INFO 634, Data Mining, Winter 2021

Project title:

Analyzing the conditions contributing to Covid-19 Deaths

Group members:

Akhila Singanal

Alicia Brandemarte

Manoj Venkatachalaiah

Srilakshmi Rao

1. Introduction

On 11 March 2020, the World Health Organization (WHO) characterized COVID-19, which is caused by severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) as a pandemic, after 118,000 cases and 4,291 deaths were reported in 114 countries. The Coronavirus disease (COVID-19) has rapidly affected mortality worldwide. There is unprecedented urgency to understand who is most at risk of severe outcomes, and this requires new approaches for the timely analysis of large datasets.

Age and gender are well-established risk factors for severe COVID-19 outcomes: over 90% of the COVID-19-related deaths in the UK have been in people over 60, and 60% in men. Various pre-existing conditions have also been associated with increased risk. For example, the Chinese Center for Disease Control and Prevention reported in a study of 44,672 individuals (1,023 deaths) that cardiovascular disease, hypertension, diabetes, respiratory disease and cancers were associated with an increased risk of death; however, correction for relationships with age was not possible. A UK cross-sectional survey of 16,749 patients who were hospitalized with COVID-19 showed that the risk of death was higher for patients with cardiac, pulmonary and kidney disease, as well as cancer, dementia and obesity .

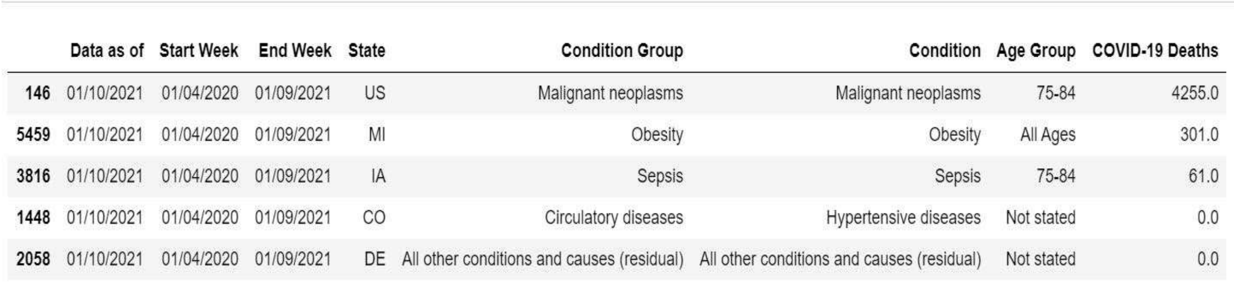
The title of our project is "**Analyzing the Conditions Contributing to Covid-19 Deaths**". Our project team objective is to find patterns in Covid-19 data released by the CDC, which can be found here, along with an additional location based public health dataset, which can be found here and a State-wise mask mandate dataset that was found on Kaggle. We want to merge both datasets to identify the patterns that are interesting and beneficial to the public health community, and examine the key conditions that are causing Covid-19 deaths.

Through knowledge discovery we can find patterns and potentially predict the number of Covid-19 related deaths by using features such as medical condition, age group, state poverty rate, statewide Covid-19 precautions, etc.

2. Data Collection

2.1 CDC Covid Data

Sample(raw data):



	Data as of	Start Week	End Week	State	Condition Group	Condition	Age Group	COVID-19 Deaths
146	01/10/2021	01/04/2020	01/09/2021	US	Malignant neoplasms	Malignant neoplasms	75-84	4255.0
5459	01/10/2021	01/04/2020	01/09/2021	MI	Obesity	Obesity	All Ages	301.0
3816	01/10/2021	01/04/2020	01/09/2021	IA	Sepsis	Sepsis	75-84	61.0
1448	01/10/2021	01/04/2020	01/09/2021	CO	Circulatory diseases	Hypertensive diseases	Not stated	0.0
2058	01/10/2021	01/04/2020	01/09/2021	DE	All other conditions and causes (residual)	All other conditions and causes (residual)	Not stated	0.0

The above data set titled “Conditions contributing to Covid-19” was acquired from the CDC website. The data set contains the number of Covid-19 deaths for each state for the entirety of 2020, which is further classified based on age groups and medical conditions. It has 12,420 instances and 11 columns.

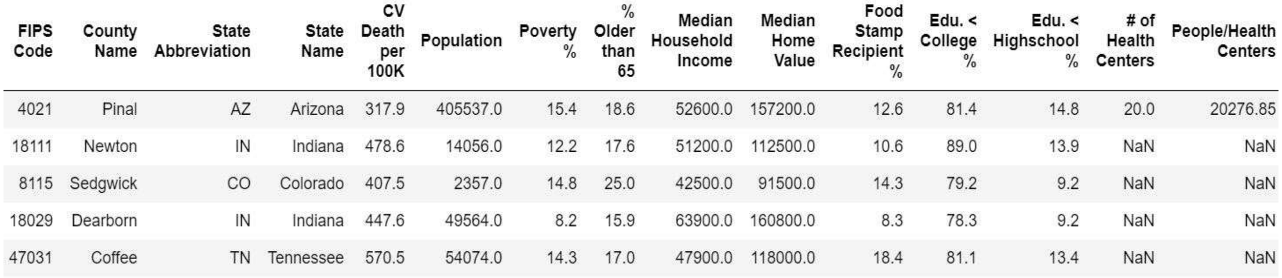
Attribute descriptions:

- ‘Data as of’ : Date the data set was released
- ‘Start Week’: Start date of the time-frame the data has been collected for
- ‘End Week’: End date of the time-frame the data has been collected for
- ‘State’: The states of U.S.A
- ‘Condition Group’: The group a medical condition belongs to
- ‘Condition’: Pre-existing medical conditions of individuals
- ‘Age Group’: Age groups the individuals belonged to

- ‘Covid-19 Deaths’: number of deaths for a state, condition and age group

2.2 Public Health Data

Sample(raw data):



FIPS Code	County Name	State Abbreviation	State Name	CV Death per 100K	Population	Poverty %	% Older than 65	Median Household Income	Median Home Value	Food Stamp Recipient %	Edu. < College %	Edu. < Highschool %	# of Health Centers	People/Health Centers
4021	Pinal	AZ	Arizona	317.9	405537.0	15.4	18.6	52600.0	157200.0	12.6	81.4	14.8	20.0	20276.85
18111	Newton	IN	Indiana	478.6	14056.0	12.2	17.6	51200.0	112500.0	10.6	89.0	13.9	NaN	NaN
8115	Sedgwick	CO	Colorado	407.5	2357.0	14.8	25.0	42500.0	91500.0	14.3	79.2	9.2	NaN	NaN
18029	Dearborn	IN	Indiana	447.6	49564.0	8.2	15.9	63900.0	160800.0	8.3	78.3	9.2	NaN	NaN
47031	Coffee	TN	Tennessee	570.5	54074.0	14.3	17.0	47900.0	118000.0	18.4	81.1	13.4	NaN	NaN

The above dataset is a County-wise public health data set assembled using data related to cardiovascular diseases on CDC and HRSA.

