COVID-19 Analysis

Diptendra Nath Bagchi(dbagchi2), Shrilesh Kathe(sdkathe2), Sahil Wadhwa(sahilw2)

15 May, 2020

Introduction

`COVID-19`, popularly known as coronavirus is caused by a virus called SARS-CoV-2. It has caused irreparable damages world-wide to both economy and to public health. It is one of the deadliest pandemics of recent times that emerged in the last 2 decades.

The main reason it caused so much of havoc is because of the long incubation time (~2 weeks) and the contagiousness of it. It transmits mainly through respiratory droplets produced when an infected person coughs or sneezes. These droplets can land in the mouths or noses of people who are nearby or possibly be inhaled into the lungs. Spread is more likely when people are in close contact with one another (within about 6 feet).

United States is the worst affected country in the world with total number of cases at 14 M with more than 90 thousand deaths and counting. The need of this hour is to understand the most vulnerable section of the society and key levers to reduce the spread of COVID-19. It is equally important to learn from the policies that worked and keep improving the quality of healthcare at an increased pace.

In this analysis, we looked at the publicly available county level data to generate insights that could help the spread of the disease. From our study, we have found that the most vulnerable section of the society are the old people especially who are above 65 years old. It also suggests that the densely populated areas with less per capita hospitals and doctors are running a high risk of the spread within the county and also beyond it. Hence, based on our analysis we recommend social distancing and providing good health care in a timely manner to reduce the total number of deaths within counties.

Literature Review

Curating a COVID-19 data repository and forecasting county-level death counts in the United States

The aforementioned talks about prediction techniques to estimate the total number of deaths due to covid-19 in the United States at county level using relevant data collected from various sources. The estimations are used to predict the deaths over a short-term period (e.g. Over the next week), and thus better understand the overall impact of the virus and accordingly implement social distancing policies. A weighted combination of 5 different exponential and linear predictive models were used for predicting the county level deaths.

We took inspiration from this paper to identify important features which the authors have used in their models. For eg, the paper explains how the total reported cases has little to no correlation with the total deaths, since high testing rate doesn't necessarily imply high number of deaths.

Data

The data is taken from a GitHub account with a large corpus of hospital-level and county-level data compiled from a variety of public sources to aid data science eforts to combat COVID-19.

The team reponsible for this data is continually updating and adding to this repository. Currently, it includes data on COVID-19-related cases, deaths, demographics, health resource availability, health risk factors, social vulnerability, and other COVID-19-related information.

For this analysis, we have used a small subset of the data set provided in the abridged version of the data due to various reasons like processing power.

Unsupervised Learning

The main objective of the part is to understand the data in the context of the problem. For this analysis, we have chosen covariates based on a combination of technical and contextual understanding of the research question.

The data come from all the major buckets defined in the GitHub account like geographical, demographics, and health-related risk and resource availability at a county level. Some of the variables seem more important than others like the *density of the county*, the *number of hospitals*, *ICU beds* that are important factors during a pandemic like COVID-19.

Missing Values

There were *missing values for a third of the covariates* but the extent of it varied for different factors but the majority of it was from the health-related factors like *percentage of diabetic population*, *Medicare enrolment percentage*, and so forth. It shows the ineffectiveness of tracking and monitoring in the health care system in the US.

Some of the missing data could be treated but not all due to constraints like non-numeric data and county-level data like latitude and longitude. Hence, the approach taken in the analysis is to divide variables into two categories; the ones that could be imputed and those which cannot be. The variables which were to be imputed were filled by the _median of those values and others remains.

