Outbreak prediction of COVID-19 in India using Machine Learning

Preethi Paripally, Vallamkondu Sai Vaishnavi, Rishitha Moturi, Kolasani Aarti Chowdary
Guidance: Dr. Sandhya N

Abstract:

The unexpected pervading of severe acute respiratory syndrome COVID-19 has been leading into an eminent crisis. It has affected various fields in the economy, for say, public transportation, agricultural, IT industry, manufacturing. Because of the highly complicated nature of the COVID-19 spread and variation in its spread from nation-to-nation, Machine Learning (ML) is being suggested as an successful tool to model the outbreak. To know the severity of the pandemic in India, we used some ML algorithms and predicted the future scenario of the nation by enumerating the confirmed cases, deaths and recoveries. Various prediction models are made by using ML algorithms and their results are computed, compared and evaluated. We collected case data from Jan 1st, 2020 to Jun 28th, 2020 for India to evaluate the predicted number of COVID-19 cases. We compared the performance of 6 ML models including Linear Regression, Logistic Regression, SVM, Decision Tree, Random Forest, Prophet. Prophet, SVM and Linear Regression had the comparatively negligible prediction error rates for tracking the dynamics of incidents cases in India. The major take-away of this work include accurate analysis of confirmed cases, recovered cases, deaths, prediction of pandemic outbreak for next 20 days. Needless to mention, there are many factors that lead to increase in number of cases in the forthcoming days. It is high time that people should take up responsibility and strive towards the decrease of the virus outbreak by following government norms and rules. It is people of the country and how responsibly they behave, will bring back the good old days. People in the country may decrease their chances of being infected or spreading COVID-19 by following some simple precautionary measures that are available on the WHO website and staying aware regarding the latest news on the COVID-19 pandemic.

Key Words:

COVID-19; Model; Prediction; Machine Learning; Logistic Regression; SVM; Prophet; Linear Regression; Random Forest; Decision Tree; Forecasting

1. Introduction:

The harsh outbreak of Severe Acute Respiratory Syndrome - Coronavirus (SARS-CoV-2) also known as COVID-2019 has brought the worldwide danger to the living body. The pandemic has been spreading rapidly everywhere the planet leading to the heavy economic hardships and immense loss to mankind. the primary pandemic attack of SARSCoV-2 was reported in Wuhan (sprawling capital of south china) on 17th November 2019. The diagnosed opening case was on 8th December 2019 and specialists didn't openly admit there was human-to-human transmission until 21st January. World Health Organization (WHO) declared on January 30, 2020 the outbreak as an emergency and pandemic for public health. COVID-19's clinical symptoms are respiratory disease, fatigue, dry cough, tiredness, etc. while 80 percent of patients heal with none care. Elder people and other people with pre-existing medical illness (such as cardiovascular disorder, obesity, asthma and diabetes) have more chances of becoming severely ill with the virus. The correct way to cease and drop transmission is maintaining social distancing. We have to guard our self and others from infection by washing our hands or using sanitizers and avoid touching face. The spread of the virus, in spite of being caused by quite a few numbers of known and unknown variables but also is caused by the complex nature of population-wide behavior in different geopolitical areas and variations in containment strategies had strikingly increased model uncertainty to a large extent. Adding to this, getting reliable results are very much challenging for any standard epidemiology model. Many novel models have been emerged to induce this challenging nature by bringing several assumptions to modelling (e.g., adding social distancing in the sort of curfews and quarantines. Forecasting the infection spread can even make things easy with the estimation of medical resources that may be necessary. It may help countries so as to confront to face up the long run and allot their valuable time and money. The goal of this paper is to predict the confirmed infection counts 20 days into the long run. While specializing in the Indian plight, the first case of the 2019–20 coronavirus epidemic in India was accounted on 30th January 2020. Specialists propose the count of the disease might be much higher as India's examination rates are among the most diminished on the planet. In India, as of 27th September 2020, there have been 59,03,932 confirmed cases of COVID-19 with 93,379 deaths. Cumulative and engrossed striving of the Government of India in association with States and Union territories succeeded in raising the recovered cases to 49,41,627 by 27th September 2020.

Machine learning assisted decision making

While numerous healthcare associations have actualized Machine Learning (ML) tools at the aim of care, few have effectively applied them to significant higher cognitive process. Machine learning gives more effectiveness and energy in healthcare, organizations still need a collaborative approach, clear knowledge of information processes and a fit leadership to impact genuine change. This can be useful for the Government and other healthcare organization for decision making, function planning, goal setting and prediction.

2. Case Study: Analyzing the Outbreak of COVID 19 using Machine Learning

a. Importing Libraries

The necessary libraries are imported in order to process data, analyze and visualize data. Mainly used library to plot graphs is plotly.

b. Data Collection

The dataset is collected from the Kaggle regarding the expansion of Covid-19 from the date of 1-Jan-2020 till 28-june-2020 globally (day-wise).

	Province/State	Country/Region	Lat	Long	Date	Confirmed	Deaths	Recovered	Active	WHO Region
0	NaN	Afghanistan	33.939110	67.709953	1/22/2020	0	0	0	0	Eastern Mediterranean
1	NaN	Albania	41.153300	20.168300	1/22/2020	0	0	0	0	Europe
2	NaN	Algeria	28.033900	1.659600	1/22/2020	0	0	0	0	Africa
3	NaN	Andorra	42.506300	1.521800	1/22/2020	0	0	0	0	Europe
4	NaN	Angola	-11.202700	17.873900	1/22/2020	0	0	0	0	Africa
...	...	...	...	...	...	...	...	...	...	...
41494	NaN	Sao Tome and Principe	0.186400	6.613100	6/28/2020	713	13	219	481	Africa
41495	NaN	Yemen	15.552727	48.516388	6/28/2020	1118	302	430	386	Eastern Mediterranean
41496	NaN	Comoros	-11.645500	43.333300	6/28/2020	272	7	161	104	Africa
41497	NaN	Tajikistan	38.861000	71.276100	6/28/2020	5849	52	4448	1349	Europe
41498	NaN	Lesotho	-29.610000	28.233600	6/28/2020	27	0	4	23	Africa

41499 rows × 10 columns

c. Data Preprocessing

Data Cleansing: We transformed the dataset in the following way

1. We firstly narrowed down the data from global scale to India as our subject interest is on pandemic outbreak in India.
2. To analyze the outbreak, we had to use the date wise data and so we changed to yyyy-mm-dd
3. We removed the other columns other than Country/Region, Date, Confirmed, Deaths, Active
4. We changed the columns Country/Region to Country.

