# Analysis of Covid19 in India


Bhargavi G
*Computer Science*
*PES University*
Bangalore, India
bhargu2000@gmail.com

Nisha S
*Computer Science*
*PES University*
Bangalore, India
nishasuresh2001@gmail.com

Trisha C Shekhar
*Computer Science*
*PES University*
Bangalore, India
trisha.c.shekar5@gmail.com

Nikhil D.B
*Computer Science*
*PES University*
Bangalore, India
ndb4263@gmail.com



***Abstract*—Corona virus which originated in a small local seafood market in Wuhan has taken over the whole world. This pandemic has impacted lives in unprecedented ways. In mere nine months Corona virus has spread to over 213 countries and territories. People have lost their jobs and unemployment rates have been on the increase. In this paper we have analysed the covid19 situation in India. We have also predicted the trend in the rising cases. A number of models have been proposed for predicting the number covid19 cases.**


## I. Introduction

COVID-19 Coronavirus disease 2019, is currently a major worldwide threat. It has infected more than a million people globally leading to hundred-thousands of deaths.In that reference, it is extremely crucial to construct models that are realistic and competent to make prediction that can help medical personals ,policy makers , government, general public to take precautionary measures . The results from statistical predictive models can be used to predict and control the threat caused by COVID-19 .Even as this summary is written the number of new cases are increasing around the world. We thought covid19 analysis would be the best topic to work on as we ourselves can get to know its impact on Indian economy and its people. We decided to restrict ourselves to analyse the spread of coronavirus in India only.

A number of papers related to the analysis of coronavirus were read and the following papers were considered.

### *A. Tracking the Spread of COVID-19 Cases in India using Data Visualizing and Forecasting Techniques*

Visualization:

Python libraries like matplotlib and plotly are used for data visualisation. Bar charts, gantt charts among others are used to represent different features of the covid19 dataset. A number of visualisations have been shown in the paper. Some of the visualisations mentioned are, a bar chart to represent the number of active cases in each state, a horizontal bar graph to show the position of India in the world in terms of the number of cases recorded and a chart to represent the number of active, recovered and deceased people in each state. Analysis is not restricted to Indian states.

Visualisations in this paper are based on data collected till the end of April but we are using data collected till the end of September.


Identify applicable funding agency here. If none, delete this.


Figure 1 is one of the data visualizations from the paper. It is a bar chart showing the top 15 countries along with the number of covid cases reported in those countries.

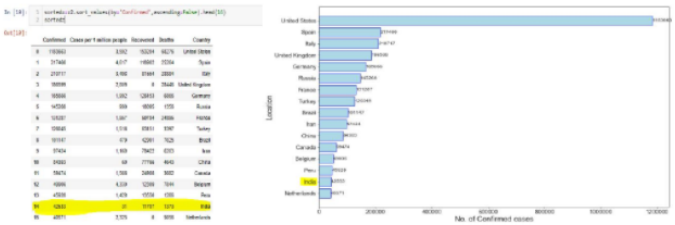


Fig.5 India has 15$^{th}$ highest Confirmed covid cases globally having greater risk of being next hotspot as per March 3rd 2020 reported data

Forecasting:

The ARIMA model is used for forecasting. In the ARIMA model Dickey-fuller test is used to test the stationarity of time series. Autocorrelation function and partial autocorrelation functions are used to determine parameters for this ARIMA model. The parameters found are then passed to the ARIMA model for forecasting the actual time series for confirmed covid cases across dates in India. According to the values predicted using the ARIMA model it is observed that the number of cases increases exponentially over time.

Holt winter model is also used for forecasting. It is simpler in terms of implementation when compared to ARIMA and has almost the same accuracy as ARIMA. This model also predicted that the number of cases would increase over time.

### *B. Time Series Prediction Based on Linear Regression and SVR*

This paper uses linear regression and support vector regression for time series analysis. They are combined and not used separately because the combined model gives better efficiency than when individual models are used. Time series is divided into linear and non-linear parts. This linear part is the stable part of the time series and the non-linear part is the unstable part of time series which contains some irregular variation. First-order linear model as a stable part. The two parts, linear and non-linear of this new model are predicted separately and then integrated together for forecasting. The new regression implemented was better than common SVR 40% in precision of forecasting.

