COVID-19 Cases Prediction

1. Introduction and background of project:

Coronavirus in general called COVID-19 is a widespread infectious disease that affected many people in different ways and has been spreading rapidly over the past several months and the US Death toll has reached 400,000. COVID-19 is dangerous not only for the elderly but for mid-aged adults and the people are affected in different ways and the infected people have a wide range of symptoms reported – from mild symptoms to severe illness or even death, few people recovered without any special treatment.

This project focuses on using Machine Learning Model for Predicting COVID-19 cases for the next 60 days and thus these types of predictive models help in providing an accurate prediction of epidemics, which is essential for obtaining information on the likely spread and consequences of infectious diseases. I have chosen the dataset which is necessary to analyse the past and the current data to predict which country will be affected or will have more COVID-19 cases and deaths further.

2. Statement of the Project Problem:

In this project, I am taking the data from the present and past COVID–19 cases which includes deaths and confirmed cases and predict the cases for the next 60 days. During this process of getting accurate data, I am going to perform data cleaning and preparing data, Analysing data, build machine learning model on the data, validate my model by Analysing and tuning and finally integrating UI and ML Model. By this prediction, the public can take more precautions by wearing a face mask, maintain social distancing, sanitizing, and handwashing to prevent the spread and these predictions might help older people, and those with underlying medical problems like cardiovascular disease, diabetes, chronic respiratory disease, and cancer.

3. Review of Literature:

Before investigating the literature on COVID-19, we conducted a brief bibliographic search about COVID-19 for two purposes. Firstly, to find out the number of research works published on COVID-19 and secondly, to identify the different research areas focusing on COVID-19. For those purposes, we used some keywords like COVID-19 and Coronavirus. It was found that more than 8000 research documents have been published on this topic. Figure 1 shows the different types of research documents published on COVID-19.

4. Related Work:

In the past, I have worked on COVID–19 data by scrapping the Twitter tweets. And thus, performed the Sentiment Analysis and topic modelling. I want to work further on this topic since COVID-19 is still getting worse in many countries.

In the past *Boosted Random Forest Algorithm* is used for COVID-19 Patient Health Prediction and the dataset has been compiled from World Health Organization and John Hopkins University and after using multiple algorithm models, Boosted Random Forest Algorithm is the best performing model, and the dataset was finely tuned for better performance **[1].** Forecasting the spread of COVID-19 under different reopening strategies used COVID-19 case data mobility data to estimate a modified *susceptible infected recovered* (SIR) model where this model focuses on forecasts that the number of COVID-19 cases would have an exponential growth for a brief period at the beginning of the contagion event or right after a reopening but would quickly settle into a prolonged period **[2].** The major drawback of statistical methods and mathematical modelling is their inability to consider massive amounts of data. This leads to poor prediction of the number of COVID-19 cases. This drawback can be avoided by using data analytics, which is explained in the next section.

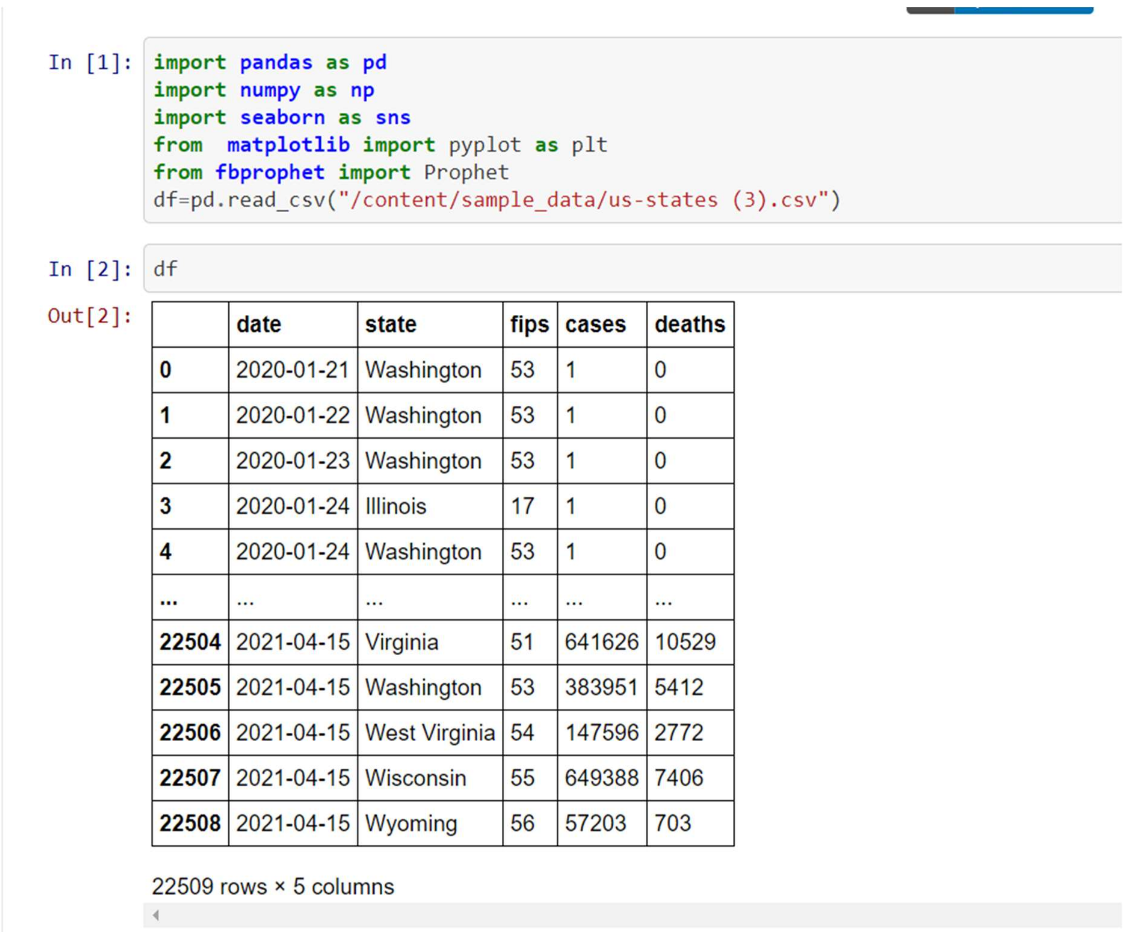
5. Objective of the Study:

The main objective of this study is to predict future spread based on the existing spread data, these predictions might help government and other legislative bodies can rely on these kinds of machine learning predictive models and ideas to suggest new policies and assess the effectiveness of applied policies.