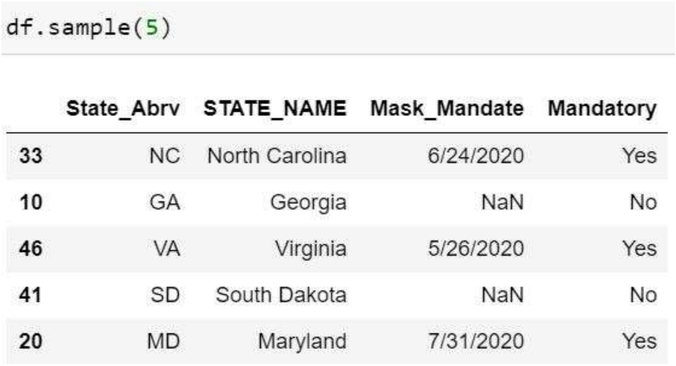
Attribute descriptions:

- ‘FIPS Code’: Federal code of a county of the United States
- ‘County Name’: Name of the county
- ‘State Abbreviation’: State code of the state a county belongs to
- ‘State Name’: Name of the state a county belongs to
- ‘CV Death per 100k’: Number of cardiovascular deaths per 100,000 people
- ‘Population’: Population of the county
- ‘Poverty %’: percentage of population in poverty
- ‘% Older than 65’: percentage of population over the age of 65
- ‘Median household income’: Middle household income among total population in a county
- ‘Median Home Value’: Middle home retail value among total population in a county
- ‘Food Stamp Recipient%’: percentage of population that receives Food Stamp benefits

- 'Edu. < College %’: percentage of population that has education less than College
- 'Edu. < Highschool %': percentage of population that has education less than High School
- '# of Health Centers’: number of health centers in the county
- 'People/Health Centers': number of people per health centers in the county

2.3 Mask Mandate Data set

Sample(raw data):



```
df.sample(5)
```

	State_Abrv	STATE_NAME	Mask_Mandate	Mandatory
33	NC	North Carolina	6/24/2020	Yes
10	GA	Georgia	NaN	No
46	VA	Virginia	5/26/2020	Yes
41	SD	South Dakota	NaN	No
20	MD	Maryland	7/31/2020	Yes

Attribute descriptions:

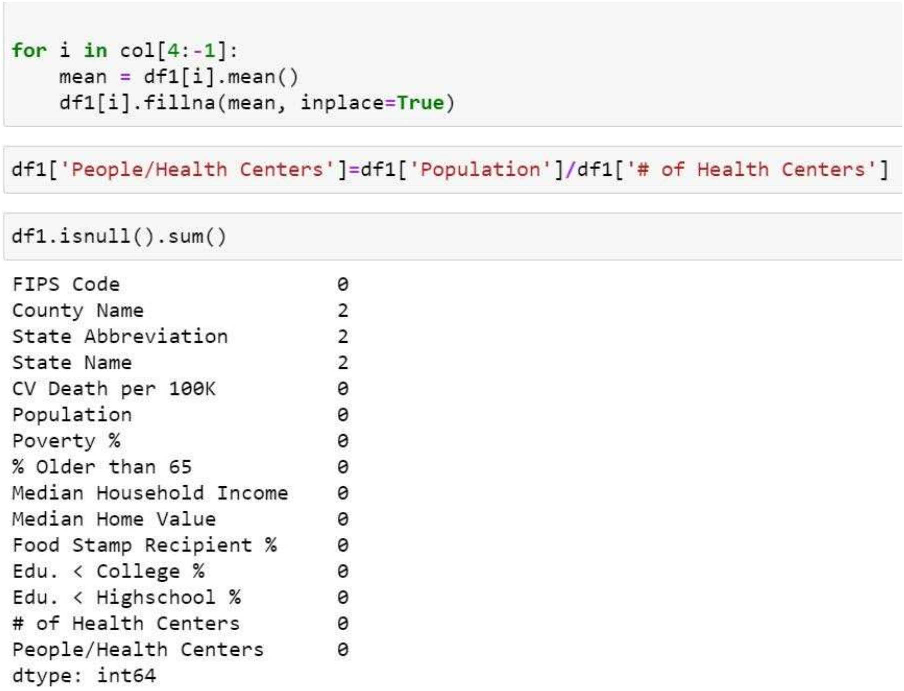
- **‘**State_Abrv’: State code of a state
- ‘STATE_ NAME’: name of the state
- ‘Mask_Mandate’: start date of the mask mandate in a state
- ‘Mandatory’: Whether it was mandatory to wear masks in a state

2.4 Pre-processing: Public Health Data set

2.4.1 Dealing with missing values in the public health data set

The columns ['CV Death per 100K', 'Population', 'Poverty %', '% Older than 65', 'Median Household Income', 'Median Home Value', 'Food Stamp Recipient %', 'Edu. < College %', 'Edu. < Highschool', '# of Health Centers'] had null values for many instances, we decided to fill these values with the mean of corresponding columns. The '# of Health Centers' null values are calculated using the logic:

‘People/Health centers’' = ‘Population’/ '# of Health Centers’



```
for i in col[4:-1]:
    mean = df1[i].mean()
    df1[i].fillna(mean, inplace=True)
```

```
df1['People/Health Centers']=df1['Population']/df1['# of Health Centers']
```

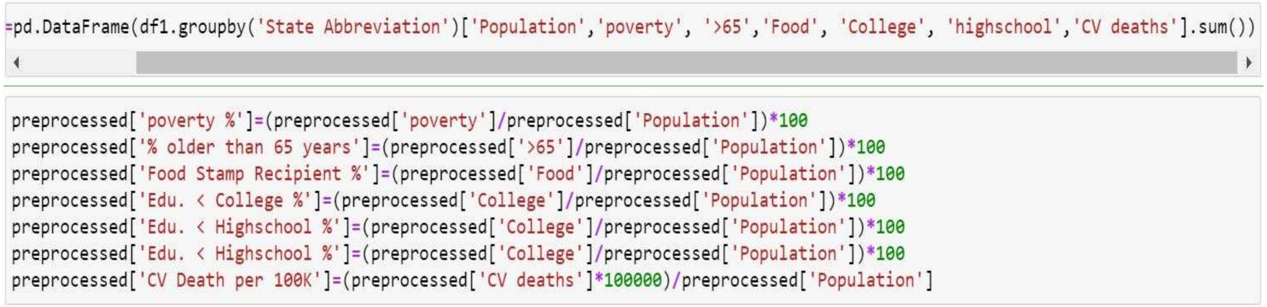
```
df1.isnull().sum()
```

```
FIPS Code                 0
County Name               2
State Abbreviation        2
State Name                2
CV Death per 100K         0
Population                0
Poverty %                 0
% Older than 65           0
Median Household Income   0
Median Home Value         0
Food Stamp Recipient %    0
Edu. < College %          0
Edu. < Highschool %       0
# of Health Centers       0
People/Health Centers     0
dtype: int64
```

There are 2 instances that have null values for categorical features in the resulting dataframe, these instances will be dropped.

2.4.2 Feature Engineering

All numerical features in the public health data set are calculated for each state by grouping the counties with the same state code. Percentages are first converted to actual values and then grouped based on State codes. These values are then converted to percentages.