Pre-processing of time series is done by splitting the time series by first order linear regression. Pre-processing of time series helps in forecasting the two parts mentioned

earlier. To do first order linear regression for a time series x t where t=1,2,...,N the following formulas were used.

$$b = \frac{\sum x_i t - n\bar{x}\bar{t}}{\sum t^2 - n\bar{t}^2}$$

$$a = \bar{x} - b\bar{t}$$

Where, $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ , $\bar{t} = \frac{1}{n}\sum_{i=1}^{n} t$

Once a and b are found the linear part of the time series x' is set to a+bt and the nonlinear part x" is set to x-x'. Pre-processing of time series is followed by phase space reconstruction and linear and SVR forecasting. The model is finalised by selecting those values for the parameters which give better accuracy. The following table is the result of this paper which shows that the precision result of the new model is better than just SVR.

Table 1 prediction data between two manners

| | | | |
|---|---|---|---|
| Source data | 41.5652 | 34.179 | 48.0958 |
| SVR data | 45.0366 | 36.2981 | 36.8933 |
| New model data | 45.4356 | 34.1775 | 35.7289 |
| Source data | 46.3303 | 38.39649 | 39.6433 |
| SVR data | 37.1143 | 35.831 | 36.5036 |
| New model data | 44.5543 | 41.7416 | 37.7681 |

*C. Coronavirus (COVID-19): ARIMA based time-series analysis to forecast near future*

Corona virus (COVID -19) is an ongoing pandemic that has infected more than 39.7 million people, taking away life's of more than 1.11 million people. In that reference it's important to construct competent and realistic models which help us to analyse present situation, forecast future number of cases, assist to take precautionary measures and control the pandemic. In this paper they have employed Auto-Regressive Integrated Moving Average (ARIMA) model for prediction.

Data: The data used for this model is Confirmed ,recovered and death cases caused by corona virus in India and south-east Asian countries from January 2020 to 13 April 2020.

Assumptions: In this paper they have followed normality assumption with residual histogram backs up the assumption. The model satisfies assumption of constant variance.Here, the independent assumption is not violated cause we can observe the p-value is larger than the significance level for all lags.

Tech: ARIMA models and parameters is expressed as in terms of p, d, q where p- order of auto-regression, d- degree of trend difference , q - the order of moving average. To find the initial number they use Autocorrelation function (ACF) graph and partial autocorrelation (PACF) graph. Then they have tested models for variance in normality and stationary. We look into the values of MAPE,MAD and MSD. The best model is the one whose value for all measures is the lowest. The best fit ARIMA model is compared with Linear Trend, Moving Average, Quadratic Trend, Single Exponential,SCurve Trend, Double Exponential models. Then they have use the model which is built to predict cases in the next 20 days.The representation of the model for prediction confirmed cases in future is represented as, ARIMA(p,d,f):

$$Xt = \alpha 1 Xt{-}1 + \alpha 2 Xt{-}2 + \beta 1 Zt{-}1 + \beta 2 Zt{-}2 + Zt$$

where $Zt = Xt - Xt{-}1$

Xt -forecasted number of confirmed cases at t th day, alpha1, alpha2, beta1 and beta2 are parameters whereas Zt - residual term for t th day.

They have also performed a comparative study to give a reality cheeck of confirmed COVID-19 cases of India with respect to those of highly infected countries and south -east Asian region.

Main claims: The results for measure of model accuracy suggests that ARIMA is best suited for forecast because of its least value for all the measures. The AR (2) and MA (2) estimates p-values imply that parameters are significant in the model. We can observe a slight deviation of residuals from the straight line which indicates that there aren't many errors but we have a few outliers in the dataset. They also claim that there is high rise in cases plotting the time series graph. They check for precision of model in forecasting by plotting actual confirmed cases vs forecasted cases. They claim that the disease has low mortality because the rate of recovery is high. They suggest that is India is one among the most infected countries.

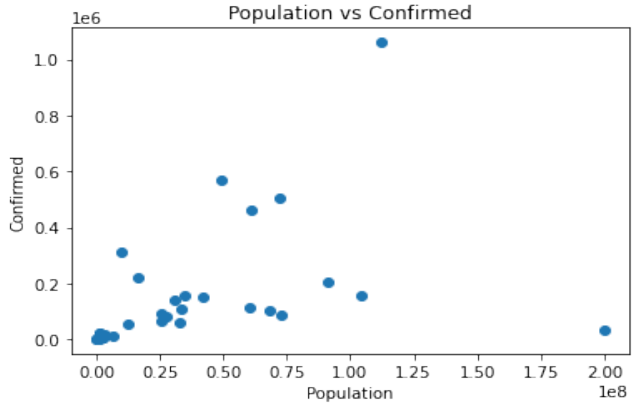
Conclusion: For natural adversities prediction ARIMA models are better when compared to other prediction models like wavelet neural network and support vector machine. We can observe ARIMA model is best suited for Forecasting problems The time series analysis shows an exponential enhancement in the infected cases. A comparative study with some of the highly infected countries and countries in south-east Asia region indicates that India was the last amongst these countries to get infected.

