International Journal of Current Research and Review
DOI: http://dx.doi.org/10.31782/IJCRR.2021.SP223

Scopus

Original Research

IJCRR
Section: Healthcare
ISI Impact Factor (2019-20): 1.628
IC Value (2019): 90.81
SJIF (2020) = 7.893

AI-based Pandemic Trend Analysis

Vergin RSM[1*], Anbarasi JL[1], Graceline JS[1], Valarmathi ML[2], Mayank V[1], Yash D[1], Upender S[1]

[1]School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, Tamil Nadu, India; [2]Professor, Dr. Mahalingam College of Engineering and Technology, Pollachi, India.

ABSTRACT

Introduction: The current outbreak of COVID-19 has caused the world to stop and go under lockdown and has quickly grown to become a pandemic. The clinicians and scientists in medical industries are observing the pandemic for screening the COVID -19 virus in a person.

Objective: In these trying times, we thought of analysing the trends in COVID-19 cases in the USA, India and Brazil using several Time Series, Machine Learning and Ensemble Learning algorithms to check out the trends.

Methods: In this paper, Auto-Regressive Integrated Moving Average (ARIMA) and Long Short-Term Memory (LSTM) under Time Series, Support Vector Regression (SVR) and Linear Regression under Machine Learning algorithms and Random Forest Regression, XGBoost and AdaBoost under Ensemble Learning were discussed.

Results: After analyzing the results of all the algorithms, we observed that ARIMA and LSTM were performing better than the others for Time-Series Forecasting. This study would be valuable for medical Researchers and the Government in the future.

Conclusion: Seven models, namely, ARIMA and LSTM models under time-series analysis models, support vector regression and linear regression under machine learning models and random forest regression, XGBoost and AdaBoost under ensemble learning were discussed. We first looked at the sample fits and then successfully forecasted the trends for the new cases, deaths and total cases for the next 30 days in the two countries with the highest number of cases, namely, India and the US. From the resultant graphs and table values, we could infer that overall, time-series models like ARIMA and LSTM perform the best in situations like these where data is continuous and forms a series.

Key Words: Pandemic, COVID-19, Time Series, Machine Learning, Ensemble Learning

INTRODUCTION

COVID–19 is a recently discovered disease that is caused by the newly discovered coronavirus or Novel SARS–CoV2 viruses. It causes infection in the respiratory tract which can be mild or in the worst case, fatal. This virus was first reported in Wuhan, the Hubei region of China in the middle of December from where it has quickly spread over the whole world to become a pandemic with countries like the US, India, Brazil, Russia and European countries being the worst hit. Medical facilities have been incapable of treating this alarming growth of new cases every day which gives us a picture of the seriousness of the situation we're facing currently. Hence, we thought that if there is any estimated figure about the new cases, deaths and total cases, then it would be a great aid to the medical forces which proves to be our motivation for this research. Our work can help them in determining a better estimate of new cases so that they can make appropriate preparations accordingly. Our objective is to work on different time series, machine learning and ensemble learning algorithms and compare these algorithms.

For our study, Auto-Regressive Integrated Moving Average (ARIMA) and Long Short-Term Memory (LSTM) under Time Series, Support Vector Regression (SVR) and Linear Regression under Machine Learning Algorithms and Random Forest Regression, XGBoost and AdaBoost under Ensemble Learning were discussed few algorithms used in this research work includes Auto-Regressive Integrated Moving

Corresponding Author:
Vergin Raja Sarobin, School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, Tamil Nadu, India.
Phone: 9710423195; Email: verginraja.m@vit.ac.in

ISSN: 2231-2196 (Print) **ISSN:** 0975-5241 (Online)

Received: 13.03.2021 **Revised:** 20.04.2021 **Accepted:** 12.05.2021 **Published:** 11.06.2021

Average (ARIMA), Long Short-Term Memory (LSTM), Support Vector Regression, Linear Regression, XGBoost, AdaBoost.[1-3]

RELATED WORK

The study of Time Series Analysis of the COVID – 19 pandemic is generally carried out for selecting the best-suited algorithm for growth prediction of the pandemic.[4-6] It emphasizes the 2 major difficulties of time-series analysis that it can't be applied to continuous data and the second one being, it includes some exogenous features which can't be mapped to ML models. It covers 3 major ways, viz., by pure machine learning models like ARIMA or 'Autoregressive Integrated Moving Average', by RNNs or 'Recurrent Neural Networks' like LSTMs or 'Long Short-Term Memory Networks' and lastly, 'Extreme Learning machines' like 'Online Sequential Extreme Learning Machines'.[7] Application of artificial intelligence and computational intelligence techniques is diversified in various areas such as e-healthcare, smart city and smart grid, data processing, predictive maintenance etc.[8-10] Likewise, the application of AI-based techniques plays a vital role in this research work.[11]