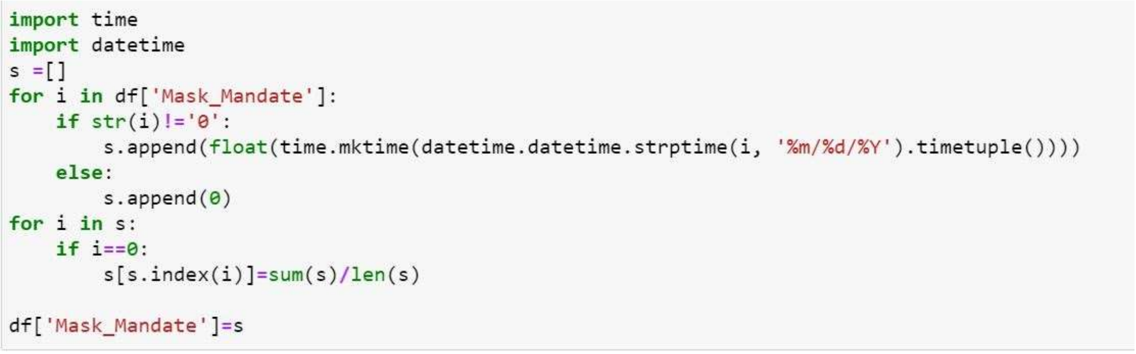
```
=pd.DataFrame(df1.groupby('State Abbreviation')['Population','poverty', '>65','Food', 'College', 'highschool','CV deaths'].sum())
```

```
preprocessed['poverty %']=(preprocessed['poverty']/preprocessed['Population'])*100
preprocessed['% older than 65 years']=(preprocessed['>65']/preprocessed['Population'])*100
preprocessed['Food Stamp Recipient %']=(preprocessed['Food']/preprocessed['Population'])*100
preprocessed['Edu. < College %']=(preprocessed['College']/preprocessed['Population'])*100
preprocessed['Edu. < Highschool %']=(preprocessed['College']/preprocessed['Population'])*100
preprocessed['Edu. < Highschool %']=(preprocessed['College']/preprocessed['Population'])*100
preprocessed['CV Death per 100K']=(preprocessed['CV deaths']*100000)/preprocessed['Population']
```

This preprocessing step was carried out because we needed a common column on which the public health and CDC data sets could be merged.

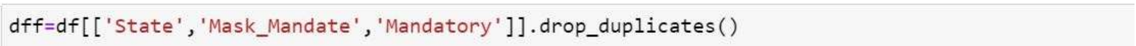
2.5 Preprocessing: Mask Mandate Data set:

2.5.1 Converting date strings to date stamps and mean-imputing the column



```
import time
import datetime
s =[]
for i in df['Mask_Mandate']:
    if str(i)!='0':
        s.append(float(time.mktime(datetime.datetime.strptime(i, '%m/%d/%Y').timetuple())))
    else:
        s.append(0)
for i in s:
    if i==0:
        s[s.index(i)]=sum(s)/len(s)

df['Mask_Mandate']=s
```



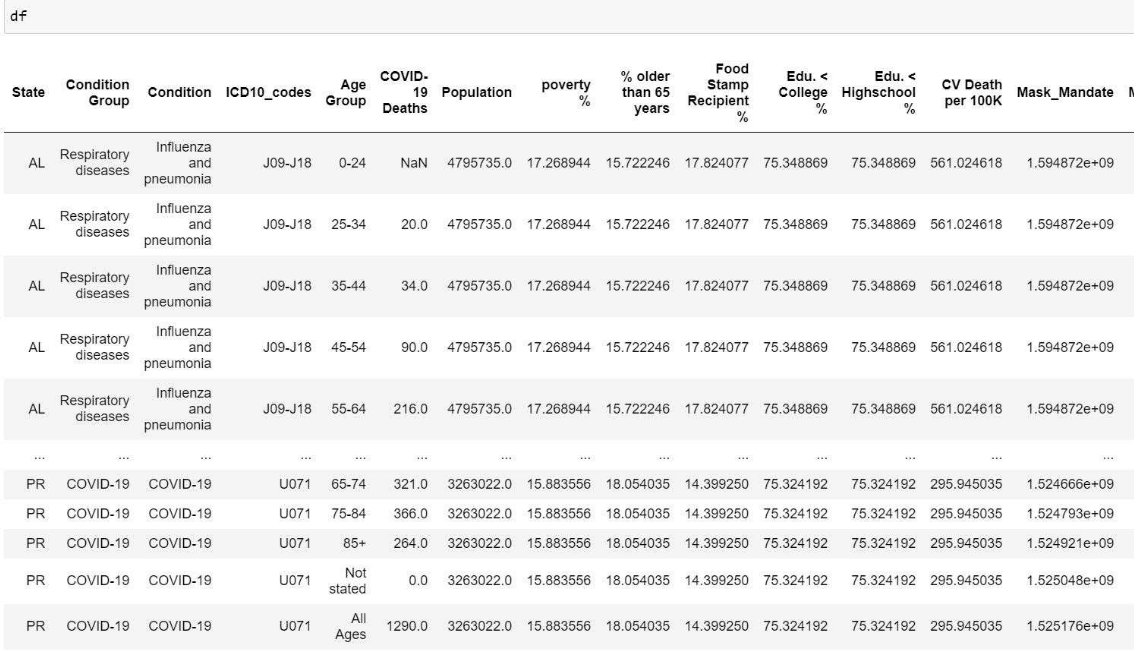
```
dff=df[['State','Mask_Mandate','Mandatory']].drop_duplicates()
```

The ‘Mask_Mandate’ column is of the data type ‘string’, converting them to date stamps will help fill missing values with the mean of the column.

2.6 Merging all three data sets

All the data sets are merged on the ‘State’ .

Final dataset:

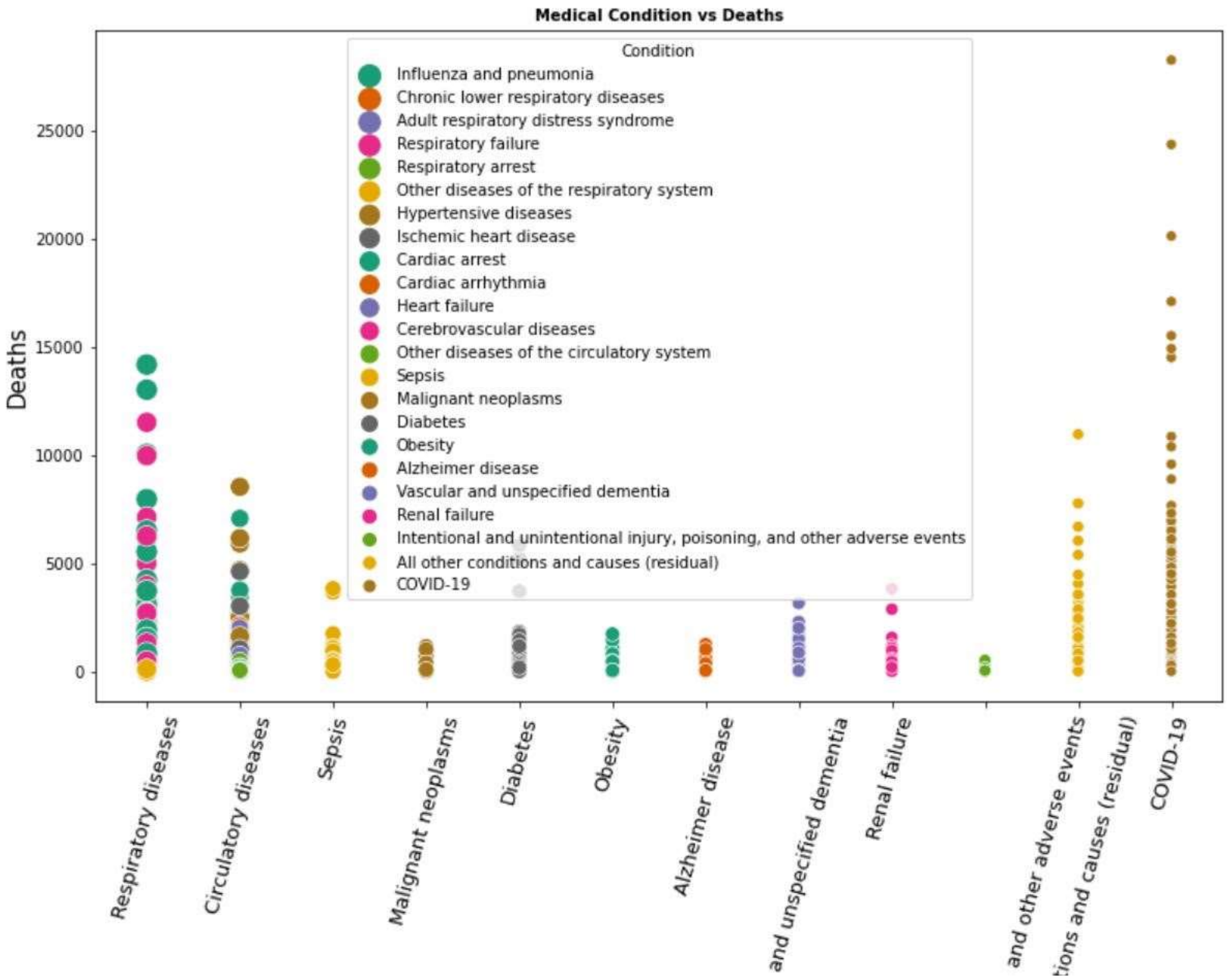


```
df
```

State	Condition Group	Condition	ICD10_codes	Age Group	COVID-19 Deaths	Population	poverty %	% older than 65 years	Food Stamp Recipient %	Edu. < College %	Edu. < Highschool %	CV Death per 100K	Mask_Mandate	M
AL	Respiratory diseases	Influenza and pneumonia	J09-J18	0-24	NaN	4795735.0	17.268944	15.722246	17.824077	75.348869	75.348869	561.024618	1.594872e+09	
AL	Respiratory diseases	Influenza and pneumonia	J09-J18	25-34	20.0	4795735.0	17.268944	15.722246	17.824077	75.348869	75.348869	561.024618	1.594872e+09	
AL	Respiratory diseases	Influenza and pneumonia	J09-J18	35-44	34.0	4795735.0	17.268944	15.722246	17.824077	75.348869	75.348869	561.024618	1.594872e+09	
AL	Respiratory diseases	Influenza and pneumonia	J09-J18	45-54	90.0	4795735.0	17.268944	15.722246	17.824077	75.348869	75.348869	561.024618	1.594872e+09	
AL	Respiratory diseases	Influenza and pneumonia	J09-J18	55-64	216.0	4795735.0	17.268944	15.722246	17.824077	75.348869	75.348869	561.024618	1.594872e+09	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
PR	COVID-19	COVID-19	U071	65-74	321.0	3263022.0	15.883556	18.054035	14.399250	75.324192	75.324192	295.945035	1.524666e+09	
PR	COVID-19	COVID-19	U071	75-84	366.0	3263022.0	15.883556	18.054035	14.399250	75.324192	75.324192	295.945035	1.524793e+09	
PR	COVID-19	COVID-19	U071	85+	264.0	3263022.0	15.883556	18.054035	14.399250	75.324192	75.324192	295.945035	1.524921e+09	
PR	COVID-19	COVID-19	U071	Not stated	0.0	3263022.0	15.883556	18.054035	14.399250	75.324192	75.324192	295.945035	1.525048e+09	
PR	COVID-19	COVID-19	U071	All Ages	1290.0	3263022.0	15.883556	18.054035	14.399250	75.324192	75.324192	295.945035	1.525176e+09	