## II. PROPOSED PROBLEM STATEMENT

With this project, we aim at analysing the current covid19 situation in India and predicting the trend in the rising cases.We also take into many other factors that might have relations with the increase in cases in each state. We take into account 7 data sets, perform Exploratory Data Analysis on various problems and answer the key problem statements related to them.

*A. Extract population and state/ union territory, Analyse the percent of active or total covid cases with respect to the population of each state and to check if the data follows a particular trend.*

Logically one might infer that states with larger population have higher number of confirmed cases compared to the others, but on analysis of the relationship between total Population and covid cases, we got a Pearson's correlation of 0.445.



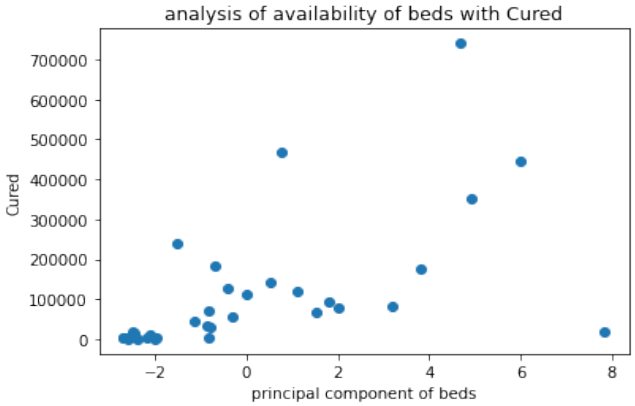

From the above graph, we see a positive correlation

We have also analysed how each state is performing and figuring out which states have contained the spread well compared to the others. Here, we take in the ratio of the confirmed cases to the population for the comparison.

Here, we find that ratios are high with places of smaller population. For example Uttarakhand, Goa and Puducherry. We do know that higher the population, more difficult it is to contain the spread of virus in the state, therefore states with lesser ratio are doing a better job despite their huge population

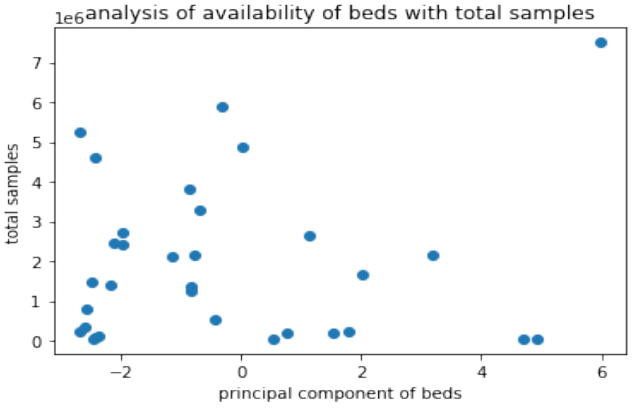
*B. Analyse the availability of hospital beds with the recovery rate*

Our Beds data set contains the number of urban beds,rural beds and public beds available along with the number of community,private,district and sub-district health cares available for each state. This dataset contains 6 NaN values in one column. We have filled the NaN values with the mean of the column as the column contains continuous values. We are assuming that all the columns are uncorrelated and analysing all these columns with respect the cured rate for each state.As the features for analysis is multi-dimensional, we must perform dimensionality reduction for further analysis. We use Principal Component Analysis to acheive the same. We find that the correlation between Principal Component value and the recovered values is 0.588.




We take the recovered cases from the covid 19 dataset where we find the the cured column is positively skewed containing outliers.

We also included the visualizations for the relation between the principal component and the total samples. Here, we find that two columns are not correlated at all with a Pearson correlation coefficient of 0.096. This explains the strong relationship between the increase in cured cases with the increase in facilities in all the states




*C. Analyse age groups at higher risk and percentage affected*

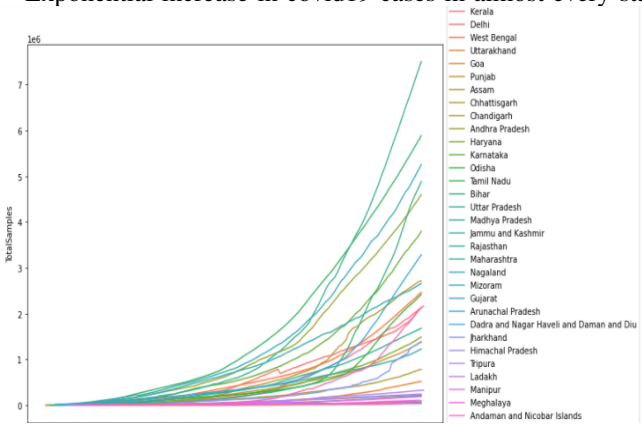
The Age dataset contains the age group, total cases and percentage. We have 9 missing values which have been taken care of. Here we find that the age group of 20-29 are affected the most. 25% of the total covid cases are from this group.