6. Data Collection:

The dataset consists of death data and confirmed cases due to COVID-19, where columns consist of Date, Cases, Deaths, Fips and states that got affected with COVID-19. The data is collected from 21st January 2020 to 15th April 2021. It contains 22509 rows and 5 columns.

Dataset URL: https://github.com/nytimes/covid-19-data/blob/master/us-states.csv



```
In [1]: import pandas as pd
        import numpy as np
        import seaborn as sns
        from  matplotlib import pyplot as plt
        from fbprophet import Prophet
        df=pd.read_csv("/content/sample_data/us-states (3).csv")

In [2]: df

Out[2]:
```

	date	state	fips	cases	deaths
0	2020-01-21	Washington	53	1	0
1	2020-01-22	Washington	53	1	0
2	2020-01-23	Washington	53	1	0
3	2020-01-24	Illinois	17	1	0
4	2020-01-24	Washington	53	1	0
...	...	...	...	...	...
22504	2021-04-15	Virginia	51	641626	10529
22505	2021-04-15	Washington	53	383951	5412
22506	2021-04-15	West Virginia	54	147596	2772
22507	2021-04-15	Wisconsin	55	649388	7406
22508	2021-04-15	Wyoming	56	57203	703

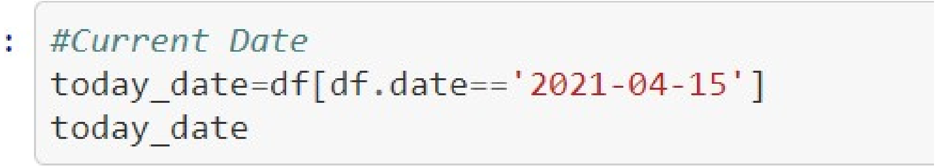
22509 rows × 5 columns

7. Exploratory Data Analysis:

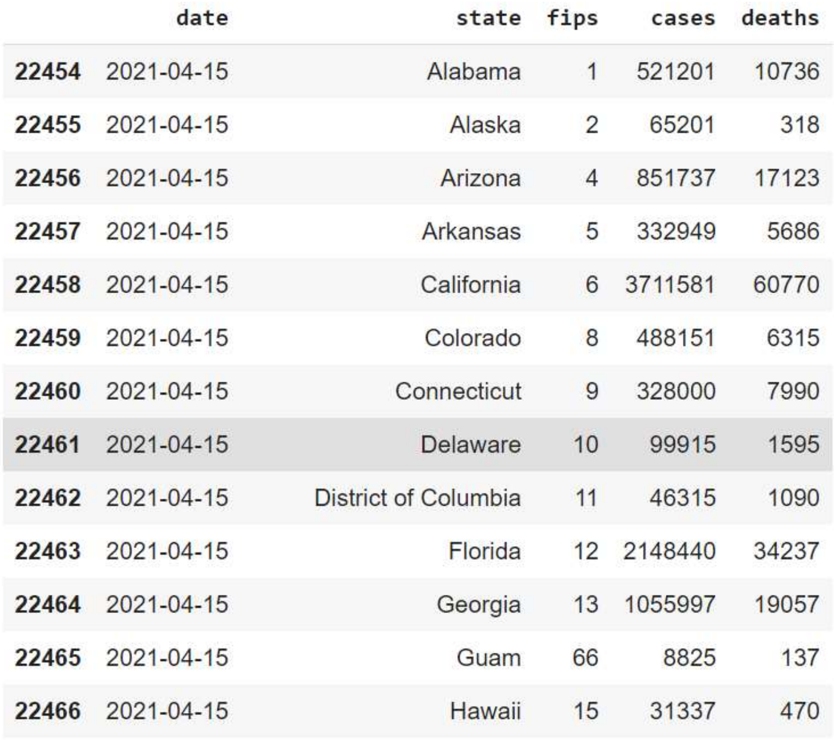
Data Pre-processing:

While collecting the data, it was observed that data were not always available for the whole 50 States. To alleviate this situation, a refinement was performed by ruling out any states that suffer from data unavailability. This results in cancelling around 3 states so that only 47 states have been considered. Our preliminary goal is to predict the new cases for all 47 states. However, this is not reasonable for two reasons. Firstly, it is not possible to present all the results in a single study due to page limitation.

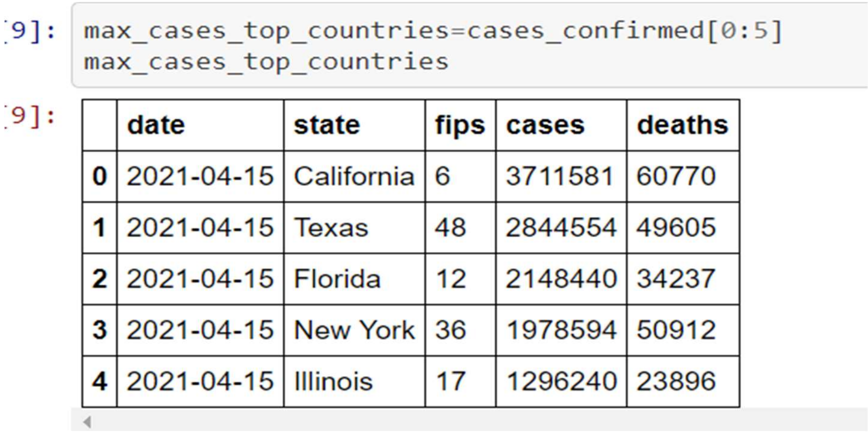
From the data, we have filtered the cases and deaths from US states until 15th of April 2021.



```
#Current Date
today_date=df[df.date=='2021-04-15']
today_date
```



	date	state	fips	cases	deaths
22454	2021-04-15	Alabama	1	521201	10736
22455	2021-04-15	Alaska	2	65201	318
22456	2021-04-15	Arizona	4	851737	17123
22457	2021-04-15	Arkansas	5	332949	5686
22458	2021-04-15	California	6	3711581	60770
22459	2021-04-15	Colorado	8	488151	6315
22460	2021-04-15	Connecticut	9	328000	7990
22461	2021-04-15	Delaware	10	99915	1595
22462	2021-04-15	District of Columbia	11	46315	1090
22463	2021-04-15	Florida	12	2148440	34237
22464	2021-04-15	Georgia	13	1055997	19057
22465	2021-04-15	Guam	66	8825	137
22466	2021-04-15	Hawaii	15	31337	470