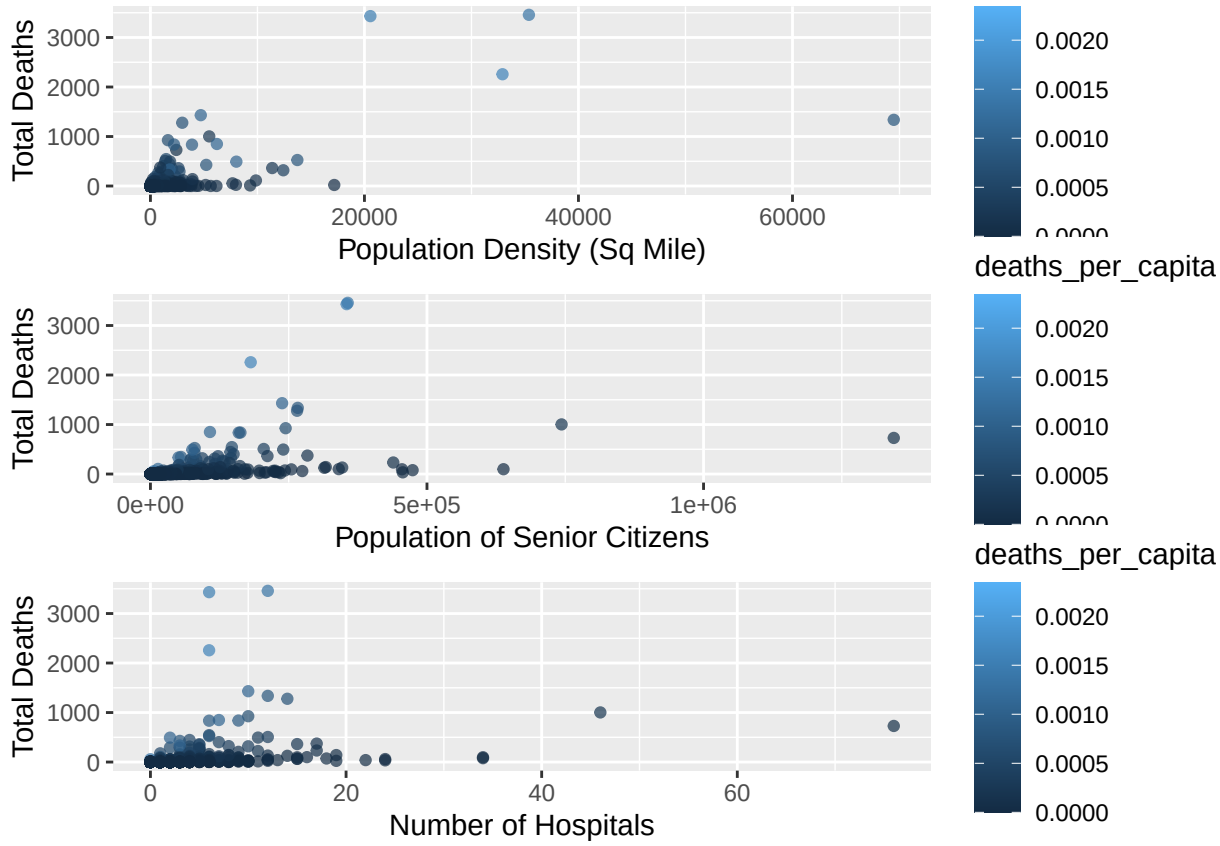
We created per capita variables to remove the size effect on the number of deaths (response variable) at each county as absolute numbers can be less informative while comparing different objects with differing sizes. To see any underlying clusters, a risk variable was also create using both county and state level data. This was defined at a state level with high, medium and low if the number of deaths were more that 593, 165 or others respectively.

After plotting, one clear pattern that emerged density causes more number of deaths both in absolute terms as well as per capita. This ties back to the research reports as one of the measures to stop this pandemic to social distancing. This means the chances of spreading the disease/infection in dense counties are higher compared to others.

Another key insight is number of deaths are higher at a state level. This means some states that are capable of introducing new measures and policies are better at preventing the numbers of deaths than others, which are either late in implementing or does not have resources of doing it.

Correlation Analysis

By doing a correlation plot of the chosen variables, the only variable that stands out is the population density of the county and higher density means more deaths. This is true for both the absolute number and per capita deaths. The limitation of the correlation plot is the linear association of variables that might not be the case in a pandemic like COVID-19. Hence, we also did clustering to generate insights that could lead us to answer some important questions.