We also have a dataset that analyses each and every individual that has contracted the virus in the beginning stages. Not much analysis can be done as most of the details of the patient history are not available. But from this data, we do gather that most of the individual cases are from the state of Maharashtra in the beginning stages. The same result was expected.

While analysing the State Dataframe, we found that the number of positive cases and the number of negative cases do not add up to the total samples given in the dataframe. This made us doubt the correctness of the dataset. We analysed the trend in the total samples of each state and found that the graph is exponential growth in the total samples from April 1st 2020 to September 14 2020. Some areas like Megalaya,Sikkim,Puducherry,Himachal Pradesh, Arunachal Pradesh etc. remained constantly low in covid

samples till the very end but places like Odhissa, Karnataka, Andhra Pradesh show an exponential growth in cases. We find Uttar Pradesh having the highest number of samples and the islands having the lowest.

Exponential increase in covid19 cases in almost every state




We plan on using the dataset on which the above graph was plotted for predicting the trend in the rising cases in each State. We plan on estimating through many approaches including ARIMA,SVMs and Regression by re training on our dataset at specific intervals of time. We have considered various other factors that are small but effective in providing some insight to the rising cases in India. Our approach for the predictions will be based on the model that gives the best outputs.

*D. Proposed models for predicting the rise in number of covid cases*

SARIMA MODEL:

1. Checking for stationarity of the time series:

Time series can be performed only on data that is stationary. Augmented Dickey Fuller (ADF) Test is used to check for the stationarity of the time series data. The mean and standard deviation of the time series is also found. It is observed from the results that there is a huge gap between the original data and the mean and standard deviation [Fig. 1]. Augmented Dickey Fuller test found p value to be 0.982 which is very high [Fig. 2]. We can conclude that the time series is not stationary.

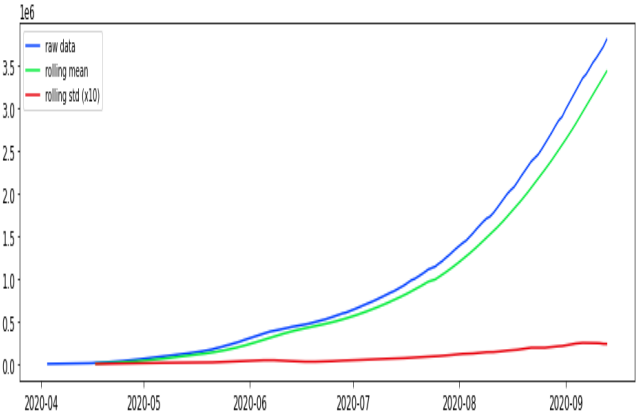



Fig 1. Mean and Standard deviation with the time series

2. Converting non stationary data to stationary:

A number of methods are available for converting non stationary data to stationary data. We have used differencing. After differencing, detrending is also performed which removes any trend present in the time series. Data is plotted after differencing and detrending. It can be seen in fig. 3 that the data after detrending and differencing is stationary.

Fig. 3 Plot after detrending and differencing

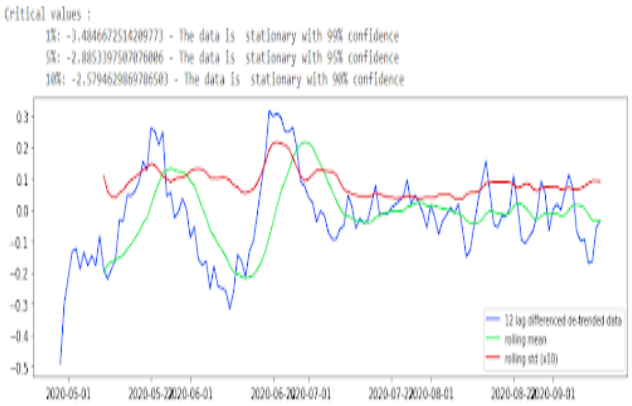



Fig. 4 Results of the Augmented Dickey Fuller test on detrended and differenced data.

```
Results of Dickey-Fuller Test:
 > Is the raw data stationary ?
Test statistic = 0.418
P-value = 0.982
Critical values :
        1%: -3.476273058920005 - The data is not stationary with 99% confidence
        5%: -2.881687616548444 - The data is not stationary with 95% confidence
        10%: -2.5775132580261593 - The data is not stationary with 90% confidence
```