Our study dives into the analysis part of COVID – 19 focusing on Long Short-Term Memory Networks or LSTMs.[12] The study first plots the general trend of COVID – 19 in India along with the fatality rates of various deadly viruses over the past 50 years. It then shows the main result plots according to the data have driven approach and the curve fitting methods. It includes prediction about estimated total confirmed cases, estimated recoveries, etc in the foreseeable future. Conclusions say that social distancing and announcing lockdowns can reduce its spread significantly.

Technical sense and the amount of mathematics involved in the algorithm are also very important to select the accurate and best-suited algorithm as per the nature of the dataset. An 'Experimental Study' goes on for the comparison of LSTM & ARIMA.[13] A brief comparison is done based on various factors like size and structure of datasets, data pre-processing and assessment metrics. Conclusions are made from the comparison of the performance of both the models based on tuning of hyperparameters like the number of epochs, batch sizes, etc.

The study and analysis about AdaBoost mainly showcase three boosting algorithms, viz., AdaBoost.R2, AdaBoost.M1 and the main one AdaBoost.RT.[14] They compare these three algorithms side by side and based on data sets and methods. AdaBoost.RT which is the newer one performs significantly better than the others and gives better results because the parameter log(1/beta) is always ensured to be non-negative and the harder examples are always given more emphasis in AdaBoost.RT.

The ARIMA model for new cases and new deaths daily during this period is more suitable for short-term prediction. ARIMA is the most common time series prediction model in the statistical model. It regards the data sequence formed by the prediction object over time as a random sequence. It analyses a portion of the data in the sequence to obtain specific parameters that describe the mathematical model of the sequence to achieve time series modelling and use the remaining data in the sequence to validate the model. It can be used to predict the subsequent values of the data series.[4,15,16]

Newly developed drugs usually take years to be successfully tested before coming to the market. X-ray images and computed tomography (CT) scans are widely used as the input of DL[17] models to automatically detect the infected COVID-19 case. Infected COVID-19 patients normally reveal abnormalities in chest radiography images. A phone-based framework for COVID-19 detection and surveillance is proposed. The DL model can be trained in the cloud, even at a server collocated at the network edge, which is then pushed to mobile phones for further purposes.[5-7]

Visualization techniques used to visually represent the spread of COVID-19 pandemic [20] is also an important part of this study and is expected to demonstrate the utility of the new system in terms of comparing rate spread across different countries and different times. It should also indicate the rate and trend of spread over time and by comparing with past examples, the system should also be used to predict the future rate of spread. It will also be integrated into automatic speech recognition and text to speech features to disseminate information to people with different range of abilities.[12-14,17]

For analyzing the growth and trend of the ongoing pandemic COVID-19, it is shown that iterative weighting for fitting Generalized Inverse Weibull distribution is a better fit that can be obtained to develop a prediction framework.[18] The study observes that using the iteratively weighted approach, the Inverse Weibull function fits the best to the COVID-19 dataset, as compared to the iterative versions of Gaussian, Beta (4-parameter), Fisher-Tippett (Extreme Value distribution), and Log-Normal functions. When applied to the same dataset, Iterative Weibull showed an average MAPE of 12% lower than non-iteratively weighted Weibull. An algorithm for iteratively weighted curve fitting using the GIW distribution (called" Robust Weibull") is used.

The SIR (Susceptible, Infected, Recovered) model,[22] commonly used to predict the growth of epidemic, is not much effective for today's scenario because of the basic assumption made in the model that the total population under consideration does not change with time and is not valid under the current Indian circumstances. The rate of change of the total population, consisting of susceptible, infected and recovered (or dead) population, with time is zero is not valid. A simple model conceptualized based on an analogy to

compound interest formula used in engineering economics, seems to work better in this case.[18,19]

The support vector regression is also being used as one of the algorithms. The idea of SVR is based on the computation of a linear regression function in a high dimensional feature space where the input data are mapped via a nonlinear function. SVR has been applied in various fields – time series and financial prediction, approximation of complex engineering analyses, convex quadratic programming and choices of loss functions, etc.[20]