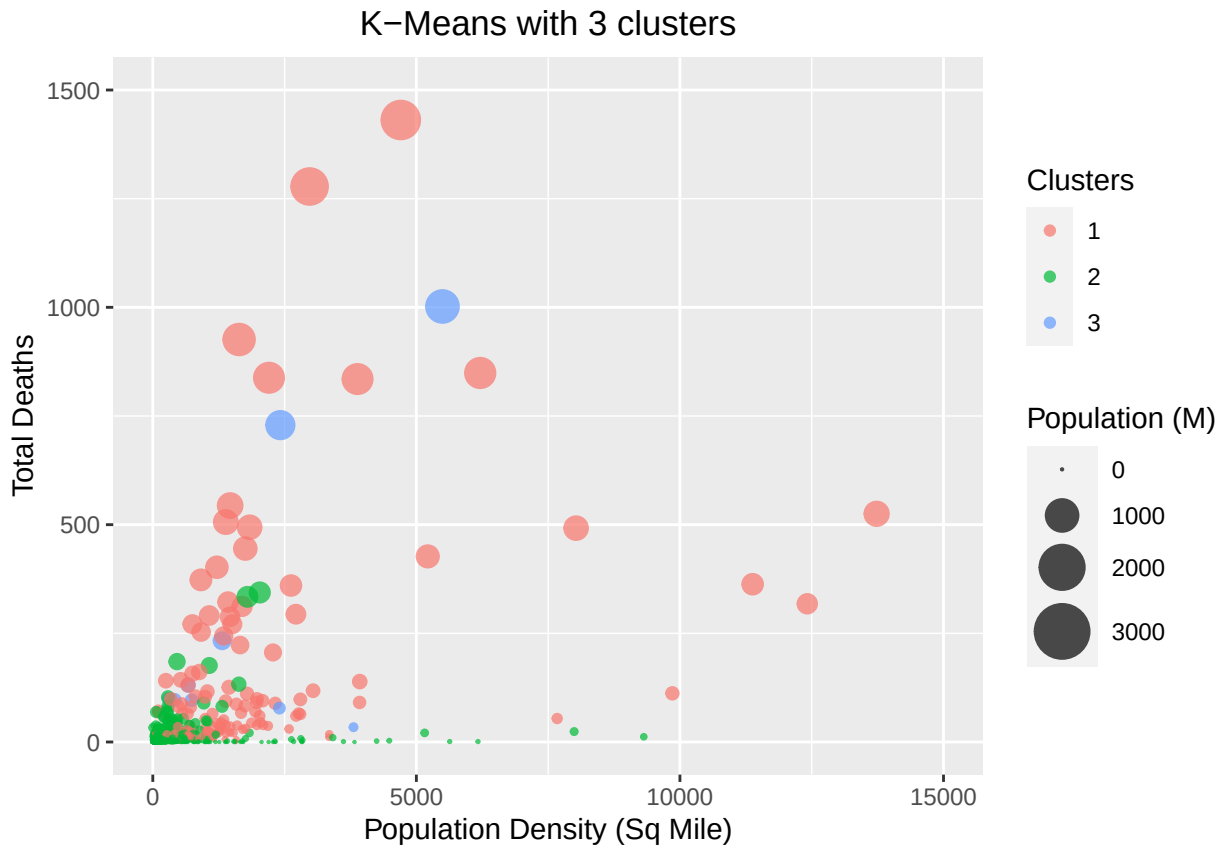


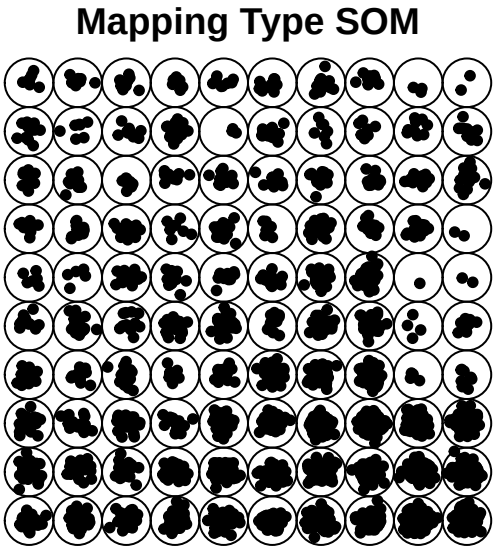
Clustering

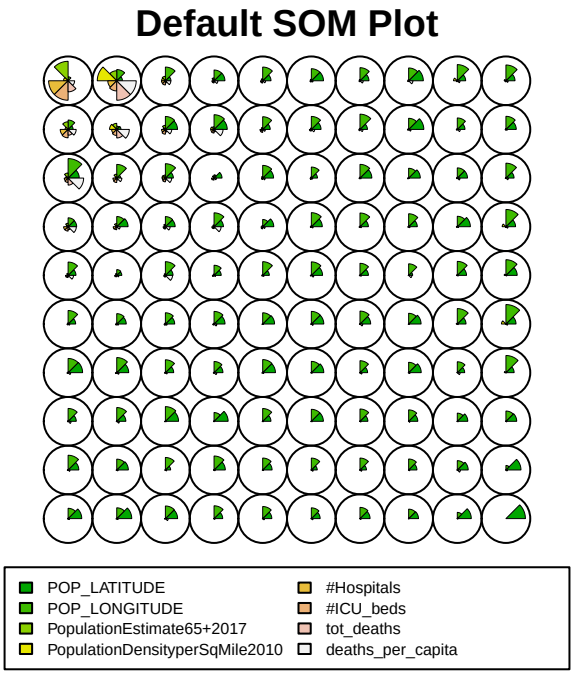
Three clustering techniques were used to find patterns in the data that could be used to cluster the counties.

- `K-Means` clustering was done with 3 clusters (based on the three levels of risk i.e. `Low`, `Medium` & `High`). Some of the key patterns that emerged from the analysis were that the there are clusters where the number of deaths were high due to the population density.
- `Hierarchical` clustering was done but due to the inherent property of the clustering, it did not generate insights that we were trying to find out. Also, one of the limitation was that the more that 90% of the data was attributed to one cluster which is not a desirable property.
- `Self Organizing Maps` were helpful in generating some of the common insighs that we got from the k-means clustering but also helped in understanding the spatial nature of the data. We used a 10*10 matrix because of the speed but we found out that `population latitude`, `number of hospitals` and `icu beds` are important factors that differentiates the counties.

Overall, this analysis gave us a solid understanding of the data and also some of the features that are helpful to distinguish counties but also an in-depth understanding of the research problem at hand.







Supervised Learning

Classification logic:

The aim here is to correctly classify counties with deaths crossing a particular threshold. We create a class variable which looks at the deaths per 100k population which is total_deaths/population_estimate(in 100k).

If this value exceeds 1, that county is classified as 1, else it is classified as 0.

Data Cleaning (Imputing Missing Data):

There are around 29 columns which contain missing data. We classified these columns into 2 types. Numeric columns and Date columns. For Numeric columns missing data was replaced by median. In date columns, we have data containing when strict measures were implemented (eg stay at home, ban on mass-gatherings, etc). Missing values for these columns may signify that strict measures were not implemented yet (as of 22nd April), which means that their situation was stable. So, the missing date values are imputed with 0.

Train test split: We split the available data until 22nd April into train test (80:20) and try to predict the trained model on unseen test data.