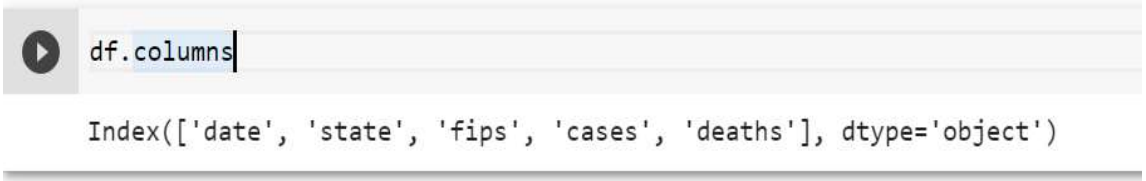


```
9]: max_cases_top_countries=cases_confirmed[0:5]
    max_cases_top_countries
```

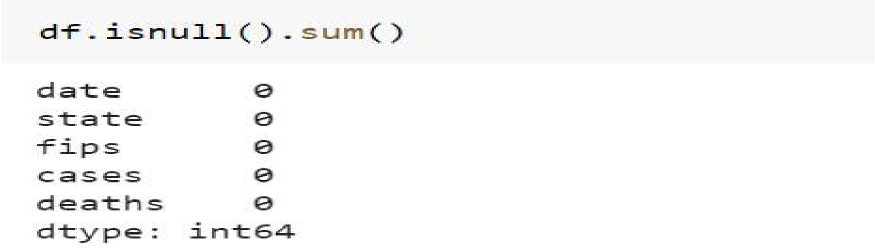
9]:

	date	state	fips	cases	deaths
0	2021-04-15	California	6	3711581	60770
1	2021-04-15	Texas	48	2844554	49605
2	2021-04-15	Florida	12	2148440	34237
3	2021-04-15	New York	36	1978594	50912
4	2021-04-15	Illinois	17	1296240	23896

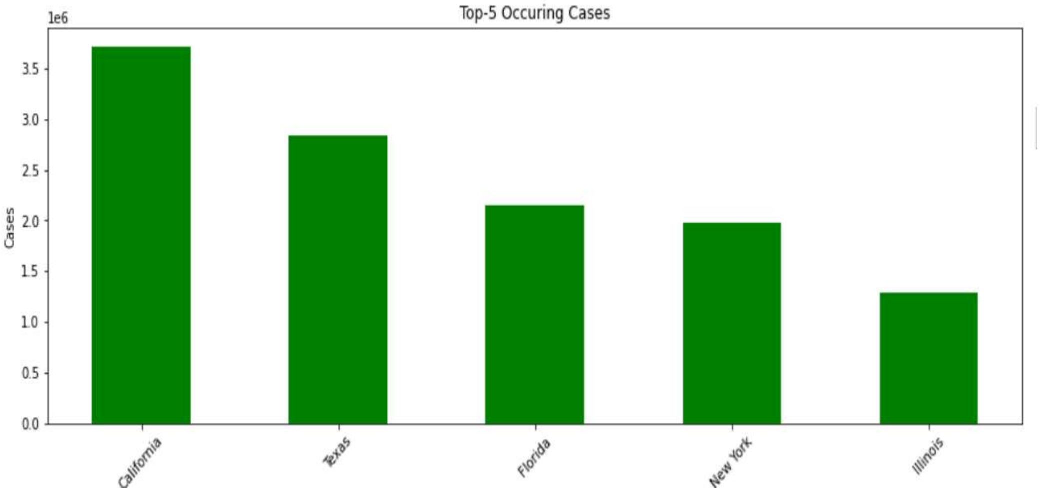
Above data frame tells us which stated has highest number of cases



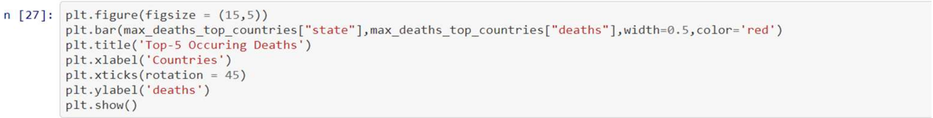
```
df.columns

Index(['date', 'state', 'fips', 'cases', 'deaths'], dtype='object')
```



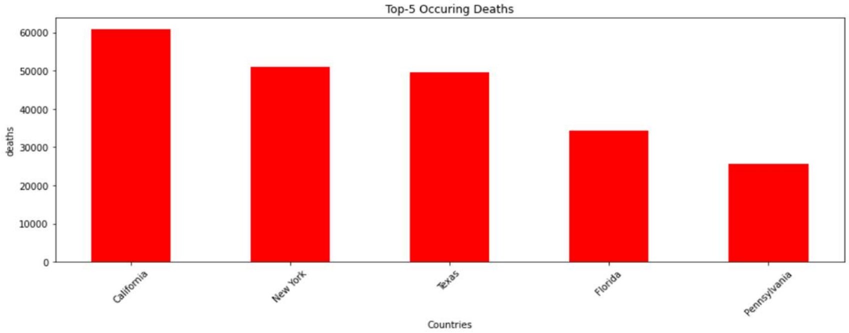
```
df.isnull().sum()

date      0
state     0
fips      0
cases     0
deaths    0
dtype: int64
```



Above Bar Graph gives us the Top-5 occurring cases.

