Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

27

COVID-19 pandemic in India: forecasting using machine learning techniques

H.S. Hota[1], Richa Handa[2], A.K. Shrivas[3]

[1]ATAL BIHARI VAJPAYEE UNIVERSITY, BILASPUR, CHHATTISGARH, INDIA; [2]DR. C.V. RAMAN UNIVERSITY, BILASPUR, CHHATTISGARH, INDIA; [3]GURU GHASIDAS UNIVERSITY, BILASPUR, CHHATTISGARH, INDIA

1. Introduction

The novel coronavirus disease 2019 (COVID-19) has been declared a world pandemic threat by the World Health Organization and is spreading rapidly across the world. It is an infectious disease caused by a virus called a novel coronavirus, which was also the reason for severe acute respiratory syndrome (SARS) in 2003. The SARS epidemic [1,2] was predicted by many authors using various machine learning (ML) techniques. Because of COVID-19, the number of confirmed cases and deaths are rapidly increasing in all most all countries; India has not escaped this pandemic. The first case of COVID-19 emerged in India on Jan. 30, 2020; 2 more cases were found on Feb. 3, 2020 and the number was stable until Mar. 1, 2020. The disease has since spread to most states and cities of India [3]. Few states or cities remain untouched by the COVID-19 outbreak in India. There is no vaccine to treat COVID-19 to prevent infection from one infected person to another. India has learned a lesson from other countries, and therefore the Indian government has made appropriate decisions and implemented various strategies to prevent the pandemic from spreading across the country well in advance. Many approaches were taken to stop spreading cases of COVID-19 from affected to unaffected parts of the country, including a citywide lockdown, closing all transports such as airports, railways, and local transportations, and closing markets, malls, cinemas, productions, and so on. Moreover, the isolation or quarantine of suspected patients is being done. The entire machinery of the government is fully involved to stop spreading it in the community; despite this, cases of COVID-19 positive are increasing every day. By the time of writing, there were more than 26,283 confirmed cases in India, 825 of whom died and about 5938 of whom had recovered, and the numbers keep rising [4].

Data Science for COVID-19. https://doi.org/10.1016/B978-0-12-824536-1.00030-7

Various statistical and mathematical analysis and studies are ongoing to forecast the future trend of COVID-19 in India, and models are being developed to predict the future situation, known as N-days in forecasting. Because of the nonlinear behavior of COVID-19 data, various ML techniques could be useful to develop a robust forecasting model. Research shows that the application of computational intelligence methods is the basis for constructing a predictive model. In this, the neural network is useful for predicting time series data because it has the ability to learn from data and capture the various dynamics of time series data [5]. Evolutionary computations, fuzzy logic, and other models are also crucial owing to their principal differences from existing mathematical approaches. Hybrid models of various intelligent techniques are also widely used in forecasting, because accuracy and efficiency are the most important criteria of focus by researchers [6].

Most work has been done on the basis of either trend already experienced by other countries such as China or statistical theory and analysis. Roosa et al. [7] worked on the real-time forecasting of the COVID-19 pandemic in China and developed models for 5-, 10-, and 15-days ahead forecasting based on the cumulative number of confirmed cases in Hubei and other provinces of China using three different techniques: the Richards model, the subepidemic model, and generalized logistic growth model (GLM). They concluded that each model predicts that the pandemic has reached saturation in Hubei and other provinces of China. Benvenuto et al. [8] performed an autoregressive integrated moving average (ARIMA) on Johns Hopkins epidemiological data to predict the epidemiological trend of the prevalence and incidence of COVID-19. Abdulmajeed et al. [9] proposed an online forecasting mechanism that streams data from the Nigeria Center for Disease Control, which provides updated COVID-19 forecasts every 24 h. The authors combine ARIMA, Prophet (an additive regression model developed by Facebook), and a Holt−Winters exponential smoothing model combined with generalized autoregressive conditional heteroscedasticity. In other work, Tuli et al. [10] applied an improved mathematical model to analyze and predict the growth of the epidemic of COVID-19 and deployed the model on a cloud computing platform for more accurate and real-time prediction of the growth behavior of the epidemic. Ardabili et al. [11] presented a comparative analysis of ML and soft computing models to predict the COVID-19 outbreak and found ML to be an effective tool to model the outbreak. Zhou et al. [12] explained the challenges to geographic information systems (GIS) with big data on COVID-19. Other authors [13] analyzed and forecast COVID-19 spread in China, Italy, and France, and concluded that the infection rate needed to be cut down drastically and quickly to observe an appreciable decrease in the pandemic peak and mortality rate.