Model 1 (XGBoost):

XGBoost is an optimized distributed gradient boosting library designed to be highly efficient, flexible and portable. It implements machine learning algorithms under the Gradient Boosting framework. XGBoost provides a parallel tree boosting (also known as GBDT, GBM) that solve many data science problems in a fast and accurate way.

https://xgboost.readthedocs.io/en/latest/

Hyperparameter Tuning:

We perform cross validation with a grid of parameter values for eta (learning rate), max depth and booster (tree or linear). We select the optimum parameters based on the cross-validation results (using Area under ROC Curve as a metric). After running on the testing data using optimum parameters, we get an accuracy of upto 76%. Below is the confusion matrix:

	Confusion Matrix_XGB	
	0	1
0	321	95
1	62	151

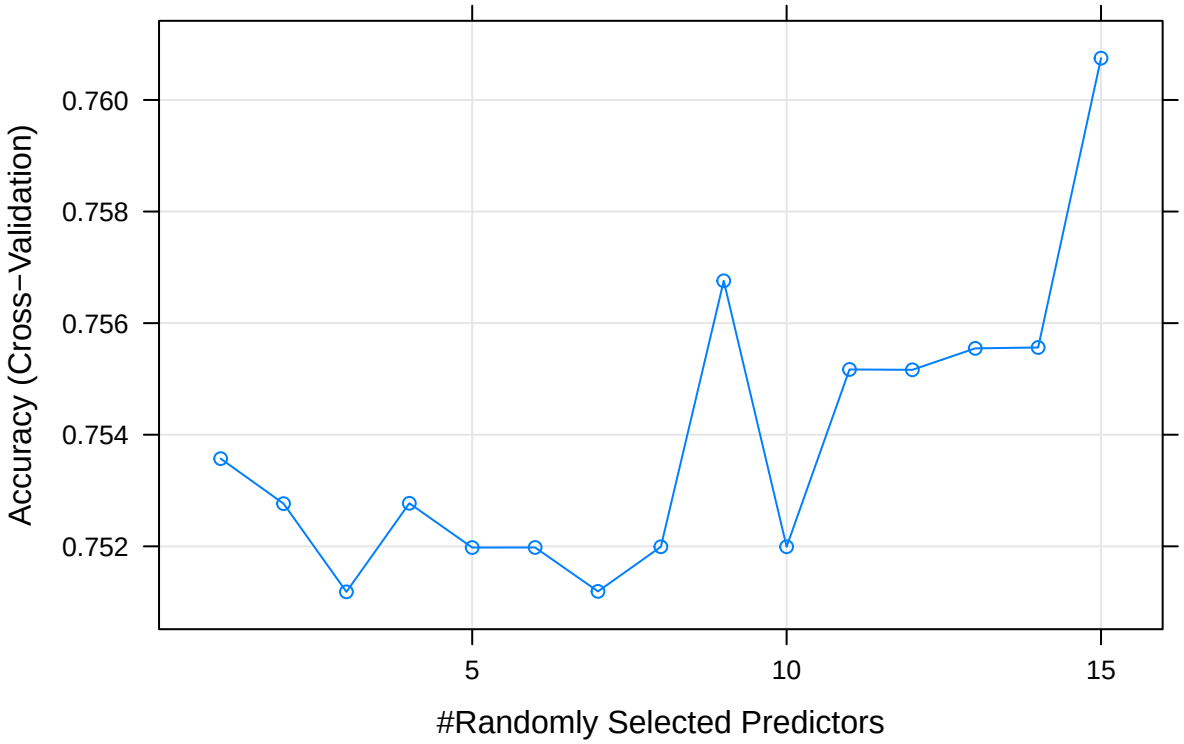
Model 2(Random Forrest):

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for decision trees' habit of overfitting to their training set.

https://en.wikipedia.org/wiki/Random_forest

Tuning

Similar to xgboost, we perform cross-validation on training data to get the optimum value for parameter mtry(Number of variables randomly sampled as candidates at each split). After parameter tuning, we use the model for predicting test data. Below are the results:



	Confusion Matrix_RF	
	0	1
0	323	102
1	60	144

Regression

We perform regression using three methods:-

- Linear Regression
- Gradient Boosting Machines
- Lasso Regression

The data we use for regression is first divided into train and test. The split of train and test is described as follows:-

- Train Data - We use data till 22 April 2020, hence our target variable is the total number of deaths in a county till 22 April 2020
- Test Data - In order to gauge the usefullness of our models, we gather total number of deaths of every county till 29 April 2020

The predictors and response variables are selected based on some assumptions and analysis as described below.

Data Preprocessing

Data is processed in the following way:-

We remove categorical features from the data such as **countyFIPS, STATEFP, COUNTYFYP, CountyName, StateName, State** as we rather use their geographical features such as lattitude, longitude and population features such as **Poulation Density, Total Population** and healthcare features such as **Number of Hospitals, Number of ICU Beds**.

We also use deaths and cases from previous days. We use dates after 2 April 2020 (reference date) and consider their impact on future deaths. We pick every k^{th} day from our reference date till our target date which is 22 April 2020 for training and 29 April 2020 for testing.