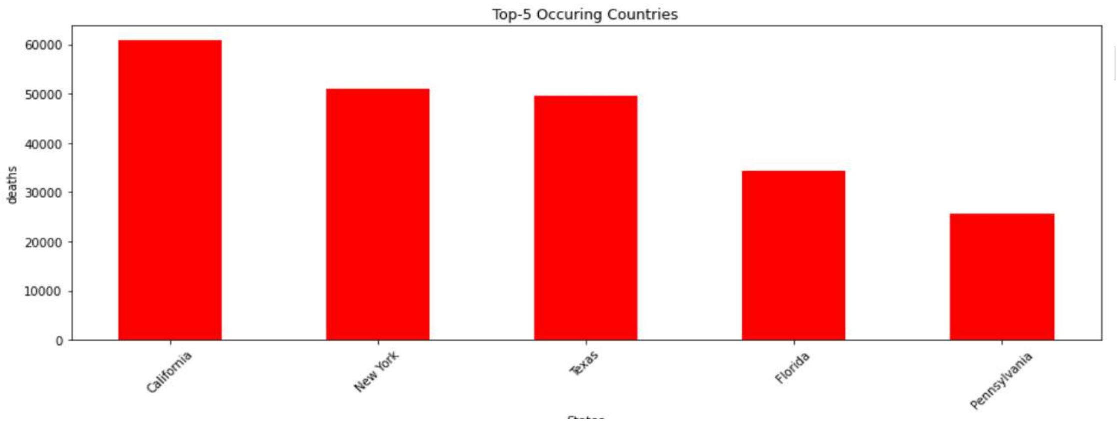


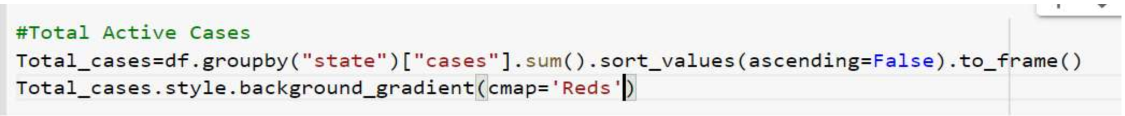
```
n [27]: plt.figure(figsize = (15,5))
        plt.bar(max_deaths_top_countries["state"],max_deaths_top_countries["deaths"],width=0.5,color='red')
        plt.title('Top-5 Occuring Deaths')
        plt.xlabel('Countries')
        plt.xticks(rotation = 45)
        plt.ylabel('deaths')
        plt.show()
```



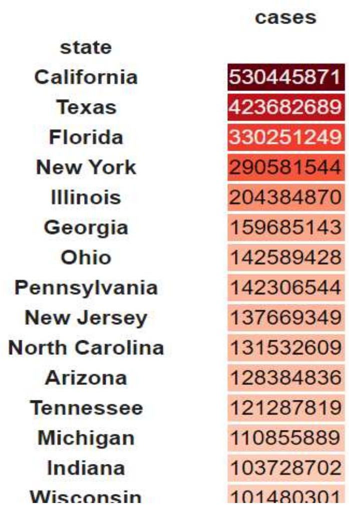
Above Bar Graph gives us the Top-5 occurring deaths. California states has highest deaths while Pennsylvania has lowest deaths.

8. Data Analytics:





```
#Total Active Cases
Total_cases=df.groupby("state")["cases"].sum().sort_values(ascending=False).to_frame()
Total_cases.style.background_gradient(cmap='Reds')
```



state	cases
California	530445871
Texas	423682689
Florida	330251249
New York	290581544
Illinois	204384870
Georgia	159685143
Ohio	142589428
Pennsylvania	142306544
New Jersey	137669349
North Carolina	131532609
Arizona	128384836
Tennessee	121287819
Michigan	110855889
Indiana	103728702
Wisconsin	101480301

From the above screenshot we can see dark colour got decreased when cases count decreased. California states has darkest colour because it has highest cases.

Hypothesis:

Null Hypothesis (H0) : Cases increases in each state for every week.

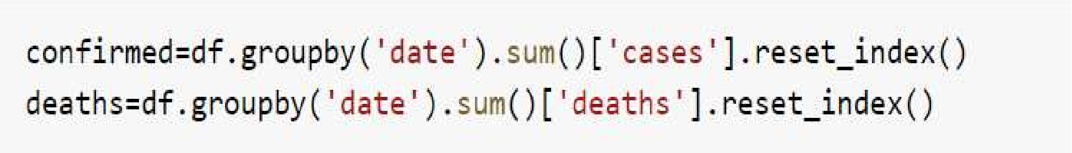
Alternative Hypothesis (H1): No increase in the cases for every week in each state.

9. Research Design and Methodology:

By collecting data through web source and then I am going to perform data pre-processing steps and make my data ready for analysis. In this project, I will perform the cleaning and preparation of the dataset. And performing the exploratory data analysis, data modelling, data analysis for prediction of COVID–19 cases for the next 60 days and performing the visualizations. Then I will develop a simple web application to deploy my model in it.

10. Data Analytics:

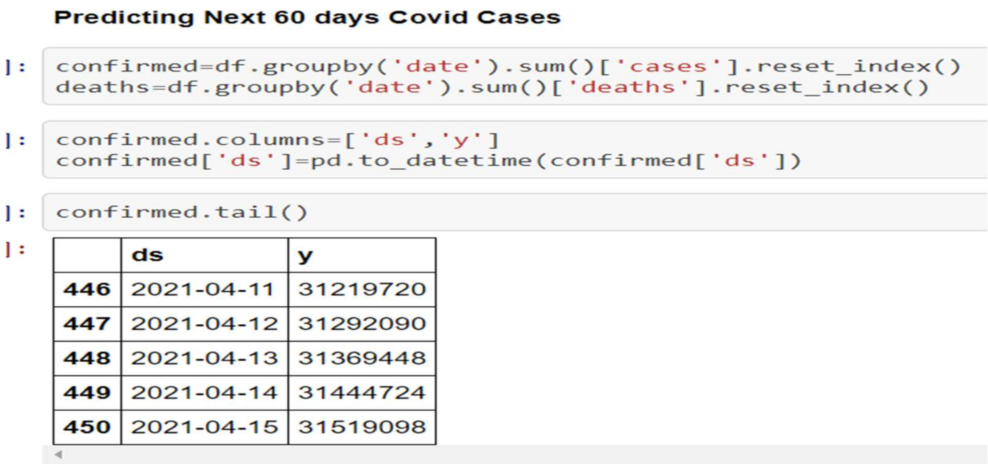
Before fitting the model, we will group by deaths and countries based on date then we can predict covid cases for next 60 days.



```
confirmed=df.groupby('date').sum()['cases'].reset_index()
deaths=df.groupby('date').sum()['deaths'].reset_index()
```

Then we will use **Prophet** model for predicting cases and deaths for next 60 days. Prophet is an open-source software released by Facebook. Prophet is a time series data forecasting technique based on an additive model that suits non-linear patterns with annual, weekly, and regular seasonality, as well as holiday impacts. It fits best with time series with heavy seasonal effects and historical data from several seasons. Prophet is forgiving of missing data and pattern changes, and it usually handles outliers well.