	Country	Date	Confirmed	Deaths	Recovered	Active
0	India	2020-01-01	0	0	0	0
1	India	2020-01-02	0	0	0	0
2	India	2020-01-03	0	0	0	0
3	India	2020-01-04	0	0	0	0
4	India	2020-01-05	0	0	0	0
...	...	...	...	...	...	...
175	India	2020-06-24	473105	14894	271697	186514
176	India	2020-06-25	490401	15301	285637	189463
177	India	2020-06-26	508953	15685	295881	197387
178	India	2020-06-27	528859	16095	309713	203051
179	India	2020-06-28	548318	16475	321723	210120

180 rows × 6 columns

The dataset is divided into three parts each consisting the number of confirmed, recovered and deaths.

```
confirmed = df.groupby('Date').sum()['Confirmed'].reset_index()
deaths = df.groupby('Date').sum()['Deaths'].reset_index()
recovered = df.groupby('Date').sum()['Recovered'].reset_index()
```

Confirmed:

	Date	Confirmed
0	2020-01-01	0
1	2020-01-02	0
2	2020-01-03	0
3	2020-01-04	0
4	2020-01-05	0
...	...	...
174	2020-06-24	473105
175	2020-06-25	490401
176	2020-06-26	508953
177	2020-06-27	528859
178	2020-06-28	548318

179 rows × 2 columns

Recovered:

	Date	Recovered
0	2020-01-01	0
1	2020-01-02	0
2	2020-01-03	0
3	2020-01-04	0
4	2020-01-05	0
...	...	...
174	2020-06-24	271697
175	2020-06-25	285637
176	2020-06-26	295881
177	2020-06-27	309713
178	2020-06-28	321723

179 rows × 2 columns

Deaths:

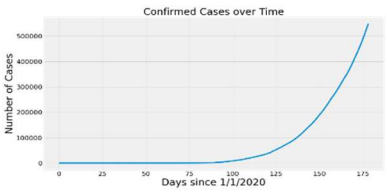
	Date	Deaths
0	2020-01-01	0
1	2020-01-02	0
2	2020-01-03	0
3	2020-01-04	0
4	2020-01-05	0
...	...	...
174	2020-06-24	14894
175	2020-06-25	15301
176	2020-06-26	15685
177	2020-06-27	16095
178	2020-06-28	16475

179 rows × 2 columns

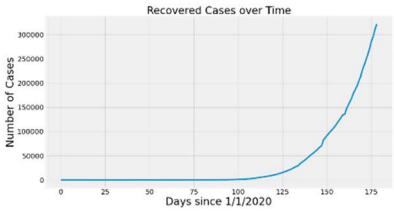
d. Data Visualization

We visualized the data by taking the values of the number of confirmed, recovered and deaths on Y-axis; count of days from 1st Jan to 28th June

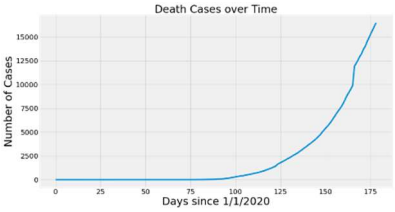
Confirmed:



Recovered:



Deaths:



e. Data Splitting

The model splits the dataset into train and test data in the ratio of 4:1 i.e., 80% as trained data and 20% as test data.

f. Modelling

Prediction is a typical data science exercise that helps the administration with function planning, objective setting, and anomaly detection.

Root mean absolute error of Machine Learning Algorithms

Machine Learning Model	Confirmed	Recovered	Deaths
SVM	327.764455033243	320.1904442973551	46.68268522744836
Random Forest	34.51638097419188	31.832068037681047	8.185522449897613
Decision Tree	61.154176744792615	39.40353904015335	8.237785570838264
Logistic Regression	149.5009290197816	78.12258885168161	10.92652226872251
Linear Regression	289.3116283104797	267.72517997069355	48.205087402502244

We have computed the values of the Covid-19 cases for next 20 days by using different algorithms:

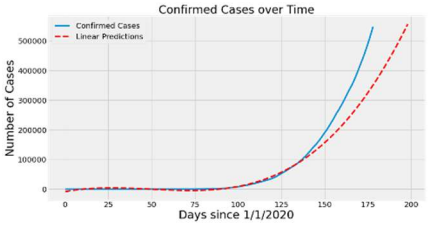
i. Linear Regression

Linear regression is basic model for regression analysis where we find the pattern between dependent variable(y) and independent variable (x). Polynomial Regression is one of the type of linear regression which is user for regression analysis. In polynomial

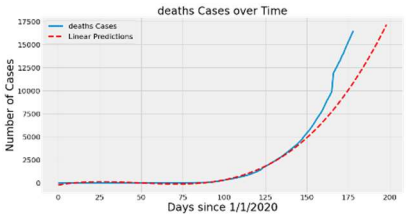
regression the dependency of x over y will be of degree n. The relationship between the value of x and the corresponding mean of y can be denoted as E(y/x) which fits is non-linear. Even though it fits a non-linear model, it is said to be linear according to the statistical estimation problem, here we consider the regression function E(y/x) is linear in the unknown parameters that is estimated from the data. Hence, polynomial regression is considered to be a special type of multiple linear regression. The polynomial equation is given by $y = a + b1x + b2x^2 + .+ bnx^n$

As the value of n increases the number of higher – order terms increase, i.e.; the equation becomes more complex.