Missing Values - In order to fill missing values we impute them with their column medians

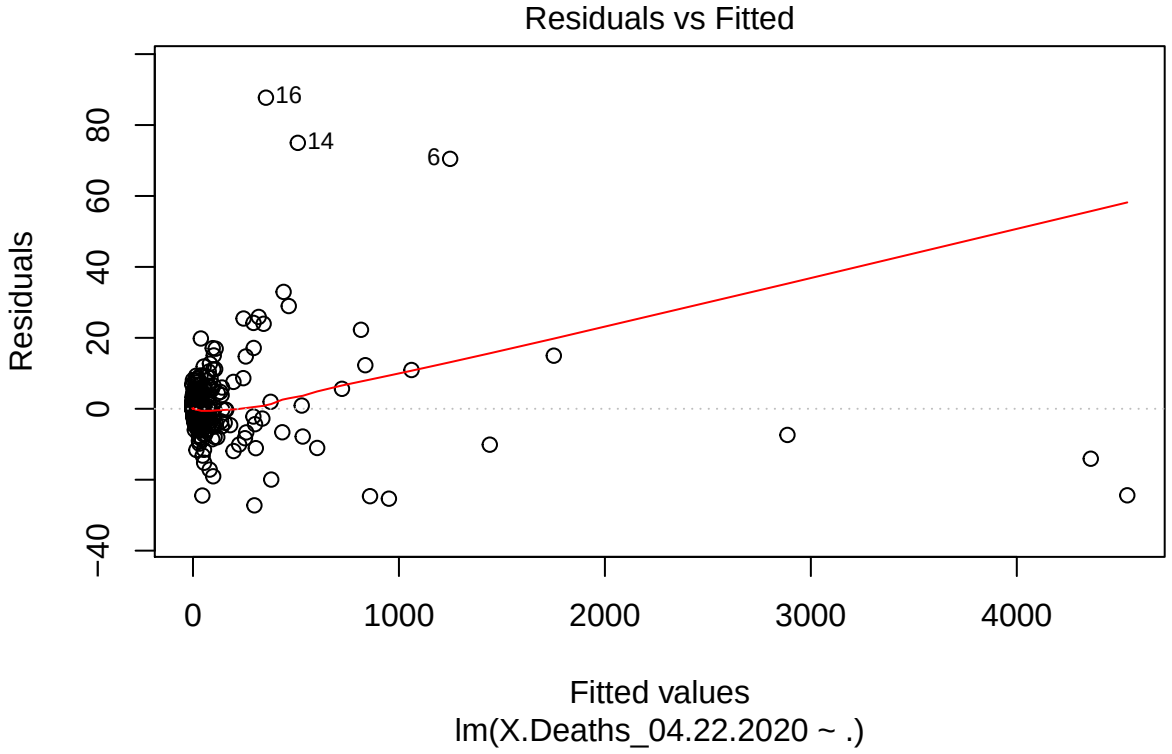
Training.Features	Testing.Features
POP_LATITUDE	POP_LATITUDE
POP_LONGITUDE	POP_LONGITUDE
PopulationDensityperSqMile2010	PopulationDensityperSqMile2010
CensusPopulation2010	CensusPopulation2010
X.Hospitals	X.Hospitals
X.HospParticipatinginNetwork2017	X.HospParticipatinginNetwork2017
X.ICU_beds	X.ICU_beds
X.Deaths_04.19.2020	X.Deaths_04.22.2020
X.Deaths_04.16.2020	X.Deaths_04.19.2020
X.Deaths_04.13.2020	X.Deaths_04.16.2020
X.Deaths_04.10.2020	X.Deaths_04.13.2020
X.Deaths_04.07.2020	X.Deaths_04.10.2020
X.Cases_04.19.2020	X.Cases_04.22.2020
X.Cases_04.16.2020	X.Cases_04.19.2020
X.Cases_04.13.2020	X.Cases_04.16.2020
X.Cases_04.10.2020	X.Cases_04.13.2020
X.Cases_04.07.2020	X.Cases_04.10.2020
X.Deaths_04.22.2020	**X.Deaths_04.29.2020**