We need to be sure before fitting data to the model there should be two attributes in the data. One attribute is dates and column name should be **ds** and other columns is target variable and column name should be **y.**

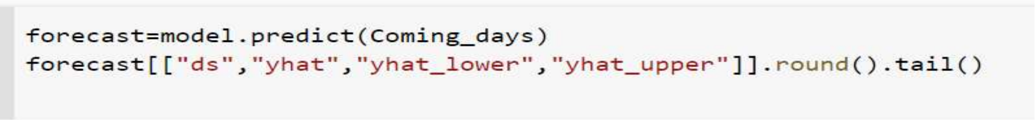


Predicting Next 60 days Covid Cases

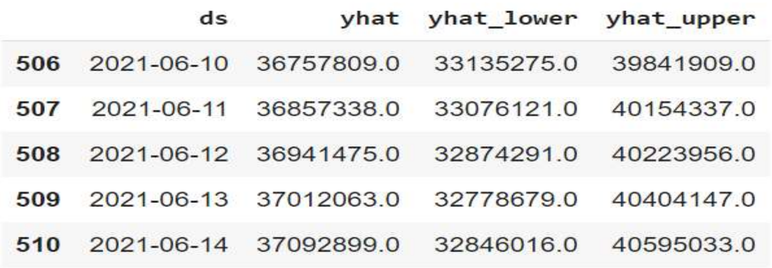
```
]: confirmed=df.groupby('date').sum()['cases'].reset_index()
   deaths=df.groupby('date').sum()['deaths'].reset_index()

]: confirmed.columns=['ds','y']
   confirmed['ds']=pd.to_datetime(confirmed['ds'])

]: confirmed.tail()
]:
```

	ds	y
446	2021-04-11	31219720
447	2021-04-12	31292090
448	2021-04-13	31369448
449	2021-04-14	31444724
450	2021-04-15	31519098

Now, we will fit our data to the model to predict cases for next 60 days.



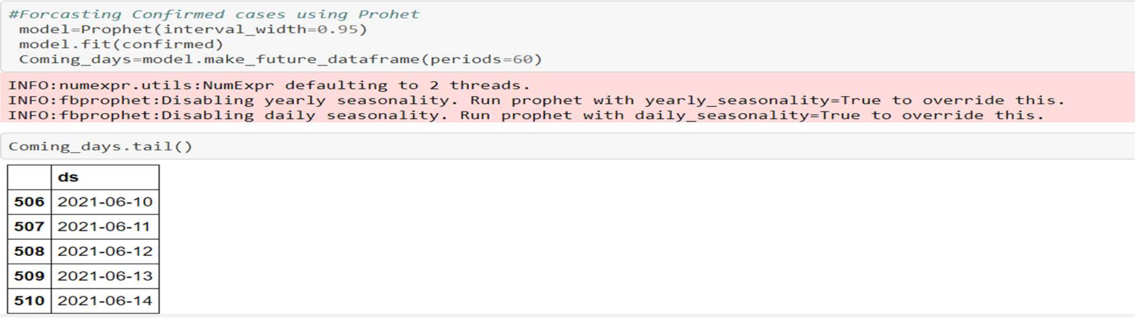
```
forecast=model.predict(Coming_days)
forecast[["ds","yhat","yhat_lower","yhat_upper"]].round().tail()
```



	ds	yhat	yhat_lower	yhat_upper
506	2021-06-10	36757809.0	33135275.0	39841909.0
507	2021-06-11	36857338.0	33076121.0	40154337.0
508	2021-06-12	36941475.0	32874291.0	40223956.0
509	2021-06-13	37012063.0	32778679.0	40404147.0
510	2021-06-14	37092899.0	32846016.0	40595033.0

Predicting Deaths for next 60 days:

Now, we will predict deaths for next 60 days.



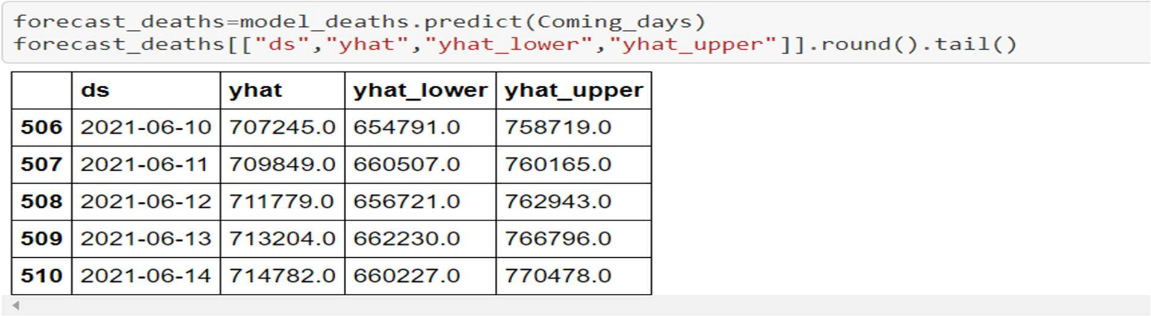
```
#Forcasting Confirmed cases using Prohet
 model=Prophet(interval_width=0.95)
 model.fit(confirmed)
 Coming_days=model.make_future_dataframe(periods=60)
```

```
INFO:numexpr.utils:NumExpr defaulting to 2 threads.
INFO:fbprophet:Disabling yearly seasonality. Run prophet with yearly_seasonality=True to override this.
INFO:fbprophet:Disabling daily seasonality. Run prophet with daily_seasonality=True to override this.
```

```
Coming_days.tail()
```

	ds
506	2021-06-10
507	2021-06-11
508	2021-06-12
509	2021-06-13
510	2021-06-14

Below is the predicted deaths for next 60 days. Yhat represents the precticted value and yhat_lower and yhat_upper represents the range that may cases get predicted.

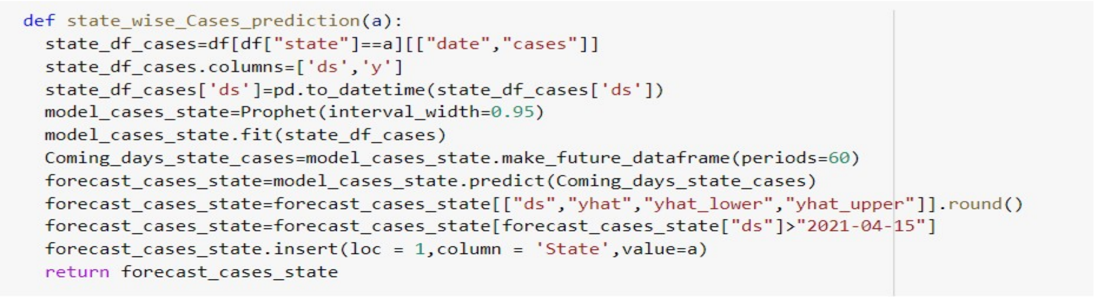


```
forecast_deaths=model_deaths.predict(Coming_days)
forecast_deaths[["ds","yhat","yhat_lower","yhat_upper"]].round().tail()
```