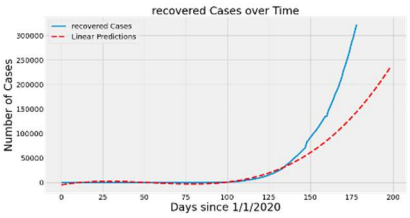
Confirmed:



Deaths:



Recovered:



Prediction for next 20 days:

Confirmed:

	Date	Linear Predicted # number of Confirmed cases in India
0	2020-06-28	359849.0
1	2020-06-29	368832.0
2	2020-06-30	377959.0
3	2020-07-01	387229.0
4	2020-07-02	396645.0
5	2020-07-03	406208.0
6	2020-07-04	415918.0
7	2020-07-05	425777.0
8	2020-07-06	435785.0
9	2020-07-07	445945.0
10	2020-07-08	456257.0
11	2020-07-09	466721.0
12	2020-07-10	477340.0
13	2020-07-11	488114.0
14	2020-07-12	499045.0
15	2020-07-13	510133.0
16	2020-07-14	521380.0
17	2020-07-15	532786.0
18	2020-07-16	544354.0
19	2020-07-17	556083.0

Deaths:

	Date	Linear Predicted # number of Deaths cases in India
0	2020-06-28	11128.0
1	2020-06-29	11403.0
2	2020-06-30	11683.0
3	2020-07-01	11967.0
4	2020-07-02	12256.0
5	2020-07-03	12549.0
6	2020-07-04	12846.0
7	2020-07-05	13148.0
8	2020-07-06	13455.0
9	2020-07-07	13766.0
10	2020-07-08	14081.0
11	2020-07-09	14402.0
12	2020-07-10	14727.0
13	2020-07-11	15057.0
14	2020-07-12	15391.0
15	2020-07-13	15730.0
16	2020-07-14	16075.0
17	2020-07-15	16424.0
18	2020-07-16	16778.0
19	2020-07-17	17136.0

Recovered:

	Date	Recovered Predicted # number of Confirmed cases in India
0	2020-06-28	147991.0
1	2020-06-29	151888.0
2	2020-06-30	155851.0
3	2020-07-01	159878.0
4	2020-07-02	163971.0
5	2020-07-03	168131.0
6	2020-07-04	172357.0
7	2020-07-05	176650.0
8	2020-07-06	181012.0
9	2020-07-07	185441.0
10	2020-07-08	189939.0
11	2020-07-09	194507.0
12	2020-07-10	199144.0
13	2020-07-11	203852.0
14	2020-07-12	208631.0
15	2020-07-13	213481.0
16	2020-07-14	218402.0
17	2020-07-15	223397.0
18	2020-07-16	228464.0
19	2020-07-17	233604.0

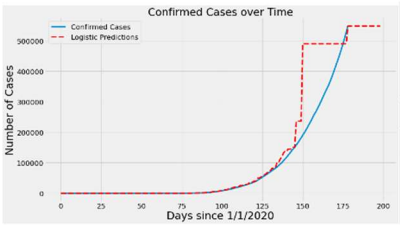
ii. Logistic Regression

The function used at the core of this algorithm is called logistic function, hence the name logistic regression. It is also called as sigmoid function. It was developed for various reasons by the statisticians such as to describe the population growth etc. The curve is S-shaped which takes any real number and map the number to a value between 0 and 1 exclusively.

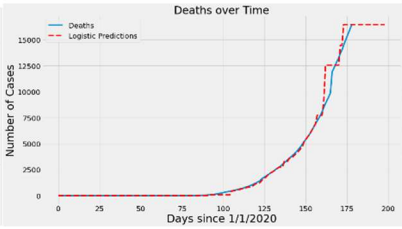
1/(1+e^-value)

Where:
e: e is the base of the natural logarithms (Euler's number or EXP () function in your spreadsheet).
value: It is the actual numerical value that you want to transform.
The representation of the equation which is used by the logistic regression is very similar to that of linear regression. To predict the output value, the input values are combined linearly using coefficient values or weights. The main difference between linear and logistic is that the output value of logistic regression is being taken as a binary value i.e., 0 or 1 where linear regression is modelled to give numerical output value.
Example for logistic regression equation:
Y=e^(b0+b1*x)/(1+e^(b0+b1*x))
Where:
Y: Predicted output
b0: bias or intercept term
b1: coefficient for the single input value(x)
Each column in input data has an associated b coefficient (a constant real value) that must be learned from your training data.

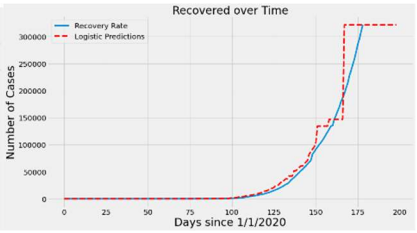
Confirmed:



Deaths:



Recovered:



Prediction for next 20 days:

Confirmed:

	Date	Logistic Predicted # number of Confirmed cases in India
0	2020-06-28	548318
1	2020-06-29	548318
2	2020-06-30	548318
3	2020-07-01	548318
4	2020-07-02	548318
5	2020-07-03	548318
6	2020-07-04	548318
7	2020-07-05	548318
8	2020-07-06	548318
9	2020-07-07	548318
10	2020-07-08	548318
11	2020-07-09	548318
12	2020-07-10	548318
13	2020-07-11	548318
14	2020-07-12	548318
15	2020-07-13	548318
16	2020-07-14	548318
17	2020-07-15	548318
18	2020-07-16	548318
19	2020-07-17	548318