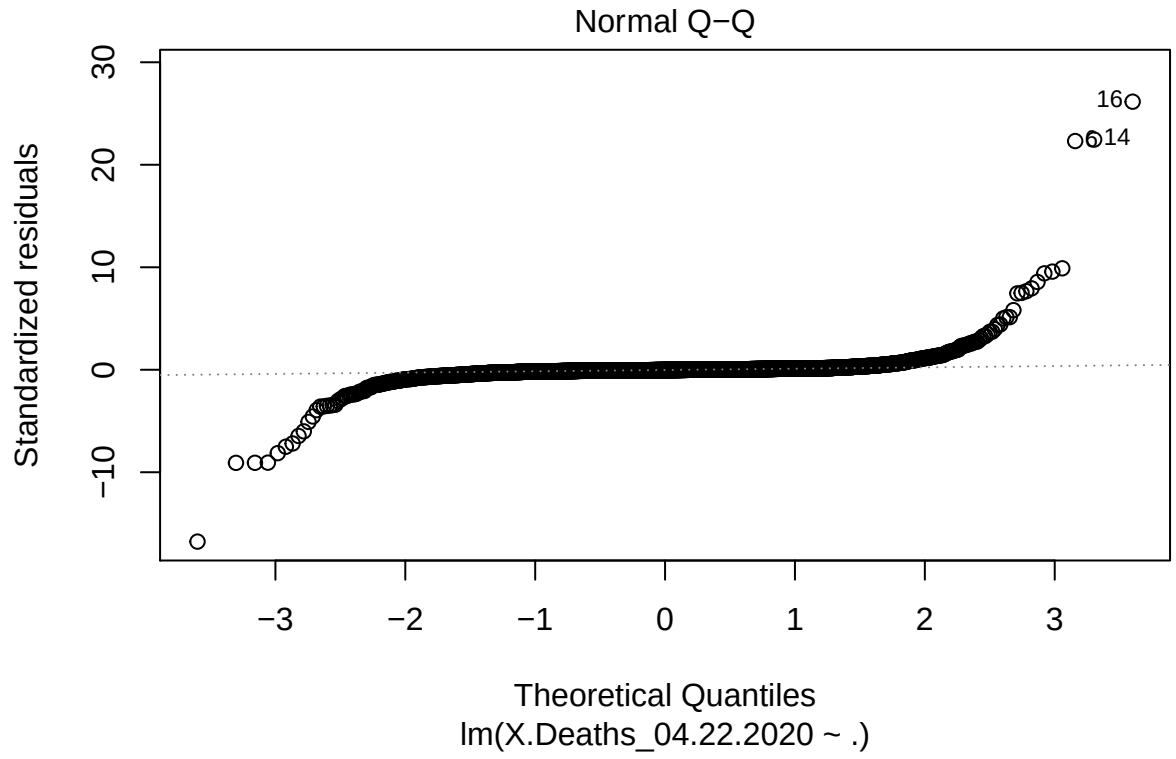
The last rows of each column define the target variable for that column.

