur = mean(EP\_UNINSUR), page65 = mean(EP\_AGE65))

covid= merge(covid, demo, by.x = "state", by.y = "ST\_ABBR",all.x = TRUE, all.y = FALSE) # merge covid and demographic datasets

# aggregate by date

covidbydate <- covid %>%

group\_by(date) %>%

summarise(case = sum(positive),case\_incrs = sum(positiveIncrease), death = sum(death),

test = sum(totalTestResults),hos = sum(hospitalizedCurrently),ppov = mean(ppov), pincome = mean(pincome),

pnoveh = mean(pnoveh), puninsur = mean(puninsur), page65 = mean(page65))

mobility <- mobility %>%

group\_by(date) %>%

summarise(residential = mean(residential), grocery = mean(grocery),park = mean(park), retail = mean(retail),

transit = mean(transit),workplace = mean(workplace))

covidbydate$date = as.Date(as.character(covidbydate$date),"%Y%m%d")

mobility$date = as.Date(mobility$date)

unemp\_race$date = as.Date(unemp\_race$date)

covidbydate = merge(covidbydate, mobility, by.x = c("date"), by.y = c("date"),all.x = TRUE, all.y = FALSE) # merge covid and mobility datasets

covidbydate = merge(covidbydate, unemp\_race, by.x = c("date"), by.y = c("date"),all.x = TRUE, all.y = FALSE)

# create new variables: outbrk\_days and test\_pos

covidbydate$outbrk\_days = round(as.numeric(difftime(covidbydate$date, "2020-01-20", units = "days")),0) # days since the first day of outbreak

covidbydate$test\_pos = (covidbydate$case/covidbydate$test)\*100

# clean NA's

summary(covidbydate)

covidbydate <- subset(covidbydate, date >= "2020-02-15" & date <= "2020-07-21") # limit the time range from 2/15 to 7/21

covidbydate$park[is.na(covidbydate$park)] <- 0 # replace 2 NAs with 0

# multiple imputation to fill in NA for race group

attach(covidbydate)

lm <- lm(black\_deaths ~ outbrk\_days,covidbydate)

covidbydate$black\_deaths[is.na(covidbydate$black\_deaths)] <- predict(lm, list(covidbydate$outbrk\_days[is.na(covidbydate$black\_deaths)]))

lm <- lm(white\_deaths ~ outbrk\_days+case+death,covidbydate)

covidbydate$white\_deaths[is.na(covidbydate$white\_deaths)] <- predict(lm, list(covidbydate$outbrk\_days[is.na(covidbydate$white\_deaths)]))

covidbydate$black\_deaths[covidbydate$black\_deaths <0] <- 0 # since death won't be negative, replace negative values to 0

covidbydate$white\_deaths[covidbydate$white\_deaths <0] <- 0

covidbydate <- covidbydate[, c(1,21,2:5,22,6,12:18,7:11,19:20)] # reorder the columns

covidbydate = data.frame(covidbydate)

write.csv(covidbydate,file = "finaldata\_coronavirus.csv")

################Part II: Data Visualization################

## summary statistics table

stargazer(covidbydate, type = "text", title="Linear Regression Results", align=TRUE,single.row=TRUE)

## daily new case

avgnewcases <- covidbydate %>%

dplyr::mutate(

new\_conf\_03da = zoo::rollmean(case\_incrs, k = 3, fill = NA),

new\_conf\_07da = zoo::rollmean(case\_incrs, k = 7, fill = NA)) %>%

dplyr::ungroup()

df <- data.frame(grp=covidbydate$date,val=covidbydate$case\_incrs)

df2 <- data.frame(grp=avgnewcases$date,val=avgnewcases$new\_conf\_07da)

names(avgnewcases)

ggplot(df, aes(x=grp, y=val)) + geom\_bar(stat="identity",fill="steelblue", width = 1, colour="white") +

geom\_line(data=df2, aes(x=grp, y=val), colour="red",size = 1)+theme\_minimal()+

labs(title = 'Daily New Covid-19 Cases from Feb to Jul in the United States',

x = 'Month',

y = 'Daily New Cases') + theme(

plot.title = element\_text(size = 14, hjust = 0.5, face = "bold")