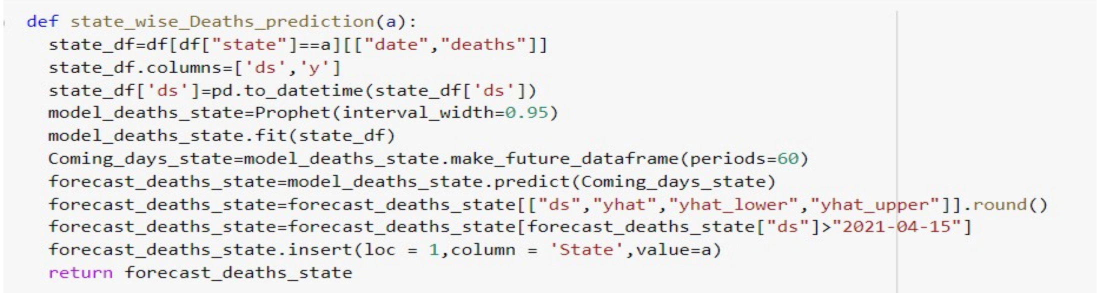
	ds	yhat	yhat_lower	yhat_upper
506	2021-06-10	707245.0	654791.0	758719.0
507	2021-06-11	709849.0	660507.0	760165.0
508	2021-06-12	711779.0	656721.0	762943.0
509	2021-06-13	713204.0	662230.0	766796.0
510	2021-06-14	714782.0	660227.0	770478.0

Predicting Cases and Deaths for next 60 days:

This below code will help us to predict cases and deaths for any state as we required.



```
def state_wise_Cases_prediction(a):
  state_df_cases=df[df["state"]==a][["date","cases"]]
  state_df_cases.columns=['ds','y']
  state_df_cases['ds']=pd.to_datetime(state_df_cases['ds'])
  model_cases_state=Prophet(interval_width=0.95)
  model_cases_state.fit(state_df_cases)
  Coming_days_state_cases=model_cases_state.make_future_dataframe(periods=60)
  forecast_cases_state=model_cases_state.predict(Coming_days_state_cases)
  forecast_cases_state=forecast_cases_state[["ds","yhat","yhat_lower","yhat_upper"]].round()
  forecast_cases_state=forecast_cases_state[forecast_cases_state["ds"]>"2021-04-15"]
  forecast_cases_state.insert(loc = 1,column = 'State',value=a)
  return forecast_cases_state
```

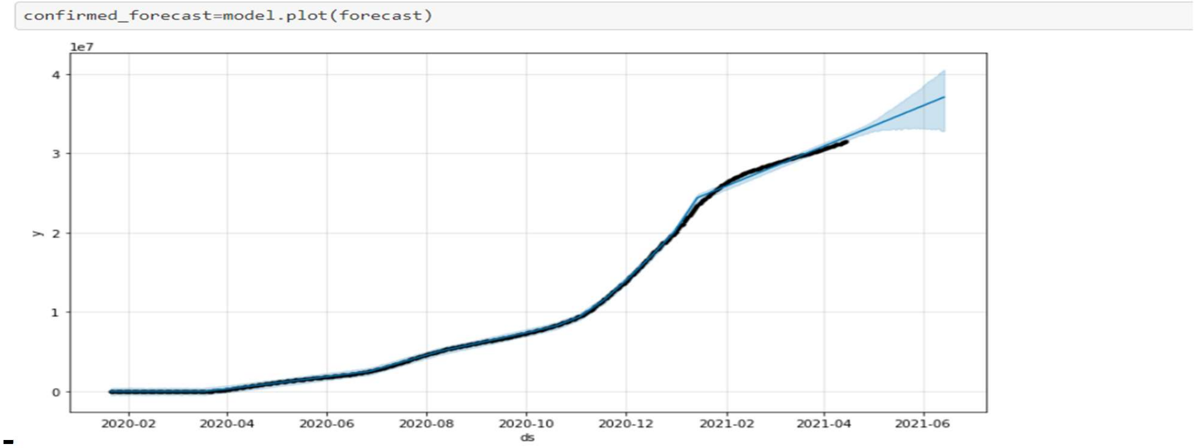


```
def state_wise_Deaths_prediction(a):
  state_df=df[df["state"]==a][["date","deaths"]]
  state_df.columns=['ds','y']
  state_df['ds']=pd.to_datetime(state_df['ds'])
  model_deaths_state=Prophet(interval_width=0.95)
  model_deaths_state.fit(state_df)
  Coming_days_state=model_deaths_state.make_future_dataframe(periods=60)
  forecast_deaths_state=model_deaths_state.predict(Coming_days_state)
  forecast_deaths_state=forecast_deaths_state[["ds","yhat","yhat_lower","yhat_upper"]].round()
  forecast_deaths_state=forecast_deaths_state[forecast_deaths_state["ds"]>"2021-04-15"]
  forecast_deaths_state.insert(loc = 1,column = 'State',value=a)
  return forecast_deaths_state
```

11. Data Visualization and Result Reports

Cases prediction for next 60 Days:

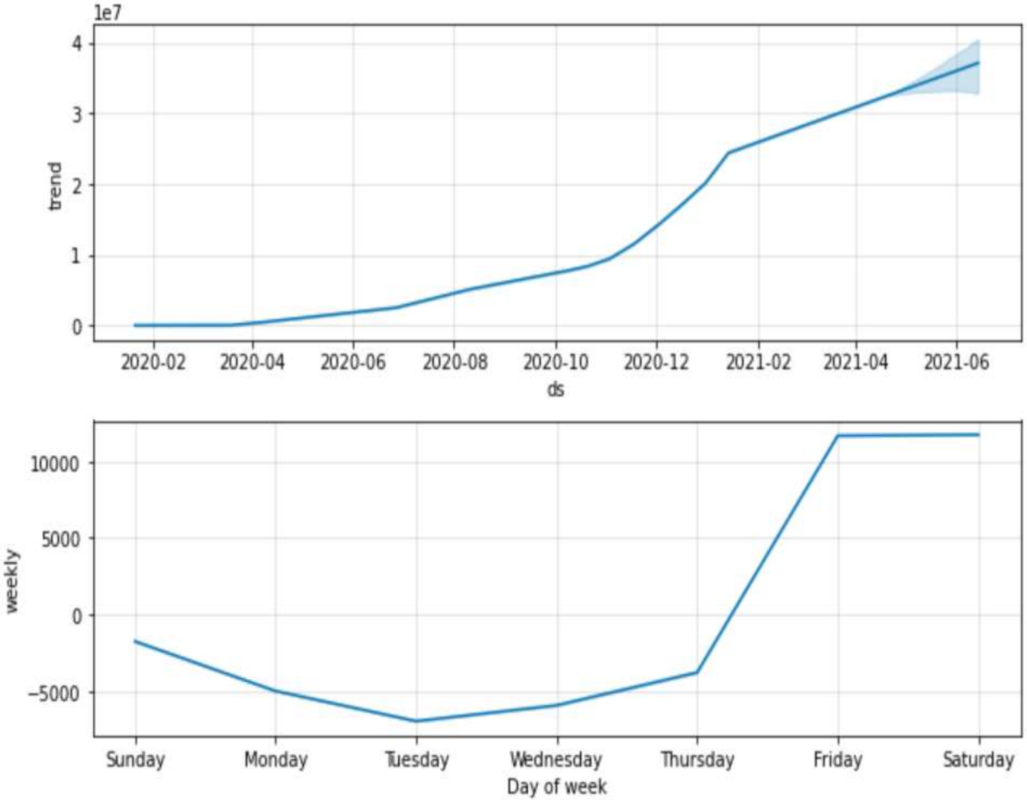
In the below line graph, we can see forecasting graph. Blue line represents the original cases and light blue line represents the next 60 days predicted cases and blue colour area represents the Max and Min range of cases for next 60 days.



```
confirmed_forecast=model.plot(forecast)
```