Deaths:

	Date	Logistic Predicted # number of Deaths in India
0	2020-06-28	16475
1	2020-06-29	16475
2	2020-06-30	16475
3	2020-07-01	16475
4	2020-07-02	16475
5	2020-07-03	16475
6	2020-07-04	16475
7	2020-07-05	16475
8	2020-07-06	16475
9	2020-07-07	16475
10	2020-07-08	16475
11	2020-07-09	16475
12	2020-07-10	16475
13	2020-07-11	16475
14	2020-07-12	16475
15	2020-07-13	16475
16	2020-07-14	16475
17	2020-07-15	16475
18	2020-07-16	16475
19	2020-07-17	16475

Recovered:

	Date	Logistic Predicted # number of Recovered cases in India
0	2020-06-28	321723
1	2020-06-29	321723
2	2020-06-30	321723
3	2020-07-01	321723
4	2020-07-02	321723
5	2020-07-03	321723
6	2020-07-04	321723
7	2020-07-05	321723
8	2020-07-06	321723
9	2020-07-07	321723
10	2020-07-08	321723
11	2020-07-09	321723
12	2020-07-10	321723
13	2020-07-11	321723
14	2020-07-12	321723
15	2020-07-13	321723
16	2020-07-14	321723
17	2020-07-15	321723
18	2020-07-16	321723
19	2020-07-17	321723

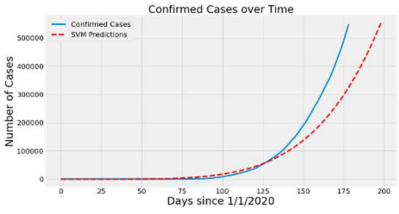
iii. SVM

Support-Vector Machines in a Machine Learning Algorithm. It also supports Support-Vector Networks. It is of two types, supervised and unsupervised learning models. Supervised learning algorithms analyzes the data for both classification and regression analysis. SVM regression is a non- parametric technique since it depends on the set of mathematical functions. These set of mathematical functions is called kernel. The data inputs are transformed into the desired form by the kernel. The regression problems by SVM are solved using linear function. It maps the input vector (x) to the n-dimensional space which is also called feature space (z), while solving non-linear regression problems. The mapping is done using mapping techniques of non-linear and then linear regression is applied to the space. As per the ML context, the concept of with a multivariate training dataset taken as xn means N number of observations with yn as the corresponding observed responses. The linear function can be written as: $f(x)=x'\beta+b$

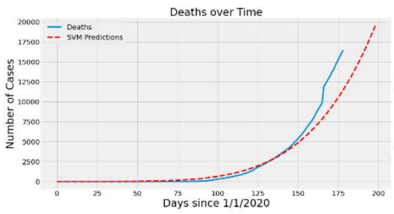
The main objective is to make the curve as flat as possible in order to find the value of f(x) and (β′β) as minimal norm values. Hence, the minimization function which fits the problem as: $J(\beta)=(½)\beta'\beta$

And with an extra condition of the values of all residuals is less than or equal to ε, as shown in below equation: $\forall n:|yn-(x'n\beta+b)|\leq\varepsilon$

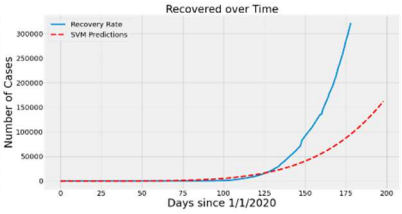
Confirmed:



Deaths:



Recovered:



Prediction for next 20 days:

Confirmed:

	Date	SVM Predicted # number of Confirmed cases in India
0	2020-06-28	333405.0
1	2020-06-29	342849.0
2	2020-06-30	352505.0
3	2020-07-01	362377.0
4	2020-07-02	372469.0
5	2020-07-03	382783.0
6	2020-07-04	393324.0
7	2020-07-05	404096.0
8	2020-07-06	415101.0
9	2020-07-07	426345.0
10	2020-07-08	437830.0
11	2020-07-09	449561.0
12	2020-07-10	461542.0
13	2020-07-11	473776.0
14	2020-07-12	486268.0
15	2020-07-13	499021.0
16	2020-07-14	512040.0
17	2020-07-15	525329.0
18	2020-07-16	538892.0
19	2020-07-17	552732.0

Deaths:

	Date	SVM Predicted # number of Deaths in India
0	2020-06-28	11797.0
1	2020-06-29	12130.0
2	2020-06-30	12471.0
3	2020-07-01	12819.0
4	2020-07-02	13175.0
5	2020-07-03	13539.0
6	2020-07-04	13911.0
7	2020-07-05	14291.0
8	2020-07-06	14680.0
9	2020-07-07	15076.0
10	2020-07-08	15482.0
11	2020-07-09	15896.0
12	2020-07-10	16318.0
13	2020-07-11	16750.0
14	2020-07-12	17191.0
15	2020-07-13	17641.0
16	2020-07-14	18100.0
17	2020-07-15	18569.0
18	2020-07-16	19048.0
19	2020-07-17	19536.0