)

## scatter plots

attach(covidbydate)

plot(covidbydate[2:14], main = "Relationship for Each Pair of Variables")

## visualize relationship between daily new cases and selected variables

par(mfrow=c(2,3))

scatter.smooth(outbrk\_days, case\_incrs, ylab = "daily new cases", xlab = "outbreak days", col="gray")

scatter.smooth(case, case\_incrs, ylab = "daily new cases",xlab = "cumulative cases",col="gray")

scatter.smooth(death, case\_incrs, ylab = "daily new cases", col="gray")

scatter.smooth(test\_pos, case\_incrs, ylab = "daily new cases",xlab = "test positive rate",col="gray")

scatter.smooth(hos, case\_incrs, ylab = "daily new cases",xlab = "current hospitalization",col="gray")

scatter.smooth(black\_deaths, case\_incrs, ylab = "daily new cases", xlab = "deaths of American Black", col="gray")

mtext("Smoothing Scatter Plot for Selected Variables",side = 3, line = -2.5, font = 4, outer = TRUE, cex = 1)

################Part III: Modeling################

formula = case\_incrs~outbrk\_days+case+death+test+test\_pos+hos+residential+grocery+park+retail+transit+workplace+ppov+punemp+pincome+pnoveh+puninsur+page65+black\_deaths+white\_deaths

## Randomly Split into trainning and test datasets

set.seed(1)

row.number = sample(1:nrow(covidbydate), 0.8 \* nrow(covidbydate))

train = covidbydate[row.number, ]

test = covidbydate[-row.number, ]

covid.train=covidbydate[row.number,"case\_incrs"]

covid.test=covidbydate[-row.number,"case\_incrs"]

## Linear Regression Models

mod1 = regsubsets(formula,train,nvmax =21)

res.sum <- summary(mod1)

data.frame(

Adj.R2 = which.max(res.sum$adjr2),

CP = which.min(res.sum$cp),

BIC = which.min(res.sum$bic)

)

coef(mod1,13)

mod2 = lm(case\_incrs~outbrk\_days+death+test+test\_pos+hos+grocery+retail+transit+workplace+ppov+pnoveh+black\_deaths+white\_deaths,train)

stargazer(mod2, type = "text", title="Linear Regression Results", align=TRUE,single.row=TRUE)

par(mfrow=c(2,2))

plot(mod2)

pred2.train = predict(mod2,newdata=train)

data.frame(

MSE.Train = (RMSE(pred2.train, covid.train)^2),

R2.Train = R2(pred2.train, covid.train)) # 12713787

pred2.test = predict(mod2,newdata=test)

data.frame(

MSE.Test = (RMSE(pred2.test, covid.test)^2),

R2.Test = R2(pred2.test, covid.test)) #21133054

## non-linear models

x = death

y = case\_incrs

a = ggplot(covidbydate, aes(x, y)) + geom\_point() + geom\_smooth(method = "lm", formula = y ~x) + labs(title = "Linear Regression Model", x = 'Death', y = 'Daily New Cases')

b = ggplot(covidbydate, aes(x, y)) + geom\_point() + geom\_smooth(method = "gam", formula = y ~s(x)) + labs(title = "Generalized Additive Model", x = 'Death', y = 'Daily New Cases')

grid.arrange(a,b)

x = hos

y = case\_incrs

a = ggplot(covidbydate, aes(x, y)) + geom\_point() + geom\_smooth(method = "lm", formula = y ~x) + labs(title = "Linear Regression Model", x = 'Current Hospitalization', y = 'Daily New Cases')

b = ggplot(covidbydate, aes(x, y)) + geom\_point() + geom\_smooth(method = "gam", formula = y ~s(x)) + labs(title = "Generalized Additive Model", x = 'Current Hospitalization', y = 'Daily New Cases')

grid.arrange(a,b)