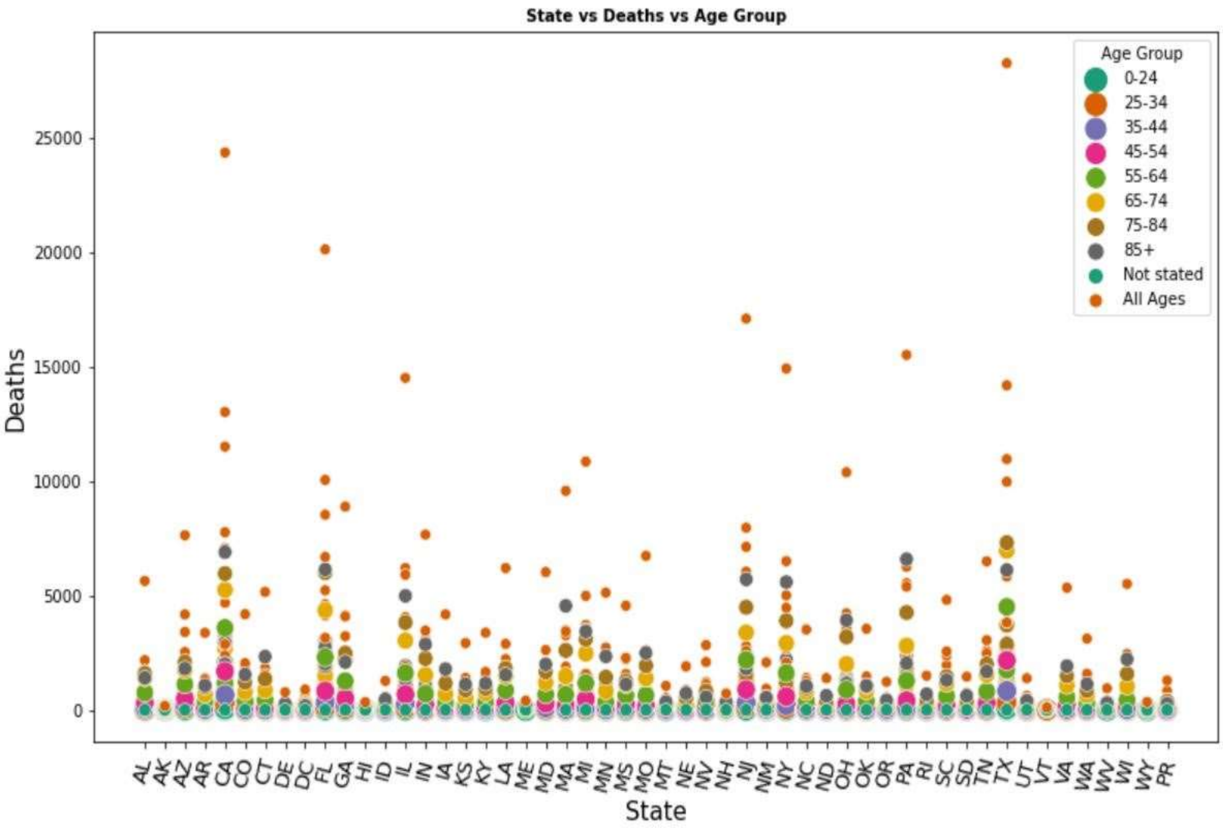
3. Exploratory Data Analysis

3.1 Medical conditions Vs Covid-19 deaths



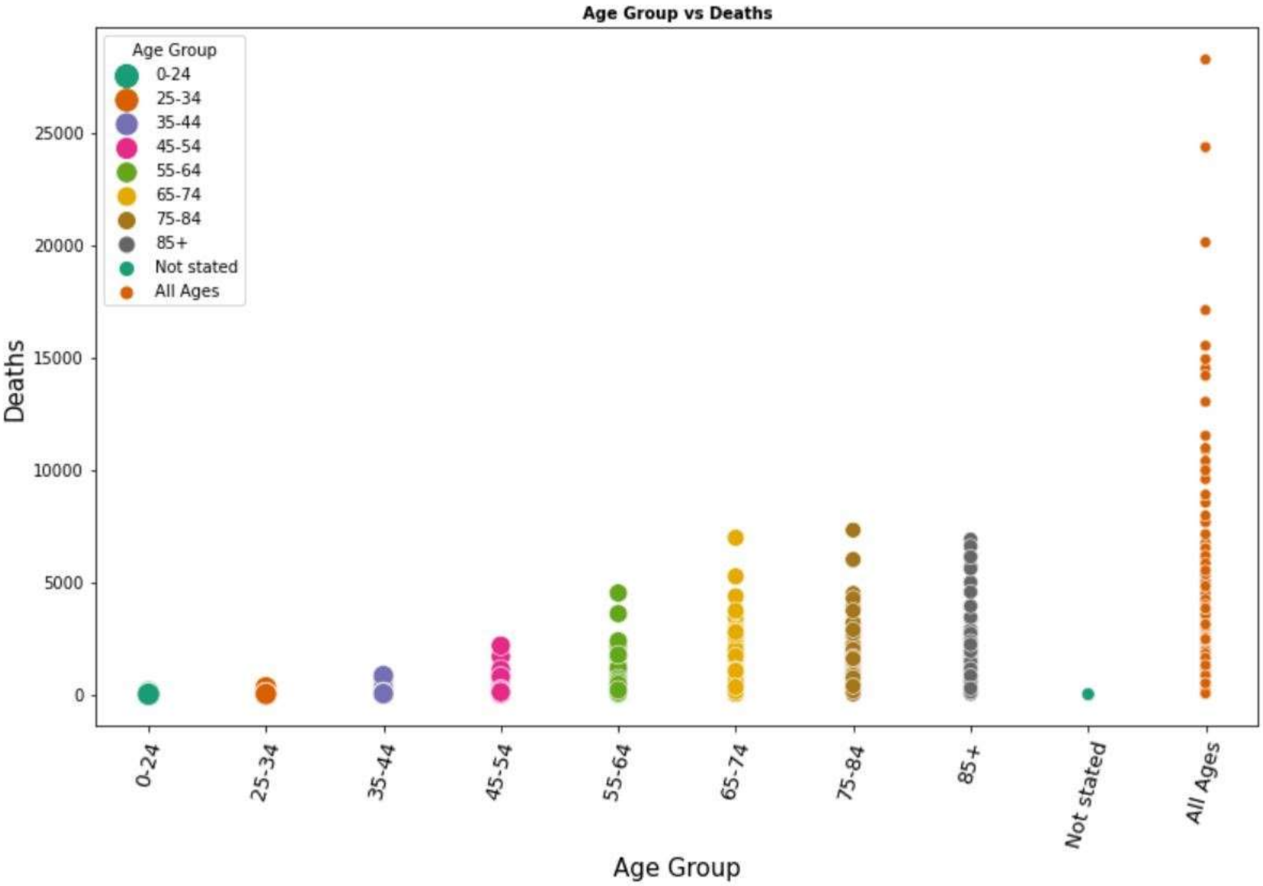
The x axis contains the medical condition groups and the y axis contains the number of Covid-19 deaths. The sizes of the bubbles indicate the magnitude of a medical condition's contribution to Covid-19 deaths. It seems that respiratory conditions, in particular Influenza and Pneumonia have had the most Covid-19 deaths.

3.2 State vs Deaths vs Age Group



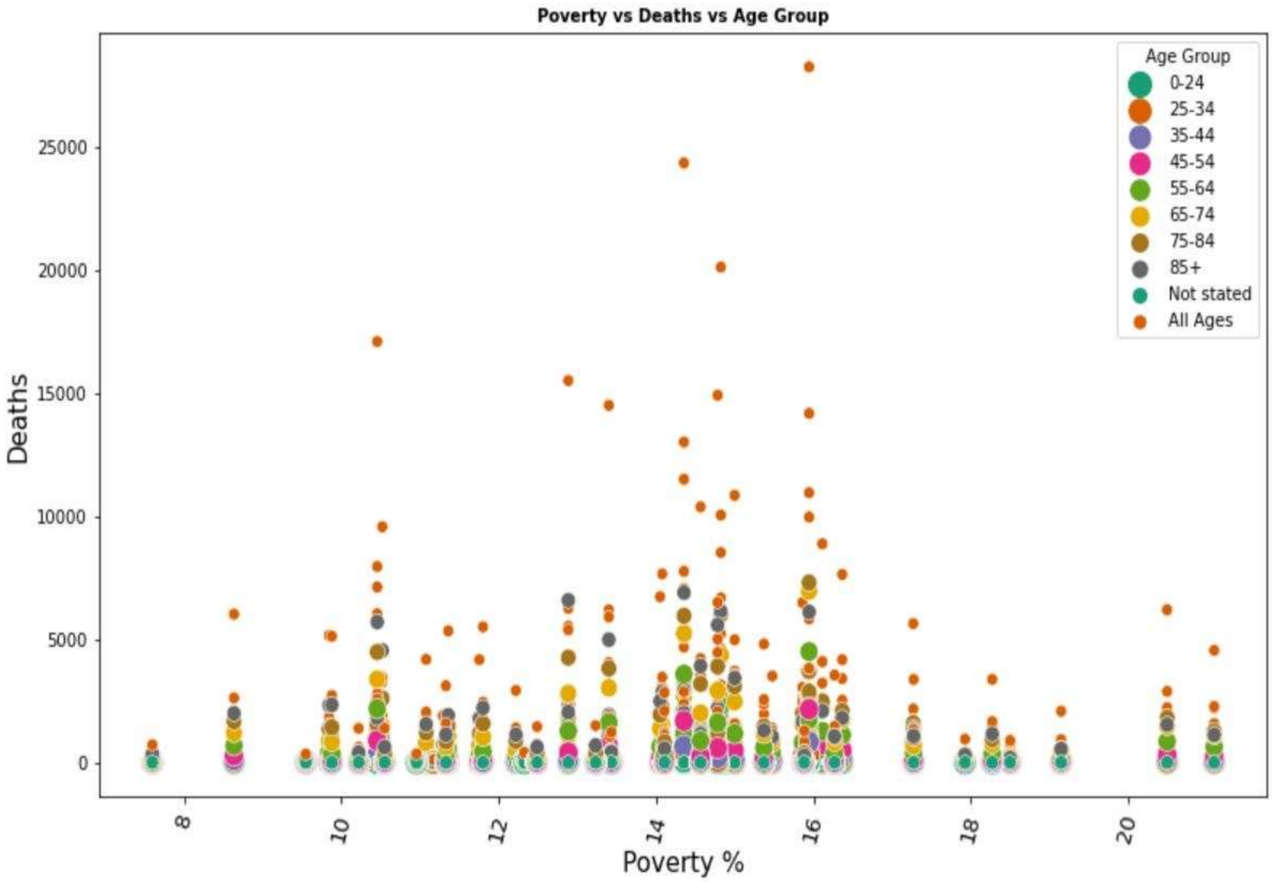
The x axis contains States and the y axis contains the number of Covid-19 deaths. The sizes of the bubbles indicate the magnitude of an age group's contribution to Covid-19 deaths. It seems that in all states, the most deaths belong to age groups 55-64 or above.