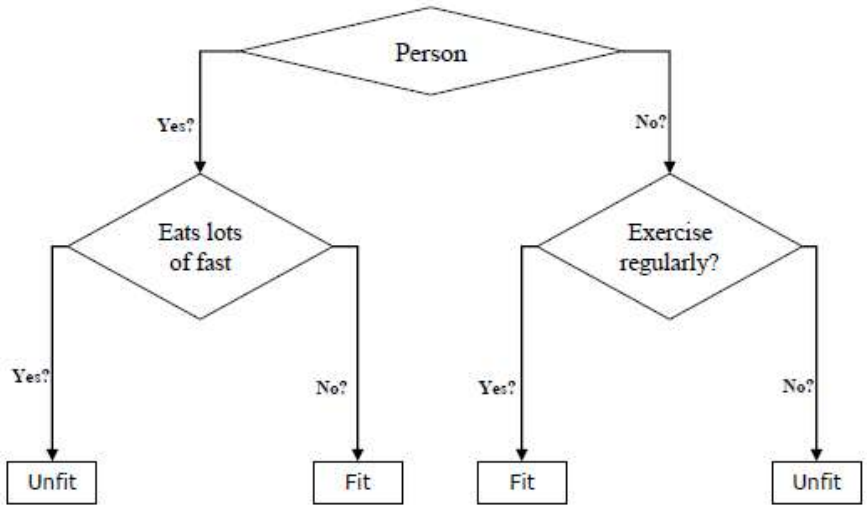
Recovered:

	Date	SVM Predicted # number of Recovered cases in India
0	2020-06-28	97932.0
1	2020-06-29	100706.0
2	2020-06-30	103543.0
3	2020-07-01	106443.0
4	2020-07-02	109407.0
5	2020-07-03	112437.0
6	2020-07-04	115534.0
7	2020-07-05	118698.0
8	2020-07-06	121931.0
9	2020-07-07	125233.0
10	2020-07-08	128607.0
11	2020-07-09	132053.0
12	2020-07-10	135573.0
13	2020-07-11	139166.0
14	2020-07-12	142836.0
15	2020-07-13	146582.0
16	2020-07-14	150407.0
17	2020-07-15	154310.0
18	2020-07-16	158294.0
19	2020-07-17	162360.0

iv. Decision Tree

Decision tree is one of the predictive modeling tools. Decision tree analysis is useful and applied in many areas. Decision tree is a supervised learning algorithm. It is one of the most powerful algorithms. In this model, we construct decision trees in an algorithmic approach. Based on different conditions, it splits the dataset into parts in different ways.

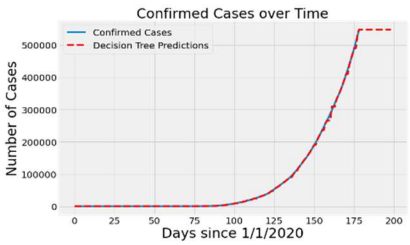
Decision trees can be used for two types of prediction models, i.e., it can be performed on both classification and regression models. There are two main entities for a decision tree. They are decision nodes and leaves. The data is split based on the decision nodes and leaves contains the outcomes. Here is an example of a decision tree(binary tree). It predicts whether the person unfit or fit considering different aspects like the person's age, his eating habits and his exercise habits.



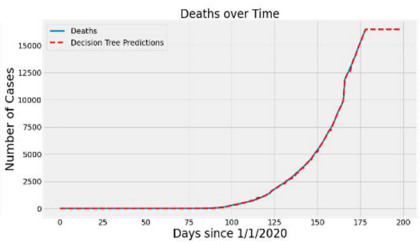
As discussed above, the decision nodes contain the questions and the leaves contains the outcomes in the given decision tree example.
Two types of decision trees are:

- Classification decision trees – The decision variable is categorical in this type of decision trees. The given example is a type of classification decision tree.
- Regression decision trees – Here the decision variable is continuous in this type of decision trees.

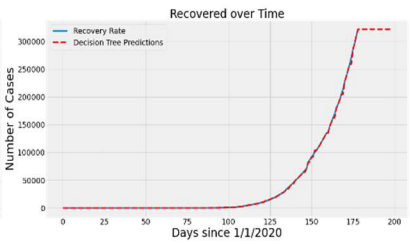
Confirmed:



Deaths:



Recovered:



Prediction for next 20 days:

Confirmed:

	Date	Decision Tree Predicted # number of Confirmed cases in India
0	2020-06-28	548318
1	2020-06-29	548318
2	2020-06-30	548318
3	2020-07-01	548318
4	2020-07-02	548318
5	2020-07-03	548318
6	2020-07-04	548318
7	2020-07-05	548318
8	2020-07-06	548318
9	2020-07-07	548318
10	2020-07-08	548318
11	2020-07-09	548318
12	2020-07-10	548318
13	2020-07-11	548318
14	2020-07-12	548318
15	2020-07-13	548318
16	2020-07-14	548318
17	2020-07-15	548318
18	2020-07-16	548318
19	2020-07-17	548318

Deaths:

	Date	Decision Tree Predicted # number of Deaths in India
0	2020-06-28	16475
1	2020-06-29	16475
2	2020-06-30	16475
3	2020-07-01	16475
4	2020-07-02	16475
5	2020-07-03	16475
6	2020-07-04	16475
7	2020-07-05	16475
8	2020-07-06	16475
9	2020-07-07	16475
10	2020-07-08	16475
11	2020-07-09	16475
12	2020-07-10	16475
13	2020-07-11	16475
14	2020-07-12	16475
15	2020-07-13	16475
16	2020-07-14	16475
17	2020-07-15	16475
18	2020-07-16	16475
19	2020-07-17	16475

Recovered:

	Date	Decision Tree Predicted # number of Recovered cases in India
0	2020-06-28	321723
1	2020-06-29	321723
2	2020-06-30	321723
3	2020-07-01	321723
4	2020-07-02	321723
5	2020-07-03	321723
6	2020-07-04	321723
7	2020-07-05	321723
8	2020-07-06	321723
9	2020-07-07	321723
10	2020-07-08	321723
11	2020-07-09	321723
12	2020-07-10	321723
13	2020-07-11	321723
14	2020-07-12	321723
15	2020-07-13	321723
16	2020-07-14	321723
17	2020-07-15	321723
18	2020-07-16	321723
19	2020-07-17	321723