PROPOSED WORK

Data Source

The data we used is the official data for COVID-19 taken from the World Health Organisation Database. The Dataset contains the data for COVID-19 patient counts from all over the world separated by country, and we've done the predictions for India, The US and Brazil only. The considered data is from the date the first COVID-19 case came in their respective countries to 7th of December, 2020 and we've forecasted the data for the next 19 days.[7]

Data Pre-processing

Firstly, we have cleaned the data and set the dates according to the U.K. format. Then we have imported the Date column along with the new cases column and total no. of deaths column. We have taken the Dates in variable X and no. of new cases or no. of deaths (depending upon what we are predicting) in variable Y. In some of the models like linear regression and random forest, the dates are accepted only in integers, therefore we have converted the dates in integers specifically for these models. Then we have taken the data, trained the data according to the chosen Machine learning model by splitting the dataset into a 90:10 ratio. 90% of the dataset trains the models and the rest 10% of the dataset compares the predicted outcomes with the actual statistics. After the comparison, we have used this data to perform forecasting for further dates. The training and testing split of ARMIA and LSTM was done without shuffling while the rest of the model was shuffled.

Time Series Algorithms

Auto-Regressive Integrated Moving Average (ARIMA)

Auto-Regressive Integrated Moving Average, in short ARIMA, is a popular machine learning model which is most suitable for forecasting a time series. It is a class of models that learns from its past values, errors and lags, and which the model uses to predict future outcomes.

Autoregressive models:

The terms auto regression indicates that it is a regression of itself. The equation for autoregressive model of order p is (1):

$$y_t = \delta + {}_1y_{t-1} + \varphi_2 y_{t-2} + \ldots + \varphi_p y_{t-1} + A_t \quad (1)$$

where y_{t-1}, $y_{t-2}\ldots y_{t-p}$ are the past series values (lags), A_t is white noise (i.e. randomness) & δ is defined in (2):

$$\delta = (1 - \Sigma_{i=1}^{p} \quad \varphi_i)\mu \quad (2)$$

Moving Average models:

Moving average models are regression like models but instead of using past values, they use past forecast errors. The equation can be written as in (3) :

$$y_t = c + \varepsilon_t + \theta_1\varepsilon_{t-1} + \theta_2\varepsilon_{t-2} + \ldots + \theta_q\varepsilon_{t-q} \quad (3)$$

Here, ε_t is white noise and q is the order.

After combining autoregressive and moving average model we get ARIMA, whose equation can be written as in (4) :

$$y'_t = c + \phi_1 y'_{t-1} + \ldots + \varphi_p y'_{t-p} + \theta_1\varepsilon_{t-1} + \ldots + \theta_q\varepsilon_{t-q} + \varepsilon_t \quad (4)$$

where y'_t is the differenced series (it may have been different more than once)? This is called the ARIMA (p,d,q) model. (Here, p = order of the autoregressive part, d = degree of first differencing involved, q = order of the moving average part.)

Long Short-Term Memory (LSTM)

LSTM is a typical Recurrent Neural Network architectural algorithm which is known for its capability to store or rather 'remember' data for a specified period because of which these are widely used in time-series analysis, speech-recognition, or any other applications where data is continuous and needs to be remembered. Since LSTM is an RNN model, it is composed of many individual cells connected to form layers where the output from one cell is passed to the further cells. Talking about the structure of a single cell of an LSTM, it mainly comprises 3 gates, viz., the forget gate, input gate and the output gate. The forget gate is responsible for the 'remembrance' function. It does so by utilizing a sigmoidal function and compares the current data with the previous data and accordingly gives a value of 0 or 1 which are for forgetting and remembering respectively. The input gate determines up to what extent this passed data has to be remembered or forgotten and hence associates a weightage to the data being passed. Again, a sigmoidal function decides what data to keep and a hyperbolic tangent function decides how much data to keep. Finally, the output gate determines which part of the cell has to be passed as an output. It again utilizes sigmoidal and hyperbolic tangent functions to decide what to pass and how much to pass to the output layer. The LSTM model can be described by the following equations

where (5) belongs to the forget gate, (6) and (7) belong to the input gate and (8) and (9) belong to the output gate.

$$f_t = \sigma(W_f . [h_{t-1}, x_t] + b_f) \quad (5)$$

$$i_t = \sigma(W_i . [h_{t-1}, x_t] + b_i) \quad (6)$$

$$C_t = \tanh(W_C . [h_{t-1}, x_t] + b_C) \quad (7)$$

$$o_t = \sigma(W_o . [h_{t-1}, x_t] + b_o) \quad (8)$$

$$h_t = o_t * \tanh(C_t) \quad (9)$$

Machine Learning Algorithms

Support Vector Regressor (SVR)

Support vector machines (SVMs), like random forest algorithms, are also a popular choice for classification and regression models. Support vector regression is the algorithm in SVM which is used for the regression problems. This regression model uses a hyperplane and fits it such that the hyperplane contains most of the points from the space. It also creates decision boundaries that enclose the points within an area and from which we determine the best fit line i.e. the hyperplane.