3.3 Deaths vs Age Group



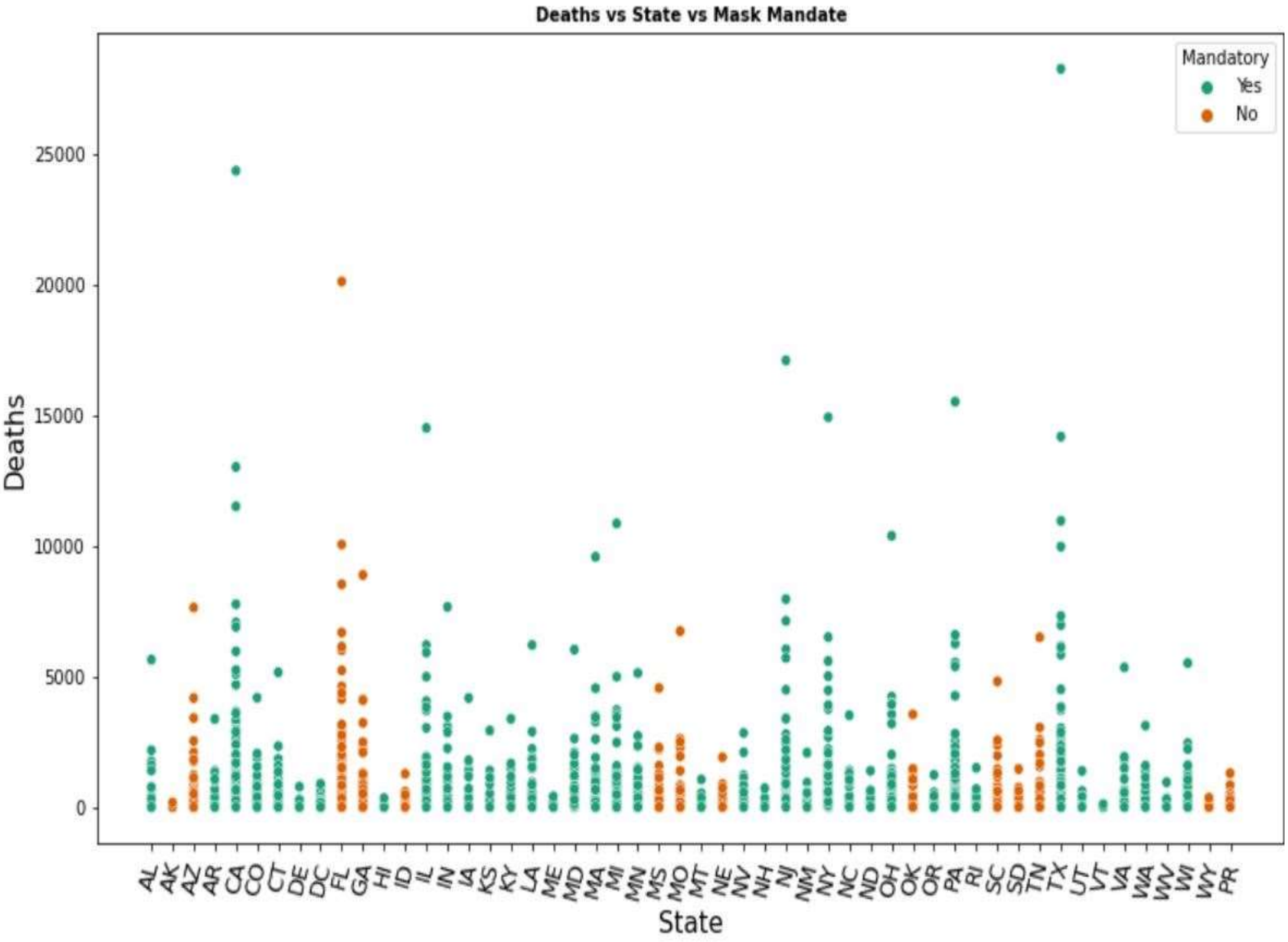
The x axis contains Age Groups and the y axis contains the number of Covid-19 deaths. It is clear from the graphs that the older age groups have been more vulnerable to the pandemic than the younger ones.

3.4 Poverty vs Deaths vs Age Group



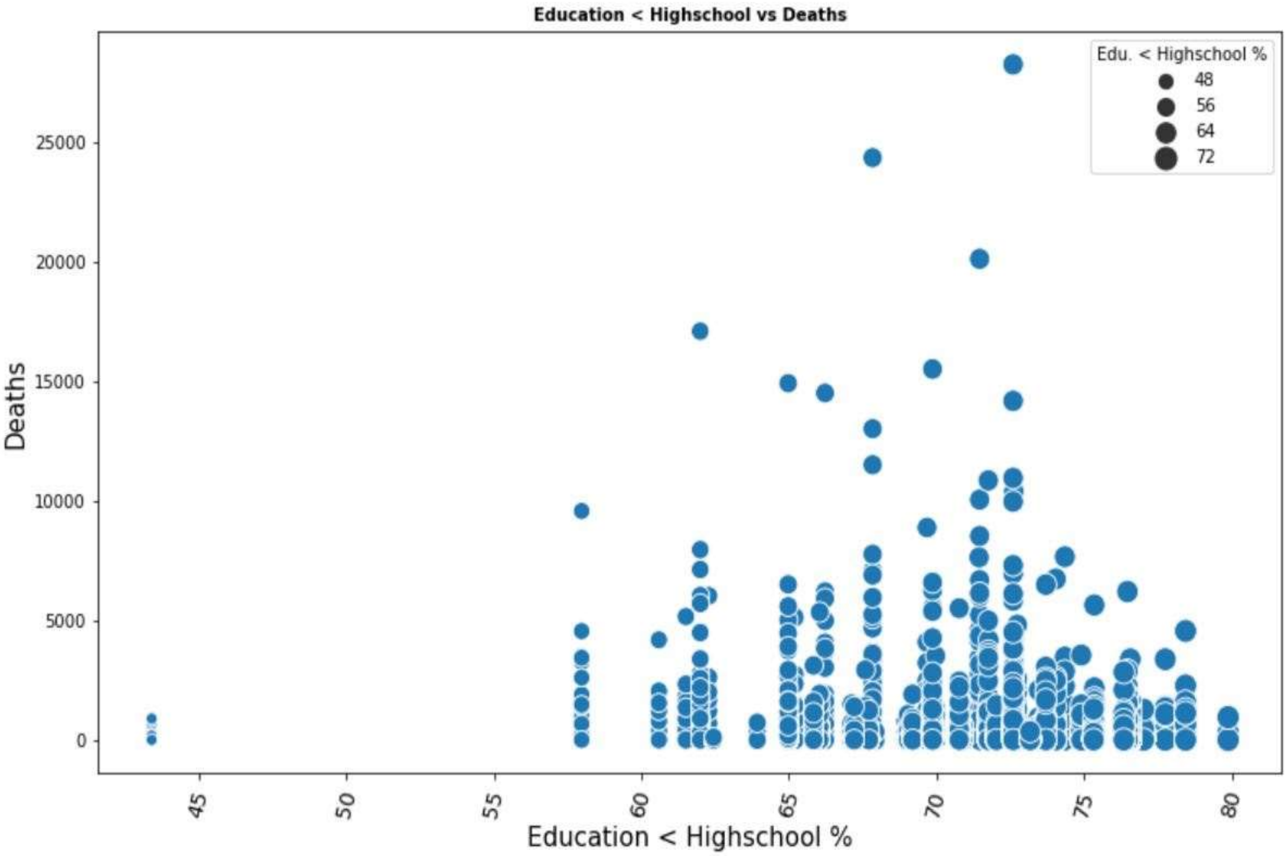
The x axis contains the Poverty % and the y axis contains the number of Covid-19 deaths. The sizes of the bubbles indicate the magnitude of an age group's contribution to Covid-19 deaths. It seems that people in the age group 65-74 in poverty have been the most vulnerable to the pandemic.