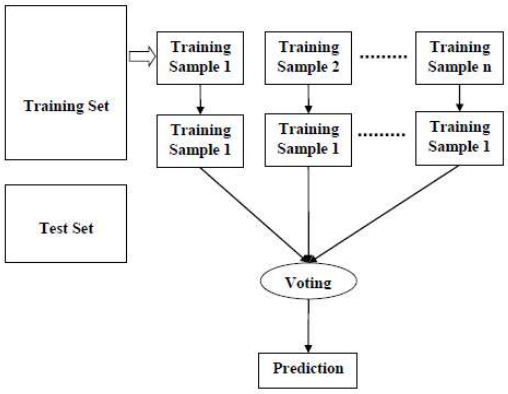
v. Random Forest

Random forest is also one of the supervised learning algorithms. It can be used for both regression and classification models. But it is mostly used for classification problems. Random forest is an extension of decision trees. As the name says, it is a forest and we know that the forest is made up of trees. The more is the number of trees; the more is the robustness of the forest. In the similar manner, decision trees are created on data samples in the random forest algorithm and the prediction is done on each of the decision tree and finally it selects the best accurate solution by some means like voting. As this algorithm uses multiple decision trees rather than one tree, this is the better algorithm than decision tree, this is an ensemble method and the over-fitting is reduced as the result is taken on an average.

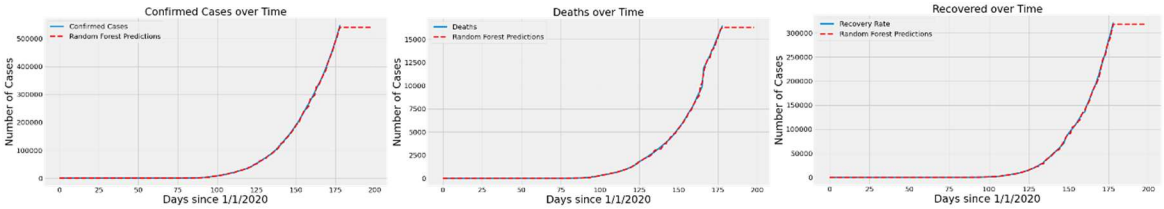
Steps to followed while working on Random Forest Algorithm:

- Step 1 – Select random samples from the given dataset.
- Step 2 - Next, a decision tree is constructed by this algorithm for each sample and the prediction result taken from each and every decision tree.
- Step 3 – Next, for every result, voting will be performed.
- Step 4 – Finally, the prediction result which is voted most is selected as the final prediction result.

The working of this algorithm is illustrated by the following diagram-



Confirmed: Deaths: Recovered:



Prediction for next 20 days:

Confirmed: Deaths: Recovered:

	Date	Naive Bayes Predicted # number of Confirmed cases in India
0	2020-06-28	540580.0
1	2020-06-29	540580.0
2	2020-06-30	540580.0
3	2020-07-01	540580.0
4	2020-07-02	540580.0
5	2020-07-03	540580.0
6	2020-07-04	540580.0
7	2020-07-05	540580.0
8	2020-07-06	540580.0
9	2020-07-07	540580.0
10	2020-07-08	540580.0
11	2020-07-09	540580.0
12	2020-07-10	540580.0
13	2020-07-11	540580.0
14	2020-07-12	540580.0
15	2020-07-13	540580.0
16	2020-07-14	540580.0
17	2020-07-15	540580.0
18	2020-07-16	540580.0
19	2020-07-17	540580.0

	Date	Random Forest Predicted # number of Deaths in India
0	2020-06-28	16282.0
1	2020-06-29	16282.0
2	2020-06-30	16282.0
3	2020-07-01	16282.0
4	2020-07-02	16282.0
5	2020-07-03	16282.0
6	2020-07-04	16282.0
7	2020-07-05	16282.0
8	2020-07-06	16282.0
9	2020-07-07	16282.0
10	2020-07-08	16282.0
11	2020-07-09	16282.0
12	2020-07-10	16282.0
13	2020-07-11	16282.0
14	2020-07-12	16282.0
15	2020-07-13	16282.0
16	2020-07-14	16282.0
17	2020-07-15	16282.0
18	2020-07-16	16282.0
19	2020-07-17	16282.0

	Date	Random Forest Predicted # number of Recovered cases in India
0	2020-06-28	318114.0
1	2020-06-29	318114.0
2	2020-06-30	318114.0
3	2020-07-01	318114.0
4	2020-07-02	318114.0
5	2020-07-03	318114.0
6	2020-07-04	318114.0
7	2020-07-05	318114.0
8	2020-07-06	318114.0
9	2020-07-07	318114.0
10	2020-07-08	318114.0
11	2020-07-09	318114.0
12	2020-07-10	318114.0
13	2020-07-11	318114.0
14	2020-07-12	318114.0
15	2020-07-13	318114.0
16	2020-07-14	318114.0
17	2020-07-15	318114.0
18	2020-07-16	318114.0
19	2020-07-17	318114.0

vi. Prophet

Prophet is an inbuilt algorithm which follows sklearn model API and can be obtained from fbprophet library. It is one of the time-series algorithm which is used to predict the output based on time as the name says. As it an inbuilt Algorithm, it is very much easier to use the Algorithm. All we need to do is create an instance of the class and use the inbuilt methods to fit and predict data.