Tobías et al. [14] analyzed the trends of incident cases, deaths, and intensive care unit admissions in Italy and Spain before and after their respective national lockdowns using an interrupted time-series design. Data were analyzed with quasi-Poisson regression using an interaction model to estimate the change in trends. Chintalapudi et al. [15] highlighted the importance of lockdown and isolation by forecasting registered and recovered cases of COVID-19 after 60 days' lockdown in Italy adopting a seasonal ARIMA forecasting package with the R statistical model. Koczkodaj et al. [16] predicted the

number of cases of COVID-19 outside China. Their approach was based on a heuristic solution and makes the realistic assumption that the current trend can continue for the next 17 days. Work by Tomar and Gupta [17] studied predicting the spread of COVID-19 in India. This study was made using deep learning techniques. The authors predicted the spread of COVID-19 in the country for next 30 days and suggested prevention measures. Fong et al. [18] presented a case study using composite monte-carlo (CMC) that is enhanced by a deep learning network and fuzzy rule induction to gain better stochastic insights about the development of the epidemic.

According to available resources and published articles, few researchers have worked on forecasting cases of COVID-19, especially for India. India is a diverse and highly populous country, so forecasting cases of COVID-19 is highly uncertain and is itself a nonlinear problem that depends on many factors. This research is an extension of the scarce research already done in this direction, with the objective of measuring the future situation of COVID-19 in India in terms of confirmed and recovered cases and deaths using an ML technique called regression [19]. Comparative analysis among various ML algorithms called estimators was also done. This research will facilitate as assessment of the critical situation in the country and will help the government to make appropriate decisions to optimize and use available health care infrastructure as well as to manage and deploy health personnel in affected areas. The outcome of the proposed research work is the ability to predict 15-day conditions of confirmed and recovered cases and deaths and can be improved for N-days ahead forecasting. Results were compared with two different models developed using cumulative data and daily data obtained elsewhere [20]. Empirical results show that the model built with cumulative data outperforms compared with the model built with daily data with an acceptable range of mean absolute percentage error (MAPE).

2. Material and methods

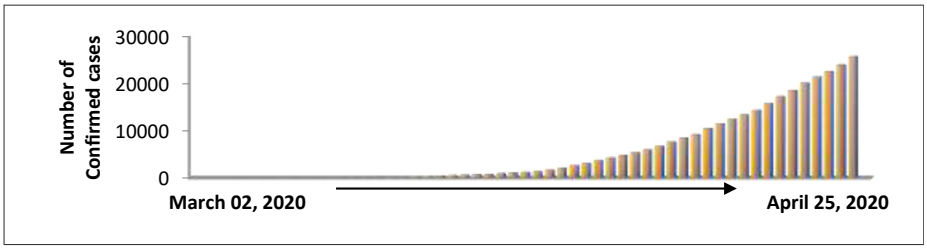
This section present details about the dataset used for ML and the preprocessing stage with a short description of ML algorithms used to develop the ML-based forecasting model.