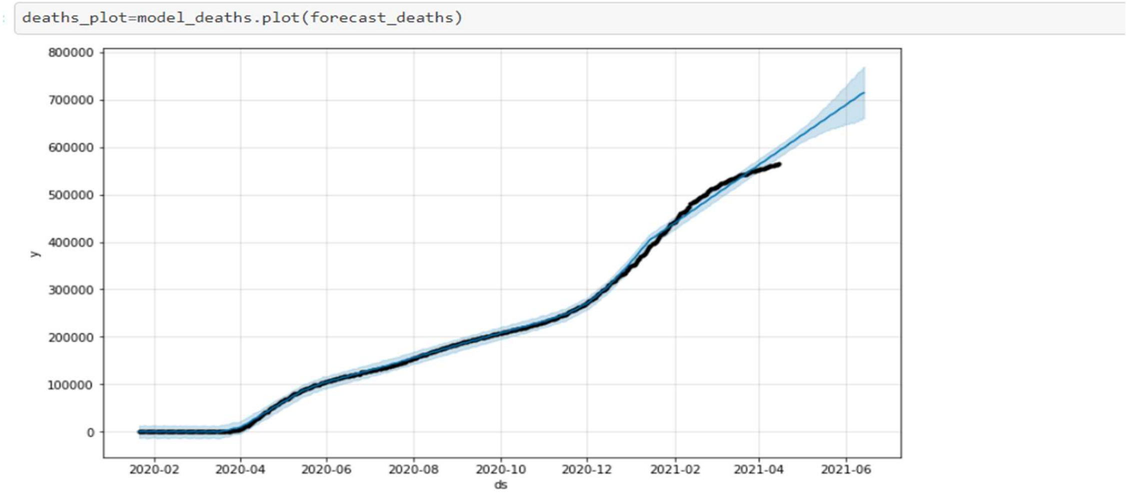
: -

Below line graphs represents the cases monthly wise and day wise. When we observe day-wise graph we can observe that cases number got increased gradually by end of the week and cases number got decreased at starting of week**.**



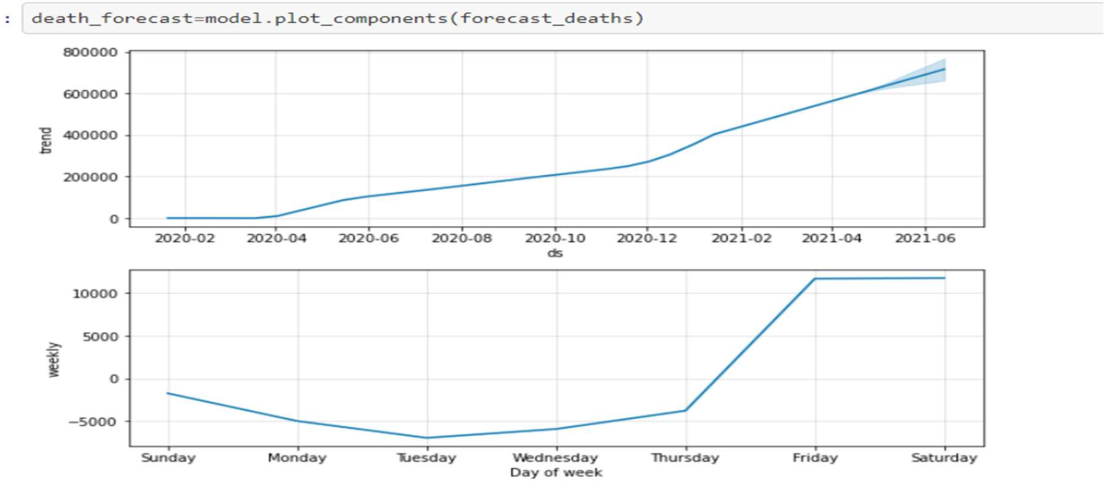
Deaths prediction for next 60 Days:

In the below line graph, we can see forecasting graph. Blue line represents the original deaths and light blue line represents the next 60 days predicted deaths and blue colour area represents the Max and Min range of deaths for next 60 days.



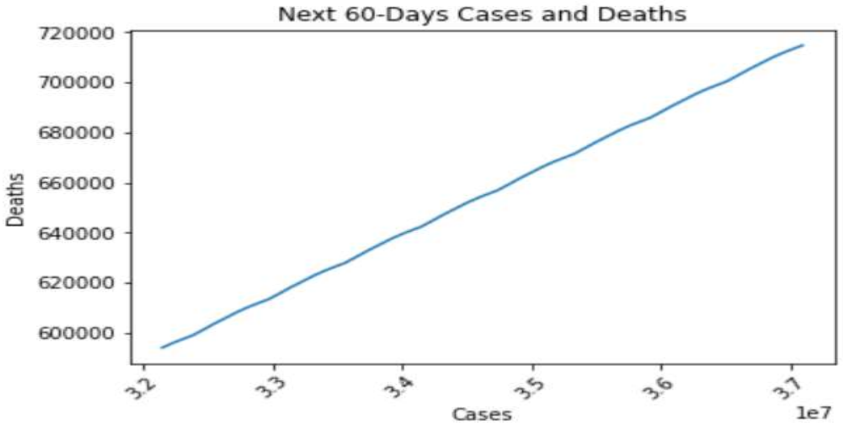
```
deaths_plot=model_deaths.plot(forecast_deaths)
```