For this, we assume the hyperplane to be in the form as in (10):

$$Y = sX + t \quad (10)$$

Then, (11) become the equations of decision boundary (where d is the assumed distance from the hyperplane):

$$sX + t = \pm d \quad (11)$$

Therefore, any hyperplane that satisfies our SVR should satisfy the condition as in (12):

$$-d < Y - sX + t < +d \quad (12)$$

Then we focus on finding the decision boundary at the distance of 'd' from the assumed hyperplane such that data points that are closest to the hyperplane (or the support vectors) are within that decision boundary line. From here, we only consider the points which are within the decision boundary and the one that has the least error rate or are within the Margin of Tolerance (epsilon **ε**). This gives us the best possible hyperplane and thus gives us a better fitting model. Support vector regressor shows the presence of the non - linear nature of data in the dataset and makes an effective prediction model.

Linear Regression

Linear Regression is a common machine learning model and a well-known algorithm in statistics that assume a linear relationship between two variables. One variable is considered a dependent model while the other variable is considered to be an independent model, i.e. y can be calculated from the linear combination of x (the input variable). The equation of a linear regression line is of the form Y = a + bX, where X is the independent variable and Y is the dependent variable. Here, b is the slope of the line, and a is the intercept. Different techniques can be used to prepare or train the linear regression equation from data, the most common of which is called Ordinary Least Squares. The linear regression uses a linear equation that assigns one scale factor to each X, known as the coefficient and is represented by the capital Greek letter Beta (β). Forgiving an additional degree of freedom, one additional coefficient is also added (e.g. moving up and down on a 2-D plot) and is called the intercept or the bias coefficient.

Ensemble Learning Algorithms

Random Forest

Random forest is a supervised learning algorithm and a machine learning model which is popularly used for classification and regression in the data. Here, we have used the Random forest regression for the prediction of the data by training the older dataset. The random forest algorithm consists of many decision trees which work on the data and combine the result. The more trees are in the training of data, the more robust and effective the random forest regression will be. This algorithm creates decision trees on the dataset and then gets the prediction from every tree to select the best solution. First, we select random samples from the given dataset. Next, a random forest constructs a decision tree for every sample data. Then it will get the prediction result from every decision tree. Then, voting will be performed for every predicted result obtained from the decision trees. At last, the most voted prediction result is selected as the final prediction result. This model is very flexible and yields high accuracy results but is also complex which is the main disadvantage for the regression part that predicts these types of dataset a difficult and time-taking task.[11-13]

XGBoost

XGBoost, the acronym for eXtreme Gradient Boosting, is a decision-tree-based ensemble Machine Learning algorithm that works on a gradient boosting framework. It has recently become popular in machine learning for structured or tabular data processing. This algorithm uses gradient boosted decision trees which helps in the performance and reduces the time taken compared to other algorithms. The gradient boosting used in this model is mainly of three forms: (i) Gradient Boosting Machine. (ii) Stochastic Gradient Boosting. (iii) Regularized Gradient Boosting. One of the main highlights of this model is the use of parallel and distributed computing for faster learning. XGBoost decision trees use gradient boosting and advance regularisation which yields more accurate approximation. These approximations are ensemble

and an optimal output is calculated while using parallel computing which makes the model faster for bigger datasets.

AdaBoost

AdaBoost is mainly compatible to work with either decision tree classifiers or random forest classifiers. In AdaBoost, the tree produced is generally based on only a single feature variable. Hence, it can have only two outputs at any given point in time. These types of tree structures with only one node and two leaves are called stumps and are said to be weak learners. An AdaBoost classifier is said to work by combining these stumps where every new stump is created by taking into account the results and errors of the previous stumps by refining them by using measures such as Gini index. Hence, an AdaBoost algorithm effectively acts as an ensemble learning algorithm and some stumps are said to have a greater say in the output. The final decision is then taken by a random forest tree which is a combination of all these stubs.[20,21]

RESULTS AND DISCUSSIONS

In this section, we have shown the graphs for the forecasting done by all the models for the next 30 days (1st Nov 2020 to 29th Nov 2020) along with the previous original data. After that, we have shown a performance metric table which evaluates the models based on four evaluation metrics, namely, R-squared, MSE, RMSE and MAE, although we'll be presenting our inference based on MAE only because it shows considerable difference in its value for every model and every case. Hence, it would be easier to present our inferences according to that metric. Figure 1(a) details the forecasting of total deaths from COVID-19 in India and figure 1(b)details the forecasting of total COVID-19 cases in India. Table 1 details the performance comparison for every model for India.