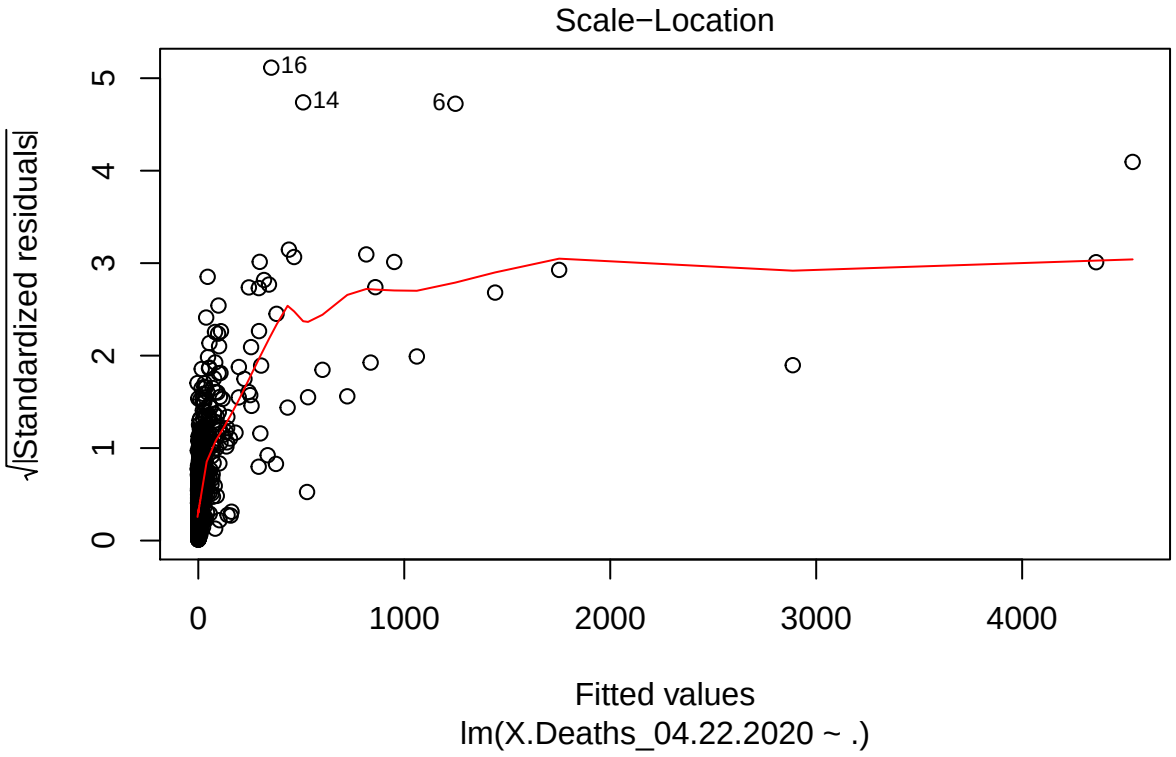
Modelling

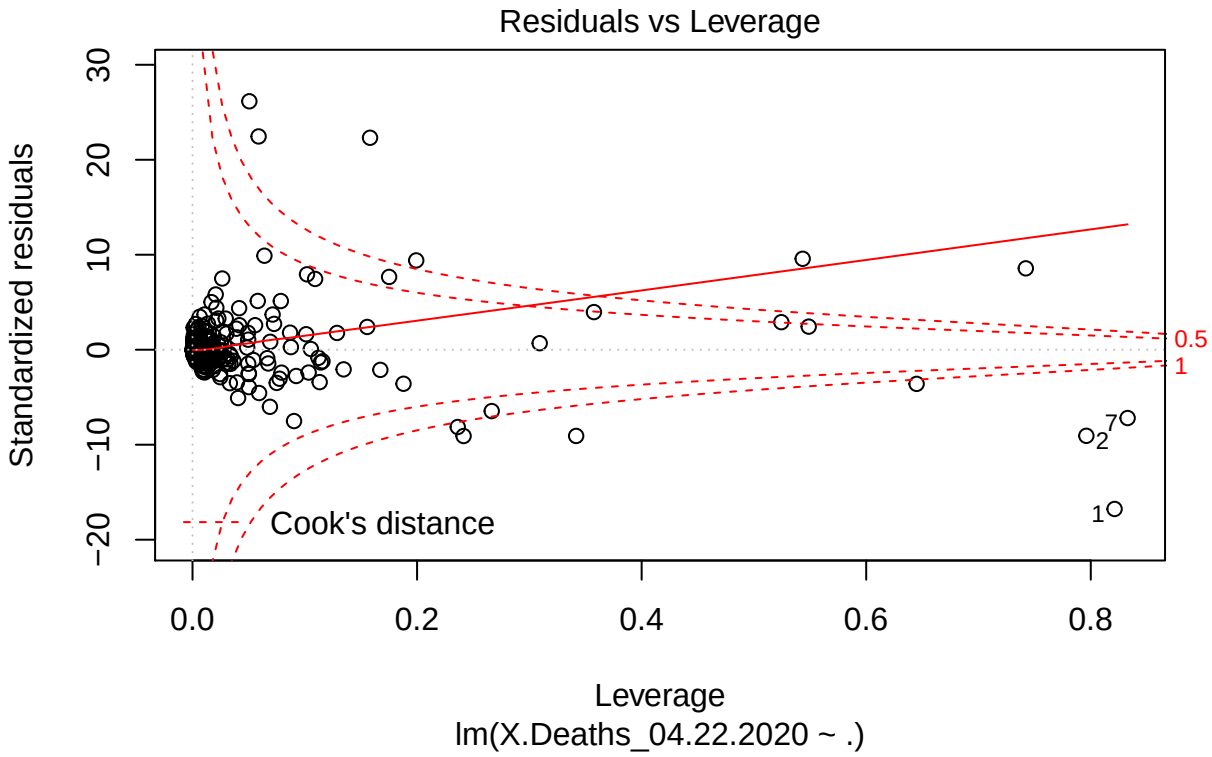
```
##
## Call:
## lm(formula = X.Deaths_04.22.2020 ~ ., data = train_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -27.230  -0.379  -0.097   0.214  87.709
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      14.12264    0.06140 230.009  < 2e-16 ***
## POP_LATITUDE                      0.20859    0.06548   3.185 0.001459 **
## POP_LONGITUDE                     0.28128    0.06661   4.223 2.48e-05 ***
## PopulationDensityperSqMile2010   -0.74950    0.11039  -6.789 1.34e-11 ***
## CensusPopulation2010              0.77945    0.21342   3.652 0.000264 ***
## X.Hospitals                      -0.16495    0.19526  -0.845 0.398302
## X.HospParticipatinginNetwork2017 -0.01556    0.11261  -0.138 0.890136
## X.ICU_beds                        1.05010    0.18905   5.555 3.02e-08 ***
## X.Deaths_04.19.2020             151.45166    2.70635  55.962  < 2e-16 ***
```

```
## X.Deaths_04.16.2020              -10.33736    2.85225  -3.624 0.000294 ***
## X.Deaths_04.13.2020                4.59292    2.05900   2.231 0.025775 *
## X.Deaths_04.10.2020               -3.19551    1.62953  -1.961 0.049968 *
## X.Deaths_04.07.2020               -4.04604    0.99223  -4.078 4.66e-05 ***
## X.Cases_04.19.2020               -10.90323    2.89900  -3.761 0.000172 ***
## X.Cases_04.16.2020                35.89397    4.80325   7.473 1.01e-13 ***
## X.Cases_04.13.2020                24.25940    4.39081   5.525 3.56e-08 ***
## X.Cases_04.10.2020               -58.14447    4.88912 -11.893  < 2e-16 ***
## X.Cases_04.07.2020                11.64569    2.32389   5.011 5.71e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.442 on 3124 degrees of freedom
## Multiple R-squared:  0.9994, Adjusted R-squared:  0.9994
## F-statistic: 3.123e+05 on 17 and 3124 DF,  p-value: < 2.2e-16
```




Normal Q−Q
Standardized residuals
30
20
10
0
−10
16
14
6
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(X.Deaths_04.22.2020 ~ .)