3. SARIMA model :

SARIMA is used for forecasting because the data is seasonal. ARIMA cannot be used when the data is seasonal. SARIMA is an extension of ARIMA but unlike ARIMA, SARIMA works on seasonal elements. Trend elements used in SARIMA are p, d and q which are auto regression, difference and moving average order respectively. Seasonal elements used are P, D, Q and m which represent seasonal autoregressive, seasonal difference, seasonal moving average and number of steps for a single seasonal period. Grid search is used to find optimal parameters for SARIMA forecasting. It uses AIC value as an evaluation metric. Lower the value of AIC better are the parameters. The optimal parameters obtained is shown in fig.5

```
sarima_grid_search(y,52)
```

```
The set of parameters with the minimum AIC is: SARIMA(1, 1, 1)x(1, 1, 0, 52) - AIC:1161.741948079849
```

Fig.5 SARIMA parameters found after grid search

The parameters are passed to the model

and the forecasted results are shown in fig 6.

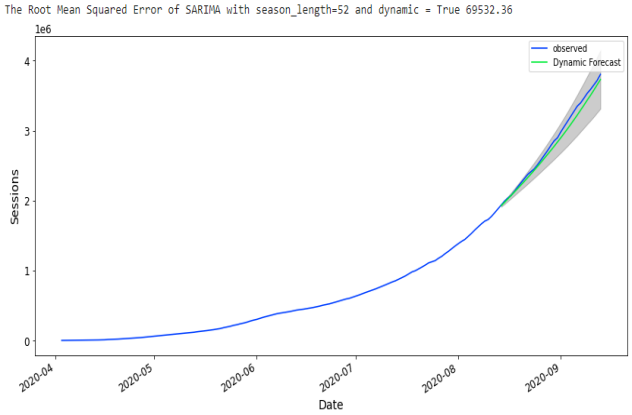



Fig.6 SARIMA forecasting

EXPONENTIAL SMOOTHING METHOD(HOLT AND HOLT WINTER)

To analyse the trend in the increase in covid cases, we restricted ourselves to analyse the trend in Karnataka. We will perform Holt and Holt Winter analysis and observe the closeness of each plot with respect to the test values. Figure 7 shows the exponential increase in the cases in Karnataka.We divided the data into train and test where train has 90 entries and test dataset has 20 entries.We fit the Holt model on the train dataset and use it to forecast the remaning 20 values. Next we compare these values with the test data and analyse our results

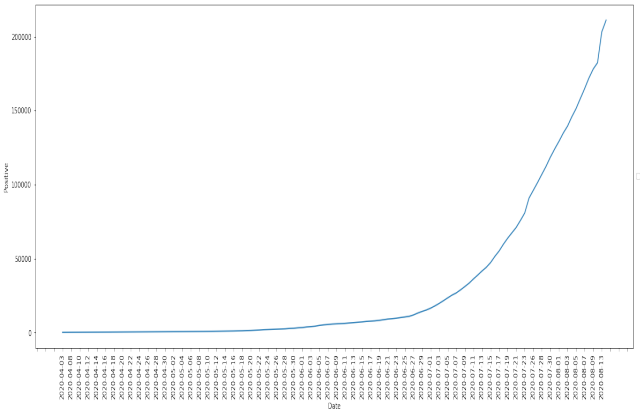
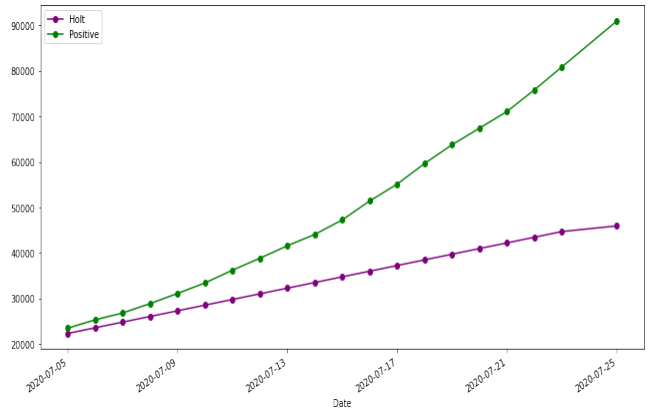


Fig 7 rising covid cases in Karnataka

HOLT TREND:

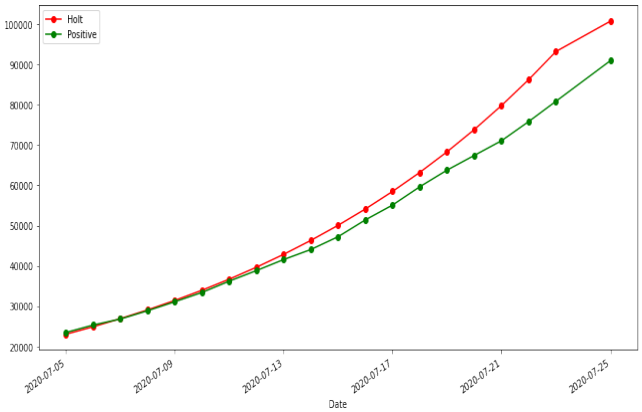
As we know, most time series data contains some kind of trend or seasonality. Our data clearly shows a trend. Therefore we estimate using the Holt Linear model that is an extension of the Simple Exponential Smoothing method that also adds a trend smoothing parameter. Here, we are testing 2 variations of the Holt trend and we find that the RMSE value for the exponential model is much lower than the linear model.

HOLT LINEAR TREND:




The Root Mean Squared Error of Holt's Linear trend 20011.1

HOLT EXPONENTIAL TREND




The Root Mean Squared Error of Holt's Exponential trend 5212.13

HOLT - WINTER MODEL:

Holt-Winters is a triple exponential smoothing algorithm which is an extension of the Holt method. Here, the seasonal component is also taken as a part of exponential smoothing.However including the seasonality component decreased the accuracy of our model. We experimented with both multiplicative and additive seasonal trend. The time series value for multiplicative seasonality and additive trend is given by:

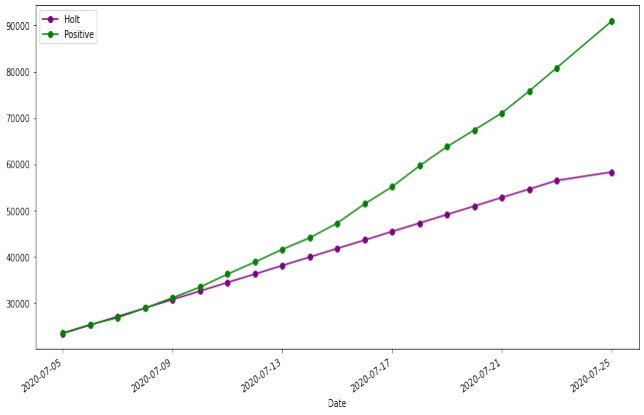
$$T(i+1) = (L(i) + B(i)) * S(i+1 - m) * N(i+1)$$

where T is estimated time series value at i+1 , L(i),B(i) are the estimated step values, S is the seasonal value, and N is the observed noise

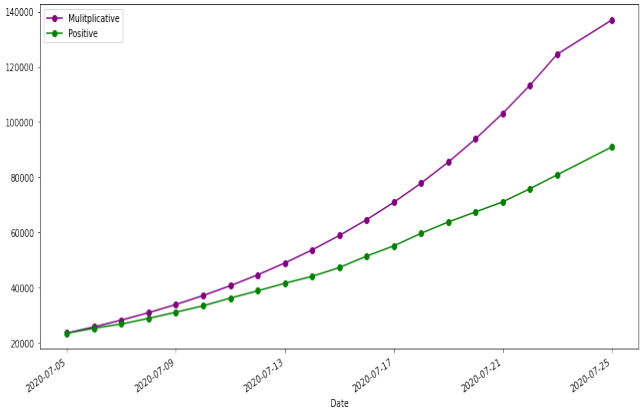
The time series value for additive seasonality and additive trend is given by:

$$T(i+1) = (L(i) + B(i)) + S(i+1 - m) + N(i+1)$$

Below is the seasonal additive and trend additive plot




Below is the seasonal multiplicative and trend additive plot



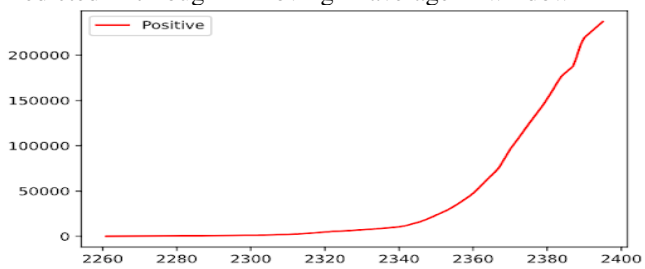

Root Mean squared error value is 20904.523

From the above observations we can infer that the HOLT EXPONENTIAL TREND without adding the seasonal component gave best results when compared to the test data.