INDIA

In Time Series Models, for cumulative cases, ARIMA gives the best results with MAE as 107740.030 followed by LSTM with MAE as 8616082.966. In cumulative deaths, ARIMA is far better than LSTM with MAE of ARIMA as 1315.637 and MAE for LSTM as 114679.320.

In Machine Learning Models, it is observed that for cumulative cases, the MAE for SVR and linear regression is 1719540.726 and 1508130.682 respectively, and for cumulative deaths, the MAE for SVR and linear regression is 37992.886 and 17203.704 respectively. From this, we can say that linear regression outperforms SVR in all cumulative cases and cumulative deaths.

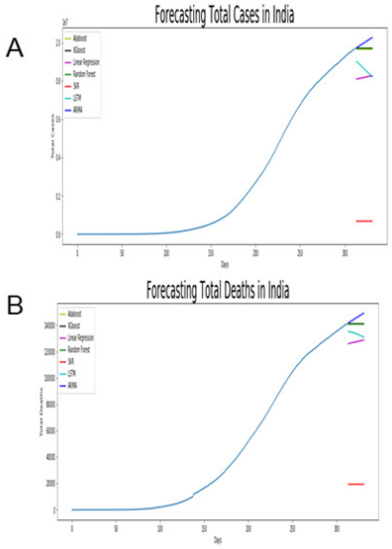

Figure 1: A. Graph for forecasting Total Deaths from COVID-19; B. Graph for forecasting Total COVID-19 cases in India.

In Ensemble Learning Models, it is observed that for cumulative cases Random forest outperformed all the three models in this section with MAE as 30311.440 followed by XGBoost with MAE as 30531.9477, AdaBoost performed least best with MAE as 42925.859. For cumulative deaths Random Forest outperformed all the three models in this section with MAE as 459.386, followed by AdaBoost with MAE as 566.521, XGBoost performed least best with MAE as 631.741.

Table 1: Performance comparison for every model for India

India	Type	Prediction Models	R Squared	MSE	RMSE	MAE
Cumulative Cases	Time Series	ARIMA	0.8758	16354393574.572	127884.297	107740.030
		LSTM	-9.0210	80953848943899.75	8997435.687	8616082.966
	Machine Learning	SVR	-0.1073	8944915599398.174	2990805.176	1719540.726
		Linear Regression	0.6743	2630821261605.109	1621980.660	1508130.682
	Ensemble Learning	Random Forest	0.9997	24995392 86.298	49995.393	30311.440
		XGBoost	0.9997	2229931592.092	47222.151	30531.947
		AdaBoost	0.9996	3170602462.414	56308.103	42925.859

Table 1: ***(Continued)***

India	Type	Prediction Models	R Squared	MSE	RMSE	MAE
Cumulative Deaths	Time Series	ARIMA	0.8889	2232560.785	1494.175	1315.637
		LSTM	-5.7230	14892942466.577	122036.644	114679.320
	Machine Learning	SVR	-0.1375	2519912801.492	50198.733	37992.886
		Linear Regression	0.8242	389344517.105	19731.815	17203.704
	Ensemble Learning	Random Forest	0.9997	472164.398	687.142	459.386
		XGBoost	0.9996	832313.411	912.312	631.741
		AdaBoost	0.9997	639043.292	799.402	566.521

United States of America

In Time Series Models, it is observed that for cumulative cases, the MAE for ARIMA and LSTM are 757658.254 and 13695809.436 respectively, for cumulative deaths, the MAE for ARIMA and LSTM are 5275.695 and 180772.712 respectively. Hence from these values, we can infer that ARIMA outperforms LSTM in cumulative cases and cumulative deaths (Fig 2a and b).