Below line graphs represents the deaths monthly wise and day wise. When we observe day-wise graph we can observe that deaths number got increased gradually by end of the week and deaths number got decreased at starting of week**.**

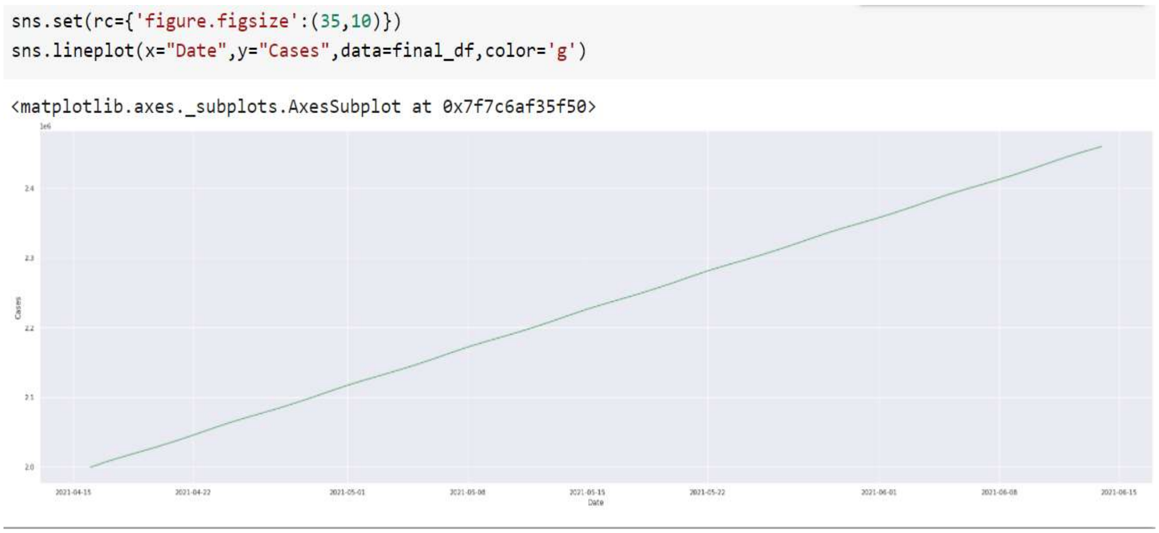


```
death_forecast=model.plot_components(forecast_deaths)
```

Below line graph represents the Next 60 days cases and deaths. There is no fluctuations by this we can say there is chance of increasing cases when days passes.



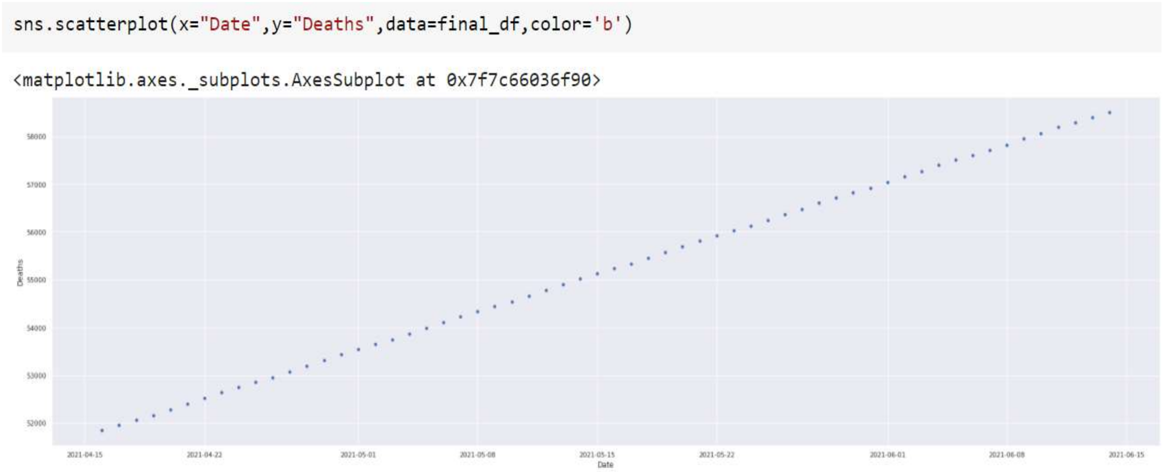
Cases & Deaths Prediction State-Wide:



```
sns.set(rc={'figure.figsize':(35,10)})
sns.lineplot(x="Date",y="Cases",data=final_df,color='g')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f7c6af35f50>
```

Above line graph represents next 60 days predicted cases for New-York. We can predict cases for any states in my project by just giving states name as input.



```
sns.scatterplot(x="Date",y="Deaths",data=final_df,color='b')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f7c66036f90>
```

Above line graph represents next 60 days predicted deaths for New-York state. We can predict cases for any states in my project by just giving state name as input.

12. Conclusion:

Assuming all data models used could provide accurate results and taking the death rate under consideration I assume that this Prophet modelling approach will provide us with reliable and accurate forecast for the next 60 days.

13. Bibliography:

[1] Iwendi, C. (2020). COVID-19 Patient Health Prediction Using Boosted Random Forest Algorithm. Frontiers. https://www.frontiersin.org/articles/10.3389/fpubh.2020.00357/full

[2] Liu, M., Thomadsen, R., & Yao, S. (2020, November 23). Forecasting the spread of COVID-19 under different reopening strategies. Scientific Reports. https://www.nature.com/articles/s41598-020-77292-8?error=cookies_not_supported&code=4dbab9bd-822a-46ce-a05f-b5c7f4a2ecf4.

[3] COVID-19 Hospitalization Tracking Project. (n.d.). Carlson School of Management. https://carlsonschool.umn.edu/mili-misrc-covid19-tracking-project

[4] Khakharia, A., Shah, V., Jain, S., Shah, J., Tiwari, A., Daphal, P., Warang, M., & Mehendale, N. (2020, October 16). Outbreak Prediction of COVID-19 for Dense and Populated Countries Using Machine Learning. Annals of Data Science. https://link.springer.com/article/10.1007/s40745-020-00314-9?error=cookies_not_supported&code=2ef5c432-75d2-4876-ba24-9883616ce316.

[5] Machine learning models for covid-19 future forecasting. (n.d.). PubMed Central (PMC). https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7723767.