3.5 Deaths vs State vs Mask Mandate



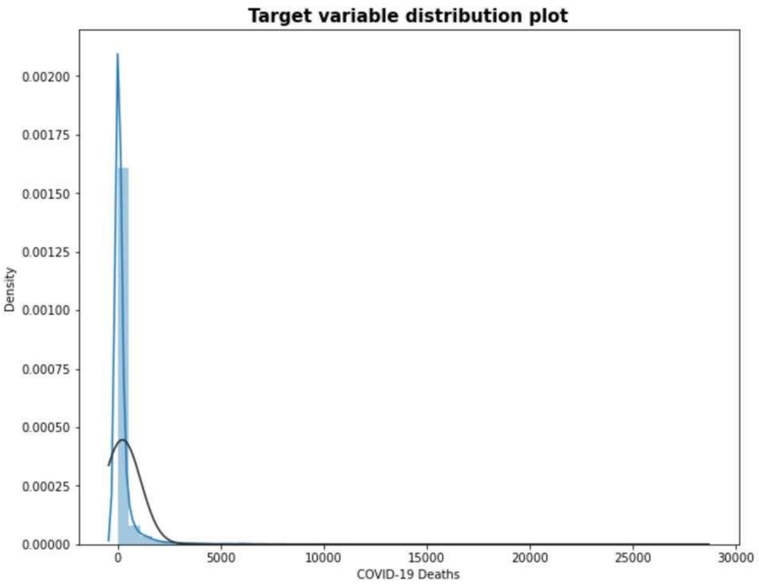
The x axis contains States and the y axis contains the number of Covid-19 deaths. It seems that people in the age group 65-74 in poverty have been the most vulnerable to the pandemic.

3.6 Education < High School vs Deaths

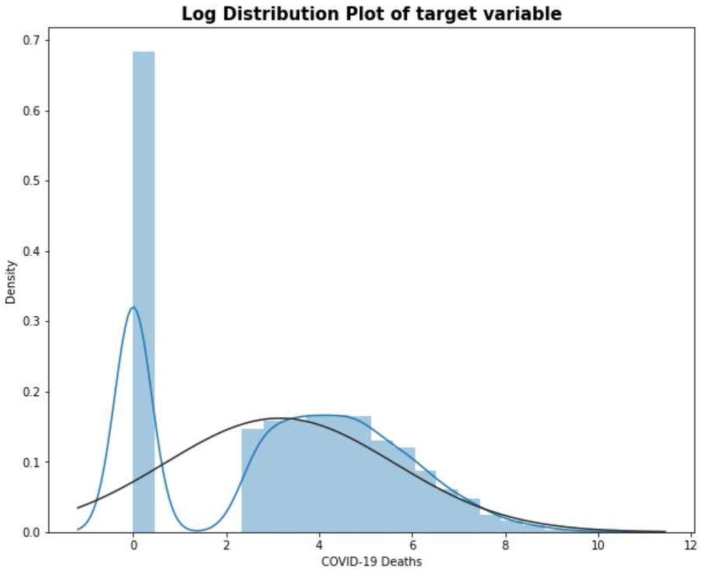


The x axis contains the percentage of population that has education less than high school and the y axis contains the number of Covid-19 deaths. The sizes of the bubbles indicate the magnitude of an Education<High school group's contribution to Covid-19 deaths. It seems that the population with the most <High school education % have been affected the most.

3.7 Examining target variable distributio



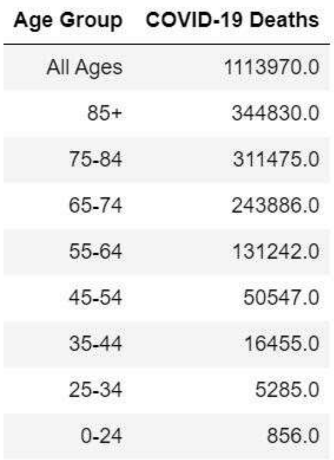
The target variable ‘Covid-19 Deaths’ has a right skewed distribution, log transforming it will help normalize the distribution and could also help during the predictive modeling stage.



The distribution seems to be a lot more normal after log transformation.

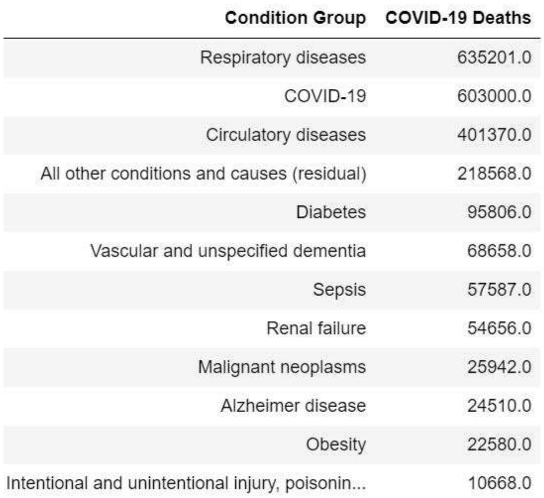
4. Descriptive statistics

4.1 Age Group vs Deaths



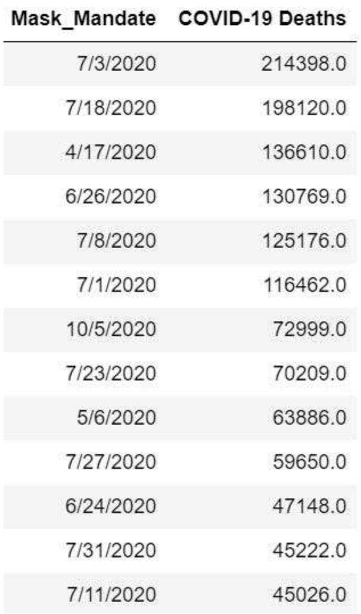
Age Group	COVID-19 Deaths
All Ages	1113970.0
85+	344830.0
75-84	311475.0
65-74	243886.0
55-64	131242.0
45-54	50547.0
35-44	16455.0
25-34	5285.0
0-24	856.0

4.2 Condition Group vs Deaths



Condition Group	COVID-19 Deaths
Respiratory diseases	635201.0
COVID-19	603000.0
Circulatory diseases	401370.0
All other conditions and causes (residual)	218568.0
Diabetes	95806.0
Vascular and unspecified dementia	68658.0
Sepsis	57587.0
Renal failure	54656.0
Malignant neoplasms	25942.0
Alzheimer disease	24510.0
Obesity	22580.0
Intentional and unintentional injury, poisonin...	10668.0

4.3 Mask_Mandate vs Deaths



Mask_Mandate	COVID-19 Deaths
7/3/2020	214398.0
7/18/2020	198120.0
4/17/2020	136610.0
6/26/2020	130769.0
7/8/2020	125176.0
7/1/2020	116462.0
10/5/2020	72999.0
7/23/2020	70209.0
5/6/2020	63886.0
7/27/2020	59650.0
6/24/2020	47148.0
7/31/2020	45222.0
7/11/2020	45026.0

5. Predictive Modeling

We have used the regression algorithms below to predict the number of deaths in each instance:

1) Linear Regression
2) Decision Tree Regressor
3) Support Vector Regressor
4) Random Forest Regressor

We decided to use the above algorithms because we believed tree based regression algorithms would work well with our data set since we had a good range of categorical features such as ‘Age Group’ and ‘Condition Group’ that had a high correlativity with the target attribute.