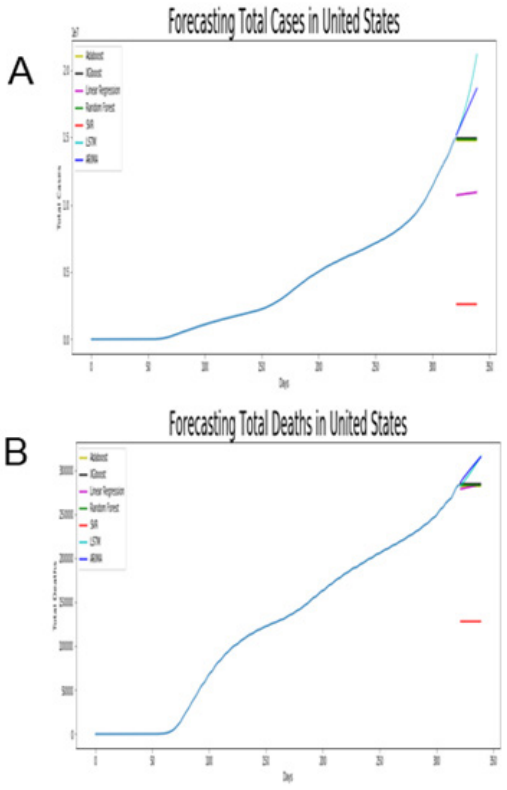

Figure 2: A. Graph for forecasting Total cases of COVID-19 in the USA, B. Total Deaths from COVID-19 in The US.

In the Machine Learning models for the cumulative cases, the performance comparison shows that the MAE for Support Vector regression is 38814.565 whereas the Linear regression model gave 17334.325 as MAE. Again, for the cumulative deaths, Support vector regression scored the MAE as 93497.918 and Linear regression with the MAE of 10546.31. Therefore, we can say that Linear Regression outperforms SVR in every case (Table 2).

In the Ensemble Learning Models, it is observed that AdaBoost performs the best among the three ensemble learning models in Cumulative cases with the MAE of 547.453 whereas XGBoost gave the MAE of 566.871 and Random Forest MAE gave 584.867 which is the least among the three. For the Total deaths, XGBoost scored the MAE value as 1348.747 followed by AdaBoost as 1418.414 then Random Forest with the MAE value as 1446.547. Hence we can conclude that XGBoost performs the best in this case. Figure 2(a) details the forecasting of total deaths from COVID-19 in the USA and figure 2(b)details the forecasting of total COVID-19 cases in the USA. Table 2 details the performance comparison for every model for the USA

Brazil

In Time Series Models, it is observed that for cumulative cases, the MAE for ARIMA and LSTM are 212248.019 and 4521220.961 respectively, for cumulative deaths, the MAE for ARIMA and LSTM are 1999.133 and 82919.289 respectively. Hence from these values, we can infer that ARIMA outperforms LSTM in both cumulative cases and cumulative deaths (Fig 3 a and b).

Table 2: Performance comparison for every model for the US

USA	Type	Prediction Models	R-Squared	MSE	RMSE	MAE
Cumulative Cases	Time Series	ARIMA	0.6332	881252785211.682	938750.651	757658.254
		LSTM	-9.0271	910169295178 49.78	13820887.436	13695809.436
	Machine Learning	SVR	-0.1455	2821946203.084	53121.994	38814.565
		Linear Regression	0.8480	374467016.035	19351.150	17334.325
	Ensemble Learning	Random Forest	0.9997	777467.689	881.741	584.867
		XGBoost	0.9997	757674.104	870.445	566.871
		AdaBoost	0.9997	630338.227	793.938	547.453
Cumulative Deaths	Time Series	ARIMA	0.7156	56989050.871	7549.109	5275.695
		LSTM	-5.1518	37541561653.428	193756.449	180772.712
	Machine Learning	SVR	-0.0449	10313311417.869	101554.475	93497.918
		Linear Regression	0.9794	203436583.360	14263.120	10546.313
	Ensemble Learning	Random Forest	0.9995	4783557.287	2187.134	1446.547
		XGBoost	0.9996	3691735.934	1921.389	1348.747
		AdaBoost	0.9996	3692941.589	1921.702	1418.414