## GAM

## fitting natural cubic spline

mod3 <- gam(case\_incrs ~ s(outbrk\_days)+s(case)+s(death)+s(test)+s(test\_pos) +

s(retail)+s(transit)+s(workplace)+s(black\_deaths)+s(white\_deaths)+hos+grocery+

ppov+pnoveh,data=train)

summary(mod3) # check the significance of smooth term

mod3 <- update(mod3,.~.-s(test\_pos)-s(retail)-s(transit)-s(white\_deaths)-s(black\_deaths)+test\_pos+retail+transit+white\_deaths+black\_deaths)

summary(mod3)

pred3.train = predict(mod3,newdata=train)

data.frame(

MSE.Train = (RMSE(pred3.train, covid.train)^2),

R2.Train = R2(pred3.train, covid.train))# MSE 2081354

pred3.test = predict(mod3,newdata=test)

data.frame(

MSE = (RMSE(pred3.test, covid.test)^2),

R2 = R2(pred3.test, covid.test))

anova(mod2, mod3, test="Chisq") # significance for the second one

## ridge

set.seed(1)

x = model.matrix(formula, covidbydate)[, -1]

y = covidbydate$case\_incrs

train = sample(1:nrow(covidbydate), 0.8 \* nrow(covidbydate))

test = (-train)

y.train = y[train]

y.test = y[test]

mod4 = glmnet(x[train, ], y[train], alpha = 0,

lambda = grid)

pred4 = predict(mod4, s = 4, newx = x[test, ])

mean((pred4 - y.test) ^ 2)

cv.out = cv.glmnet(x[train, ], y[train], alpha = 0)

par(mfrow=c(1,1))

plot(cv.out)

bestlam = cv.out$lambda.min

bestlam

pred4.train = predict(mod4, s = bestlam, newx = x[train, ])

pred4 = predict(mod4, s = bestlam, newx = x[test, ])

mean((pred4.train - y.train) ^ 2)

mean((pred4 - y.test) ^ 2)

out = glmnet(x, y, alpha = 0)

predict(out, type = "coefficients", s = bestlam)[1:20, ]

R2(pred4, covid.test)

## lasso

set.seed (1)

mod5 = glmnet(x[train, ], y[train], alpha = 1, lambda = grid)

plot(mod5)

cv.out = cv.glmnet(x[train, ], y[train], alpha = 1)

plot(cv.out)

bestlam = cv.out$lambda.min

pred5.train = predict(mod5, s = bestlam, newx = x[train, ])

pred5 = predict(mod5, s = bestlam, newx = x[test, ])

mean((pred5.train - y.train) ^ 2)

mean((pred5 - y.test) ^ 2)

out = glmnet(x, y, alpha = 1, lambda = grid)

lasso.coef = predict(out, type = "coefficients", s = bestlam)[1:20, ]

lasso.coef

R2(pred5, covid.test)

## random forest

set.seed (1)

install.packages("devtools")

library(devtools)

devtools::install\_github('skinner927/reprtree')

library(reprtree)

covid.test = covidbydate[-train, "case\_incrs"]

mod6 = randomForest(formula, data = covidbydate, subset = train,

mtry = 6, importance = TRUE)

mod6

plot(mod6)

reprtree:::plot.getTree(mod6) #tree plot

pred6.train = predict(mod6, newdata = covidbydate[train, ])

pred6.test = predict(mod6, newdata = covidbydate[-train, ])

mean((pred6.train - covid.train) ^ 2)

mean((pred6.test - covid.test) ^ 2)

plot(pred6.test, covid.test)

abline(0, 1)

varImpPlot(mod6)

R2(pred6.test, covid.test)