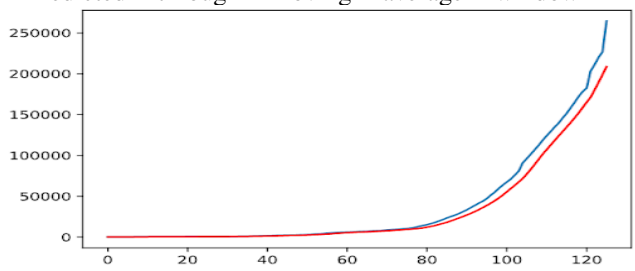
MOVING AVERAGE AND AUTOREGRESSIVE MODEL

Moving Average: In moving average method of time series prediction, we have considered the dataset with Karnataka as the state to predict the number of positive cases on a daily basis (based on date). Neglecting the Nan values, we find out the rolling mean with a window = 3 in our example i.e. it finds the mean value of the 2 most recently used data combined with 1 next new value in order to predict the number. The blue plot shows the original data and the red plot represents the predicted value obtained through moving average. We find the different predicted values of the model by varying the window size (2,3,5) and here we find that it performs well for window=2 rather than when window=5. One of the main problems of the Moving Average method is the LAG. Since the average of the data is calculated every time, it always lags behind the actual value. Through multiple iterations as specified above and as shown in the combined plot, window=2 performs relatively better as it captures the trend in the time series data much faster and more importance is given to the most recent data. Whereas when the window size increases the contribution of the most recent data to the moving average is much less. Thus, we can say that window size is inversely proportional to the accuracy of the model. Please note that equal weightage is given to all the values considered for average. The moving average technique generally works fine when the data is nearly constant or when the increase or decrease in the values is gradual. If the data fluctuates rapidly, this method is definitely not preferred. An enhancement to this model could be to introduce weights into the moving average model, where more weightage is given to the most recent value and the weights subsequently decrease thereafter to the older values. Nevertheless, the lag of the data remains in Moving Average Technique.

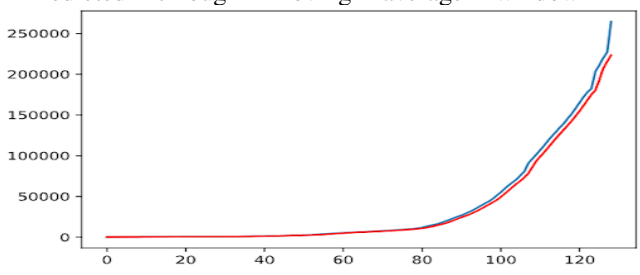
Predicted through moving average window = 3




Predicted through moving average window = 5



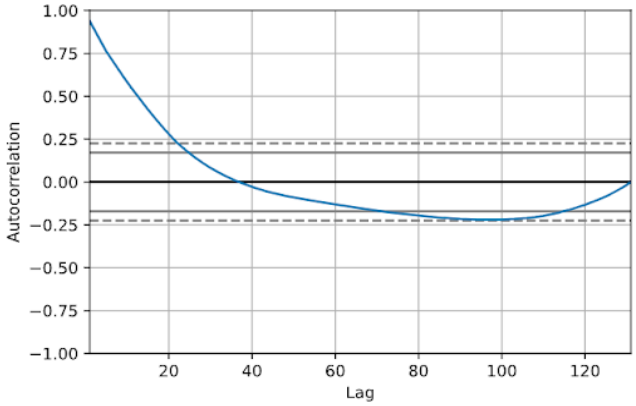

Predicted through moving average window = 2

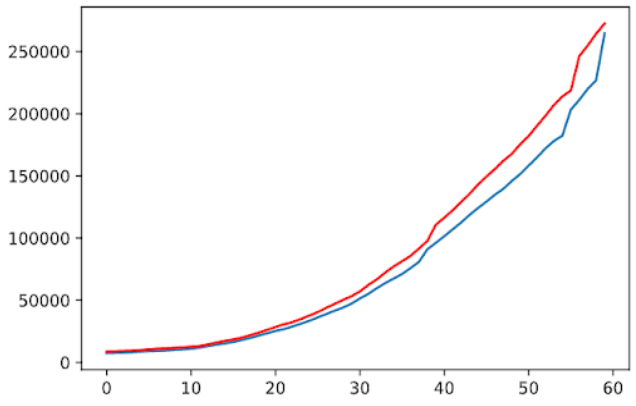



Autoregressive Model:

In this model too we have considered the dataset with Karnataka as the state to predict the number of positive cases. The Autocorrelation Function and plot is made use of in this model. This function is actually a similarity of the observations from the dataset when compared to the time lag in between them. Refer to the ACF plot provided alongside. One more way to define the ACF is that it can be looked at as a linear relationship between a particular observation under consideration and all the previous observations taken with respect to it. The order of the Autoregressive Model can be found out by using the PACF (Partial Autocorrelation

Function). The AR Model is used here to predict the time series data when there seems to be a correlation in the values. The first step is to train the autoregressive model with partial dataset values. After multiple iterations the optimum value for lag and window and the other parameters have been found out. For this model we have used the Autoreg function available in the statsmodels tsa-ar model module in python. The blue plot shows the original data values and the red plot shows the predicted value in the combined plot for AR Model. The expected and predicted values have been printed for reference along with the RMSE. From this model we find out that it somewhat works correctly for predicting the values of positive cases. In the plot shown the predicted values almost seem to follow the original values but towards the end the gap seems to be increasing. With more training data the model could have been better at predicting the positive cases.

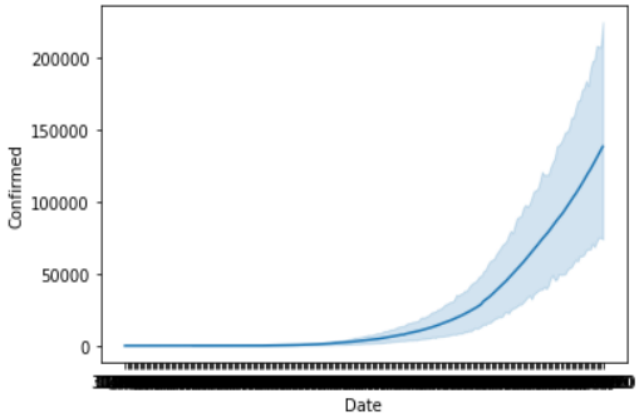


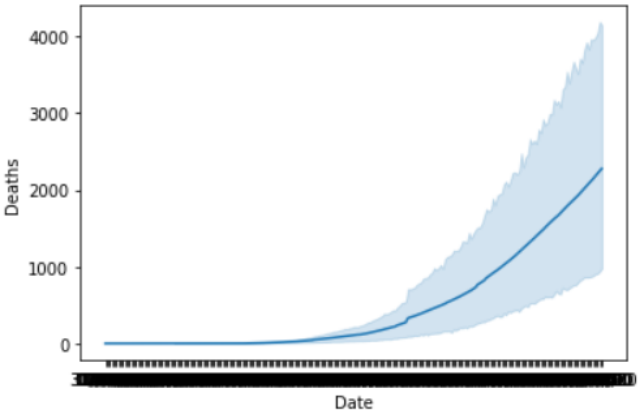

ACF AutocorrelationPlot

RANDOM FOREST REGRESSION MODEL:

Random Forest is an effective ensemble machine learning algorithm. It can be used for both classification modeling and regression predictive modeling problems. RF model is be used for forecasting, time series problems and find the causal relationship between the variables. Here, we try to find the relationship of between date and number of cases both confirmed and death ,use it to predict and see if cases would increase in the future. We construct trees to train the model, then use it to forecast. It is tough to make a Random Forest algorithm to properly extrapolate. To use RF for time series forecasting we need to first transform data set into a supervised learning problem. We'll have to use walk-forward validation to evaluate the model. Because if we use k-fold cross validation for evaluating (error measure) it would result in optimistically biased results. RF is an extension of bootstrap aggregation of decision trees, which is also called bagging Therefore, prediction is the average of the prediction across the trees in the ensemble. First we make Time series data as supervised learning data, This is done by using the preceding time step variable as input to the model to forecast value of next time step variable which will be the output value. Now we train the model on the past and predict the future. We used Walk-forward validation to evaluate the model. That is, we first choose a cut-point to split the dataset into train and test. The evaluation is done by first training and then testing the model. We train it first using the training dataset. Then we try to forecast the first value of the testing dataset. We can then remodel it by adding the first real value from test dataset to train dataset and then test the model again, that is predict value of the second time step. We do this for the entire dataset to get the error measure and evaluate the model. The model predicted the mean to be 22907 confirmed cases and 4000 deaths.




Predicted confirmed cases




Predicted Death cases

## III. References:

[1]https://www.researchgate.net/publication/341297682$_Tracking_the_Sp$
$19_Cases_in_India_using_Data_Visualizing_and_Forecasting_Techniques$

[2]https://arxiv.org/ftp/arxiv/papers/2004/2004.07859.pdf

[3]https://core.ac.uk/download/pdf/41373796.pdf

[4]https://www.bbc.com/news/business-51706225