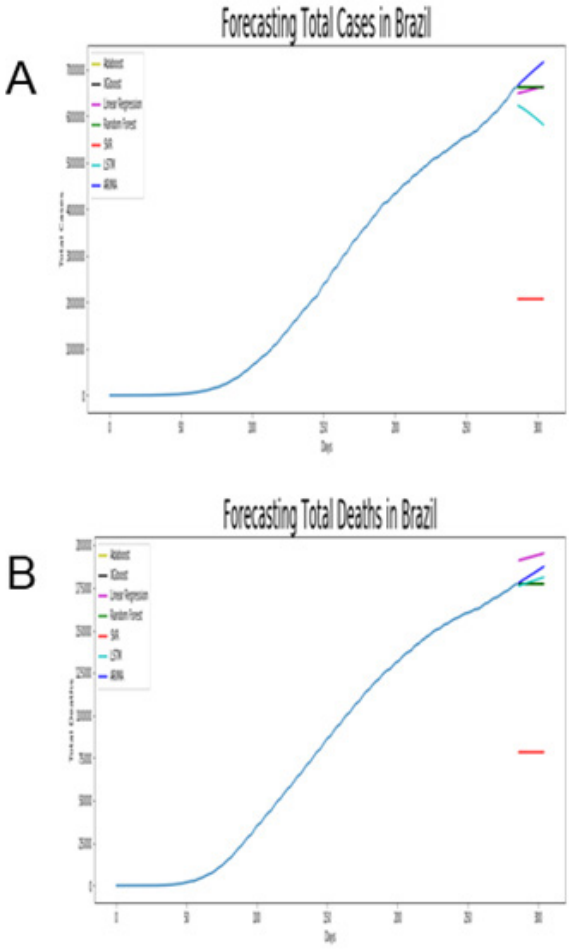

Figure 3: A. Forecasting Total cases of COVID-19, B. FTotal Deaths from COVID-19 in Brazil.

In the Machine Learning models for the cumulative cases, the performance comparison shows that the MAE for Support Vector regression is 2289047.022 whereas the Linear regression model gave 406039.526 as MAE. Again, for the cumulative deaths, Support vector regression scored the MAE as 57368.560 and Linear regression the least with the MAE of 7879.481 Therefore, we can say that Linear Regression outperforms SVR in every case (Table 3).

Table 3: Performance comparison for every model for Brazil

Brazil	Type	Prediction Models	R-Squared	MSE	RMSE	MAE
Cumulative Cases	Time Series	ARIMA	0.2018	60750397226.694	246475.956	212248.019
		LSTM	-3.041	25384459923740.91	5038299.309	4521220.961
	Machine Learning	SVR	-0.0270	6450971958832.363	2539876.367	2289047.022
		Linear Regression	0.9614	2424288143 75.774	492370.607	406039.526
	Ensemble Learning	Random Forest	0.9998	1203480559.545	34691.217	23866.147
		XGBoost	0.9998	1000910055.636	31637.163	21933.902
		AdaBoost	0.9998	1401766053.138	37440.166	27662.759
Cumulative Deaths	Time Series	ARIMA	0.7312	4873997.914	2207.713	1999.133
		LSTM	-1.7826	10761991942.264	103740.021	82919.289
	Machine Learning	SVR	-0.0249	3963977791.979	62960.129	57368.560
		Linear Regression	0.9740	100508622.761	10025.399	7879.481
	Ensemble Learning	Random Forest	0.9998	857197.328	925.849	720.713
		XGBoost	0.9997	1251878.876	1118.874	835.352
		AdaBoost	0.9997	1031079.684	1015.421	773.260

In the Ensemble Learning Models, it is observed that XGBoost performs the best among the three ensemble learning models in Cumulative cases with the MAE of 21933.902 whereas AdaBoost gave the MAE of 27662.759 which is the least among the three and Random forest MAE gave 23866.147. For the Total deaths, again AdaBoost scored the MAE value as 773.260 followed by XGBoost as 835.352 then Random Forest with the MAE value as 720.713. Hence we can conclude that Random forest performs the best in this case. Figure 3(a) details the forecasting of total deaths from COVID-19 in Brazil and figure 3(b)details the forecasting of total COVID-19 cases in Brazil. Table 2 details the performance comparison for every model for Brazil.

CONCLUSION

To conclude, in this paper, seven models, namely, ARIMA and LSTM models under time-series analysis models, support vector regression and linear regression under machine learning models and random forest regression, XGBoost and AdaBoost under ensemble learning were discussed. We first looked at the sample fits and then successfully forecasted the trends for the new cases, deaths and total cases for the next 30 days in the two countries with the highest number of cases, namely, India and the US. From the resultant graphs and table values, we could infer that overall, time-series models like ARIMA and LSTM perform the best in situations like these where data is continuous and forms a series. These are followed by the ensemble learning models like random forest regression, XGBoost and AdaBoost which perform the next best and finally followed by the machine learning models like support vector regression and linear regression performing not as per expectations. From our analysis, we can safely conclude that time-series models perform the best in situations where a continuous series of data is involved. As a caution about using these models, it is advised to not fully rely on the forecasting produced by the models every time as they highly depend upon the dynamics of the daily changing COVID-19 data.