Scale−Location
√|Standardized residuals|
5
4
3
2
1
0
16
14
6
0
1000
2000
3000
4000
Fitted values
lm(X.Deaths_04.22.2020 ~ .)

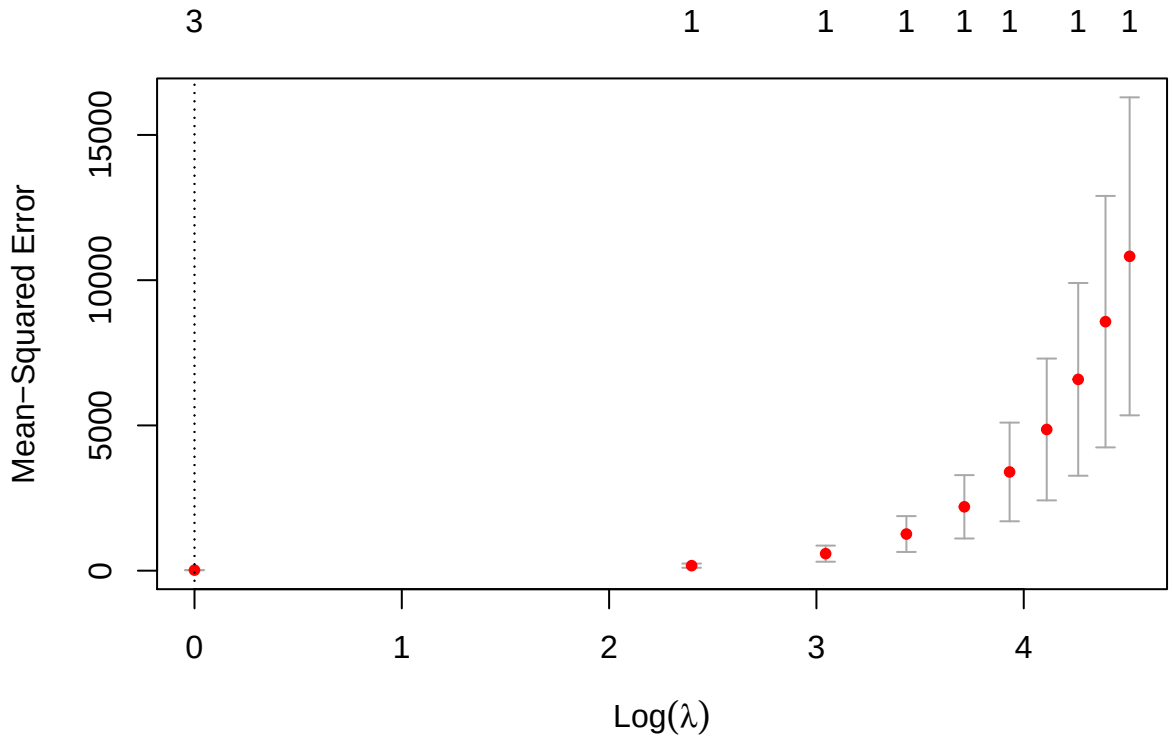


Observation of the above plots of linear regression:-

- Residuals vs Fitted Values - A pattern could be identified from this plot. Some points are clustered heavily whereas some are really far off.
- Norma QQ Plot - The plot signifies that errors in the data are not normally distributed fully
- Scale Location - Points are clustered around the red line when standardized residuals and fitted values are small. However no clear cut pattern could be observed which tells that errors are non homoscedastic
- Residuals vs Leverage - There are some influential points such as 1, 2, 7 but we don't specifically remove them

```
## Using 1850 trees...
```

We tried multiple configuration for **shrinkage**, **depth** and **n_trees** and found 0.1, 5 and 6000 to perform best. We also tried multiple family distributions such as **poisson**, **bernoulli** but their performance was significantly bad than **gaussian**



We perform LASSO regression with 20 folds CV. The purpose of this exercise was to actually get rid of the parameters which don't play a role a significant role in the analysis. Non zero coefficients of the best CV model are as follows

Non.Zero.Coefficients.of.Lasso.Model
Intercept
CensusPopulation2010
X.Deaths_04.19.2020
X.Cases_04.19.2020

The model disregards previous happened deaths and cases and other features from the analysis. This is in line with the modelling technique that has been used by Prof. Yu and her group. They also consider only most recent deaths to model future deaths. The performance of Lasso is worse than using OLS model.

Remark - We tried various configuration for lambda and got lambda as 1 to perform the best.

RMSE
31.02750
31.63765
33.78736

Conclusion:

- Linear model performs the best amongst the other two models
- From all the three models, it is quite evident that recent deaths in counties are most useful in predicting the future deaths. Counties in certain states follow similar pattern and hence the next logical step should be models for state level.
- Some features such as **population density**, **number of ICU beds**, **number of hospitals** etc make logical sense but in analysis they don't show any significant changes in regression analysis.

- Tree based model (GBM in our case) doesn't perform well than the Linear Model as the data is approximately normally-distributed and in cases like these OLS estimator tends to perform better. Had there been more significant features and data, we could anticipate a different scenario

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

For classification task, after clearly analysing the data and using properly tuned models, we were able to generate classifiers which can correcly classify susceptible counties with about 76% accuracy.