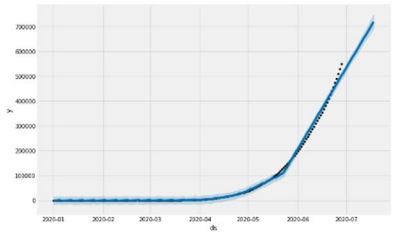
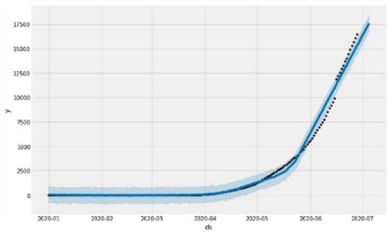
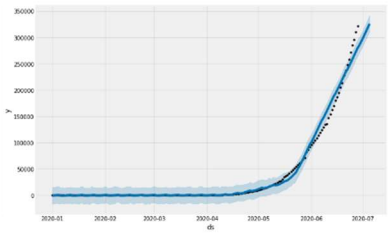
There are three main model components in the decomposable time series model we use, they are:

trend**,** seasonality**,** holidays
They are combined as the following equation:
$Y(t)=g(t)+s(t)+h(t)+\varepsilon_t$

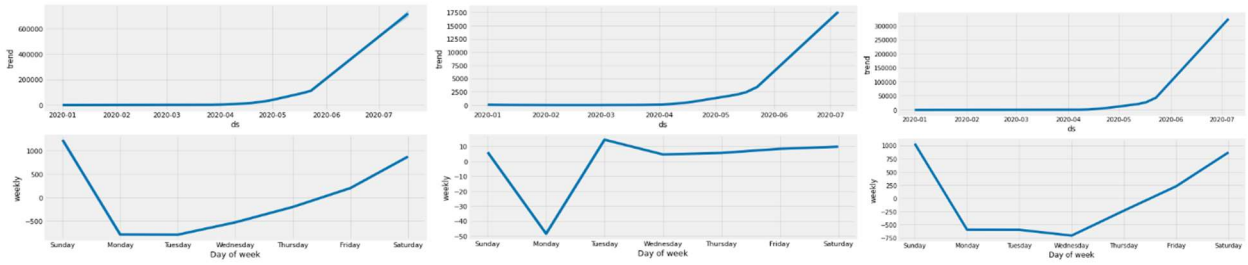
- **g(t)**: piecewise linear or logistic growth curve foe modelling non-periodic changes in the time series.
- **s(t)**: periodic changes (e.g. yearly/weekly seasonality).
- **h(t)**: Effects of holidays with irregular schedules.
- **ε_t**: Error term accounts for any unusual changes not accommodated by the model.

As prophet is a time series algorithm it always tries to fit different functions of linear as well as nonlinear by using time as a regressor. Modeling seasonality as an additive component is the same approach taken by exponential smoothing in "Holt-Winters technique" . We are, in effect, framing the forecasting problem as a curve-fitting exercise rather than looking explicitly at the time-based dependence of each observation within a time series.

Confirmed: Deaths: Recovered:

Weekly analysis:



Prediction for next 5 days:

Confirmed:

	ds	yhat	yhat_lower	yhat_upper
194	2020-07-14	673461.832708	652069.213263	698103.598540
195	2020-07-15	684573.993469	661131.042410	709588.921690
196	2020-07-16	695751.068073	670349.359841	723497.339096
197	2020-07-17	707002.012517	677988.591898	734710.751460
198	2020-07-18	718521.641870	691029.467285	747501.111860

Deaths:

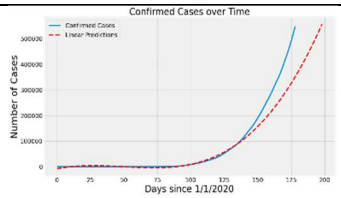
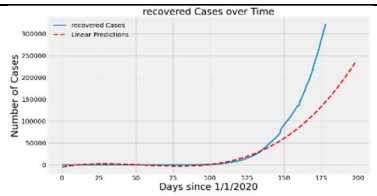
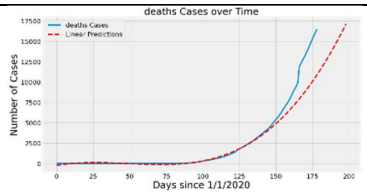
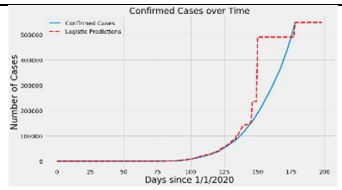
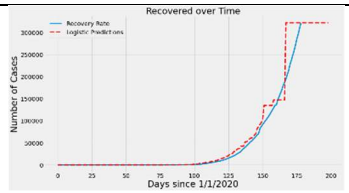
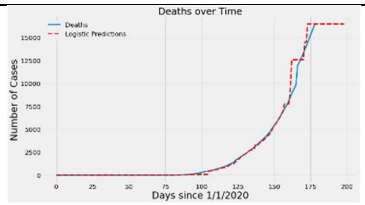
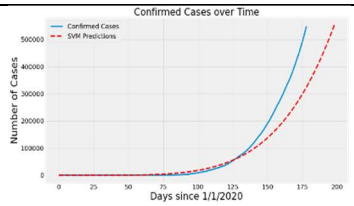
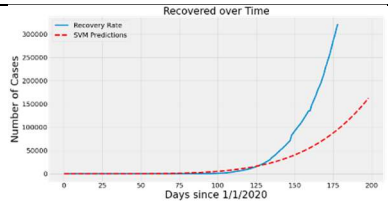
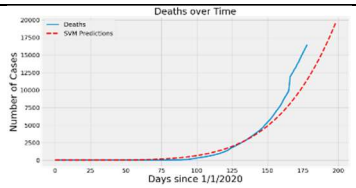
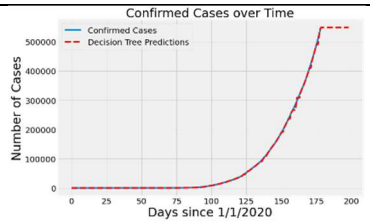
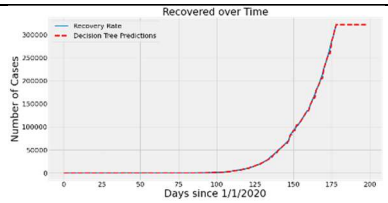
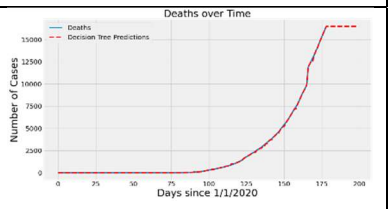
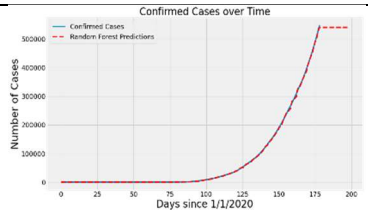
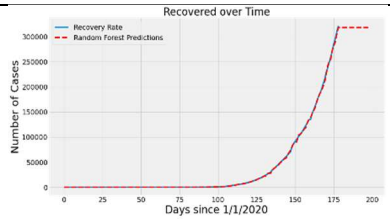
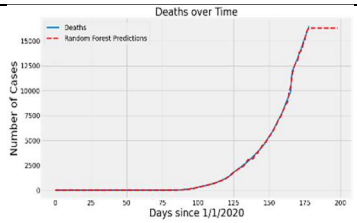
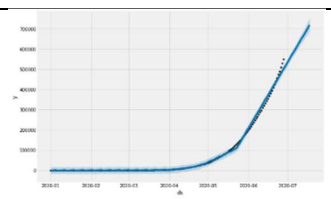
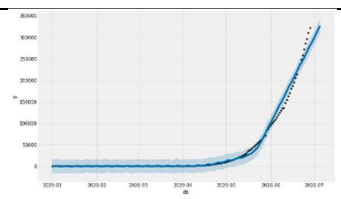
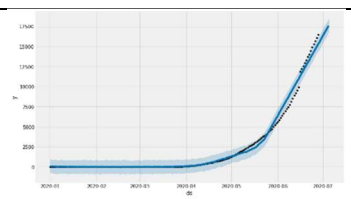
	ds	yhat	yhat_lower	yhat_upper
181	2020-07-01	16269.635781	15430.492112	17069.673059
182	2020-07-02	16600.628079	15790.778174	17474.098577
183	2020-07-03	16933.385971	16070.487452	17802.597846
184	2020-07-04	17264.791639	16433.338403	18055.174366
185	2020-07-05	17591.092015	16821.428042	18392.930518