ACKNOWLEDGEMENT

Authors acknowledge the immense help received from the scholars whose articles are cited and included in references of this manuscript. The authors are also grateful to authors/editors/publishers of all those articles, journals and books from where the literature for this article has been reviewed and discussed."

Conflict of Interest: "The Author(s) declare(s) that there is no conflict of interest.

REFERENCES

1. Wong SY, Tan BH. Megatrends in infectious diseases: the next 10 to 15 years. Ann Acad Med Singapore. 2019;48(6):188-194.
2. Sekar SN, Anbarasi LJ, Dhanya D. An Efficient Distributed Compressive Sensing Framework For Reconstruction of Sparse Signals in Mechanical Systems. J Mech Engg Tech. 2018;9(13):1286–1292.
3. SenthilKumar AP, Narendra M, Anbarasi LJ, Raj BE. Breast cancer Analysis and Detection in Histopathological Images using CNN Approach. In Proceedings of International Conference on Intelligent Computing, Information and Control Systems 2021;335-343.
4. World Health Prganisation. Covid-19 Dashboard (accessed on 7th Dec. 2020) WHO Coronavirus Disease (COVID-19) Dashboard | WHO Coronavirus Disease (COVID-19) Dashboard
5. Randhawa GS, Soltysiak MP, El Roz H, de Souza CP, Hill KA, Kari L. Machine learning using intrinsic genomic signatures for rapid classification of novel pathogens: a COVID-19 case study. PloS One. 2020;15(4):e0232391.
6. Guan P, Huang DS, Zhou BS. Forecasting model for the incidence of hepatitis A based on artificial neural network. Wor J Gastrol. 2004;10(24):3579.
7. Katarya R, Rastogi S. A study on neural networks approach to time-series analysis. In2018 2nd International Conference on Inventive Systems and Control (ICISC) 2018;116-119.
8. Sarobin MV, Alphonse S, Gupta M. Joshi T. December. Rapid Eye Movement Monitoring System Using Artificial Intelligence Techniques. Int Conf Inform Manag Mach Intellig. 2019;4:605-610.
9. Sarobin MV. Ganesan R. Swarm intelligence in wireless sensor networks: a survey. Int J Pure Appl Math. 2015;101(5):773-807.
10. Vasudevan S, Chauhan N, Sarobin V. Geetha S. Image-Based Recommendation Engine Using VGG Model. Adv Comm Comp Techn. 2019;13(6):257-265.
11. Chazhoor A, Mounika Y, Sarobin MV, Sanjana MV, Yasashvini R. October. Predictive Maintenance using Machine Learning-Based Classification Models. In IOP Conference Series: Mat Sci Engg. 2020;954(1):012001
12. Tomar A, Gupta N. Prediction for the spread of COVID-19 in India and effectiveness of preventive measures. Sci tot Envt. 2020;728:138762.
13. Siami-Namini S, Tavakoli N, Namin AS. A comparison of ARIMA and LSTM in forecasting time series. In 2018 17th IEEE International Conference on Machine Learning and Applications 2018 Dec 17:1394-1401.
14. Solomatine DP, Shrestha DL. AdaBoost. RT: a boosting algorithm for regression problems. In 2004 IEEE International Joint Conference on Neural Networks 2004;2:1163-1168.
15. Winters PR. Forecasting sales by exponentially weighted moving averages. Mang Sci. 1960;6(3):324-42.
16. Hyndman RJ, Koehler AB, Snyder RD, Grose S. A state space framework for automatic forecasting using exponential smoothing methods. Int J Forecas. 2002;18(3):439-54.
17. Pham QV, Nguyen DC, Hwang WJ, Pathirana PN. Artificial intelligence (AI) and big data for coronavirus (COVID-19) pandemic: A survey on the state-of-the-arts. Mat Sci Engg. 2019;94(1):012002
18. Tuli S, Tuli S, Tuli R, Gill SS. Predicting the growth and trend of COVID-19 pandemic using machine learning and cloud computing. Internet of Things. 2020;11:100222.
19. UdhayaKumar S, Thirumal Kumar D, Christopher BP, Doss C. The Rise and Impact of COVID-19 in India. Front Med (Lausanne). 2020;7:250.
20. Awad M, Khanna R. Support vector regression. Efficient Learning Mach. 2015:67-80.