5.1 Linear Regression

Linear regression is a basic and commonly used type of predictive analysis. The overall idea of regression is to examine two things: (1) does a set of predictor variables do a good job in predicting an outcome (dependent) variable? (2) Which variables in particular are significant predictors of the outcome variable, and in what way do they–indicated by the magnitude and sign of the beta estimates–impact the outcome variable? These regression estimates are used to explain the relationship between one dependent variable and one or more independent variables. The simplest form of the regression equation with one dependent and one independent variable is defined by the formula y = c + b*x, where y = estimated dependent variable score, c = constant, b = regression coefficient, and x = score on the independent variable. First, the regression might be used to identify the strength of the effect that the independent variable(s) have on a dependent variable. Typical questions are what is the strength of relationship between dose and effect, sales and marketing spending, or age and income. Second, it can be used to forecast effects or impact of changes. That is, the regression analysis helps us to understand how much the dependent variable changes with a change in one or more independent variables. A typical question is, "how much additional sales income do I get for each additional $1000 spent on marketing?" Third, regression analysis predicts trends and future values. The regression analysis can be used to get point estimates. A typical question is, "what will the price of gold be in 6 months?"

5.2 Decision Tree Regressor

Decision tree builds regression or classification models in the form of a tree structure. It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes. A decision node (e.g., Outlook) has two or more branches (e.g., Sunny, Overcast and Rainy), each representing values for the attribute tested. Leaf node (e.g., Hours Played) represents a decision on the numerical target. The topmost decision node in a tree which corresponds to the best predictor called root node. Decision trees can handle both categorical and numerical data.

5.3 Support Vector Regressor

Support Vector Machine can also be used as a regression method, maintaining all the main features that characterize the algorithm (maximal margin). The Support Vector Regression (SVR) uses the same principles as the SVM for classification, with only a few minor differences. First of all, because output is a real number it becomes very difficult to predict the information at hand, which has infinite possibilities. In the case of regression, a margin of tolerance (epsilon) is set in approximation to the SVM which would have already requested from the problem. But besides this fact, there is also a more complicated reason, the algorithm is more complicated therefore to be taken in consideration. However, the main idea is always the same: to minimize error, individualizing the hyperplane which maximizes the margin, keeping in mind that part of the error is tolerated.

5.4 Random Forest Regressor

A Random Forest is an ensemble technique capable of performing both regression and classification tasks with the use of multiple decision trees and a technique called Bootstrap and Aggregation, commonly known as bagging. The basic idea behind this is to combine multiple decision trees in determining the final output rather than relying on individual decision trees. Random Forest has multiple decision trees as base learning models. We randomly perform row sampling and feature sampling from the dataset forming sample datasets for every model. This part is called Bootstrap. We need to approach the Random Forest regression technique like any other machine learning technique:

1. Design a specific question or data and get the source to determine the required data.

2. Make sure the data is in an accessible format else convert it to the required format.

3. Specify all noticeable anomalies and missing data points that may be required to achieve the required data.

4. Create a machine learning model

5. Set the baseline model that you want to achieve

6. Train the data machine learning model.

7. Provide an insight into the model with test data

8. Now compare the performance metrics of both the test data and the predicted data from the model.

9. If it doesn't satisfy your expectations, you can try improving your model accordingly or dating your data or use another data modeling technique.

10. At this stage you interpret the data you have gained and report accordingly.

6. Analysis and Results

6.1 Algorithms on Weka:

6.1.1 Linear Regression 66% Split:

- Correlation coefficient: 0.2352
- Mean absolute error: 0.6344
- Root mean squared error: 0.76
- Relative absolute error: 96.65 %
- Root relative squared error: 97.19 %

6.1.2 Linear Regression 10 fold cross validation:

- Correlation coefficient: 0.2262
- Mean absolute error: 0.6291
- Root mean squared error: 0.758
- Relative absolute error: 96.39 %
- Root relative squared error: 97.41 %

6.1.3 Random Forest 10 fold cross validation:

- Correlation coefficient: 0.7888
- Mean absolute error: 0.3041
- Root mean squared error: 0.4787
- Relative absolute error: 46.60 %
- Root relative squared error: 61.51 %

6.2 Algorithms using python

6.2.1 Linear Regression 80/20 split:

- Mean squared error: 5.47
- R2 score: 0.05

6.2.2 Linear Regression 10 fold cross validation:

- Mean squared error: 5.47
- R2 score: 0.05

6.2.3 Decision Tree Regressor 80/20 split:

- Mean squared error: 2.48

- R2 score: 0.57

6.2.4 Decision Tree Regressor 10 fold cross validation:

- Mean squared error: 2.58
- R2 score: 0.55

6.2.5 Support Vector Regressor 80/20 split:

- Mean squared error: 3.71
- R2 score: 0.35

6.2.6 Support Vector Regressor 10 fold cross validation:

- Mean squared error: 4.03
- R2 score: 0.30

6.2.7 Random Forest Regressor 80/20 split:

- Mean squared error: 2.02
- R2 score: 0.65

6.2.8 Random Forest Regressor 10 fold cross validation:

- Mean squared error: 2.03
- R2 score: 0.65

7. Conclusion

- Categorical attributes such as poverty %, population, medical condition, age group and mask_made showed expected trends when plotted against covid-19 deaths

- The number of deaths increased as poverty % and population increased, it also seemed to be high for respiratory medical conditions among older age groups.
- The target variable initially had a skewed distribution, but showed a more normal distribution after log transformation, which helped the prediction

Predictive modeling:

- Tree based algorithms such as Decision Tree and Random Forest produced the best results with R2 scores of 0.57 and 0.78 respectively.
- This could be because of the fact that our dataset had a lot of categorical attributes that had an effect on the target variable that numerical attributes didn't.

8. References

1. 'Factors associated with COVID-19-related death using OpenSAFELY' by Elizabeth J. Williamson, Alex J. Walker, Ben Goldacre
2. CDC data set: https://catalog.data.gov/dataset/conditions-contributing-to-deaths-involving-coronavirus-disease-2019-covid-19-by-age-group
3. Public Health Data set: https://github.com/JKRosen/DSCI-511-Project
4. Health Resources and Service Administration: hrsa.gov

9. Contributions:

Preprocessing and predictive modeling: Alicia Brandemarte, Manoj Venkatachalaiah

EDA and Descriptive statistics: Akhila Singanal, Srilakshmi Rao