Recovered:

	ds	yhat	yhat_lower	yhat_upper
181	2020-07-01	297862.781891	282642.777331	312448.441466
182	2020-07-02	304886.653168	290134.116983	321065.453923
183	2020-07-03	311899.997613	296387.113945	327052.498722
184	2020-07-04	319093.747107	304760.499680	335151.767951
185	2020-07-05	325810.365710	310516.145468	341669.063152

3. Result and Discussion:

We drew the following conclusions from our algorithms that Prophet, SVM, Linear Regression provided us better accuracy when compared to the remaining.

Name of Algorithm	Graph of Prediction for Confirmed Cases	Graph of Prediction for Recovered Cases	Graph of Prediction for Death Cases
Linear Regression			
Logistic Regression			
SVM			
Decision Tree			
Random Forest			
Prophet			

Here, Prophet, SVM and Linear Regression has given us the more precise predicted values compared to the others as their curve is not flattened all of a sudden and we can also see that their graphs are mostly matched with actual values.

4. Conclusion:

The study on the prediction of Covid-19 pandemic infection using machine learning reveals the comparative discussion on the confirmed cases, recovered cases, and deaths in India. While we go through the epidemic status, the lack of proper social distancing and personal hygiene playing an important role in leading to the community widespread. The effective management using symptomatic therapy and quarantine system can control the spread of the disease state up to a limit. Although, the condition is getting worse may question the human existence inside the earth. Hence, undoubtedly, we can fetch out in an assumption that SARS- Covid-19 has been tremendously defeating over larger developed country in the world.

5. References:

1. https://www.researchgate.net/publication/340849271_COVID-19_Outbreak_Prediction_with_Machine_Learning
2. https://www.researchgate.net/publication/341778862_Analysis_Prediction_and_Evaluation_of_COVID-19_Datasets_using_Machine_Learning_Algorithms
3. https://www.researchgate.net/publication/341265011_Prediction_and_Spread_Visualization_of_Covid-19_Pandemic_Using_Machine_Learning
4. https://www.researchgate.net/publication/342699785_Forecasting_COVID-19_cases_using_Machine_Learning_models
5. https://www.researchgate.net/publication/342495716_Covid-19_Pandemic_Data_Analysis_and_Forecasting_using_Machine_Learning_Algorithms
6. https://www.who.int/
7. https://mhrd.gov.in/
8. https://pib.gov.in/PressReleasePage.aspx?PRID=1636605
9. https://www.who.int/docs/default-source/wrindia/situation-report/india-situation-report-25.pdf?sfvrsn=8269893f_2
10. https://www.edureka.co/blog/covid-19-outbreak-prediction-using-machine-learning/
11. https://en.wikipedia.org/wiki/Coronavirus_disease_2019#cite_note-autogenerated1-13
12. https://www.medrxiv.org/content/10.1101/2020.04.08.20057679v2
13. https://www.altexsoft.com/blog/datascience/machine-learning-project-structure-stages-roles-and-tools/
14. https://www.analyticsvidhya.com/blog/2018/05/generate-accurate-forecasts-facebook-prophet-python-r/
15. https://www.tutorialspoint.com/machine_learning_with_python/machine_learning_with_python_regression_algorithms_linear_regression.htm
16. https://en.wikipedia.org/wiki/Polynomial_regression