2.1 Dataset

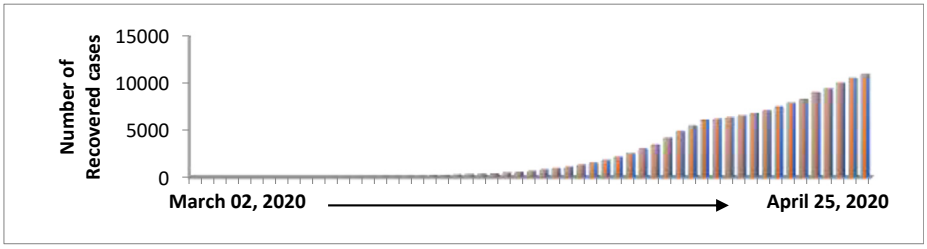
Data are an essential ingredient of ML algorithms. COVID-19—related data are actually time series data collected over a fixed interval of 1 day. However, an ML-based approach always needs a large sample size to train the models, but in the current situation, data are unavailable and it is also equally important to predict the future impact of COVID-19 for the government to make appropriate decisions and manage many other things in the various parts of the country. Details about the data used in this research are shown in Table 27.1, which are collected from www.covid19india.org [20].Two types of data, cumulative and daily, were collected to build forecasting models from Mar. 02, 2020 to Apr. 25, 2020. Figs. 27.1 and 27.2 show bar graphs of collected cumulative and daily data, respectively, for all three cases.

Table 27.1 Description of dataset.

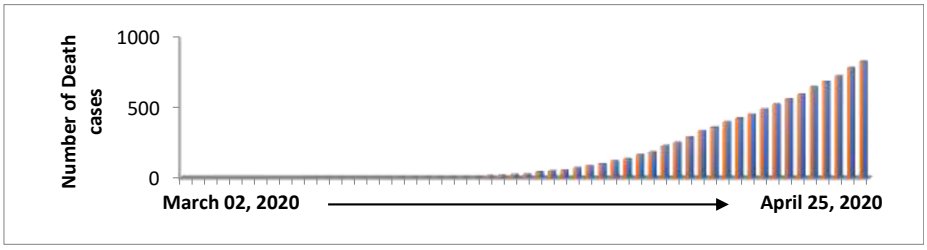
Particular	Detail
Nature of novel coronavirus disease 2019 data	Time series (daily) data: Cumulative and daily
Period	Mar. 2 to Apr. 25, 2020
Total observations, n	55
Source	www.covid19india.org
Data partition	10-fold cross-validation (training/testing)
Training and testing period	Mar. 2 to Apr. 14, 2020 (44 observations)
Validation periods	Apr. 15 to Apr. 25, 2020 (11 observations)



(a)

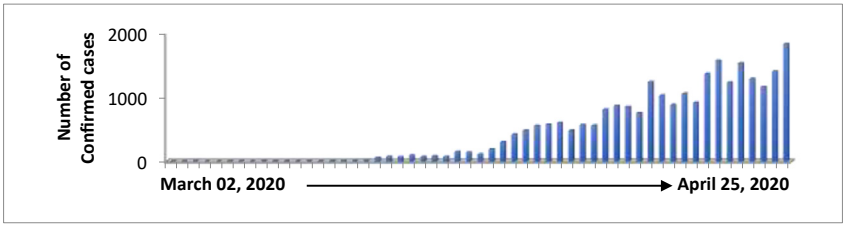


(b)

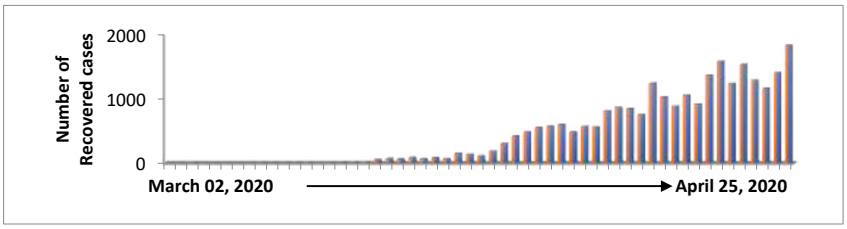


(c)

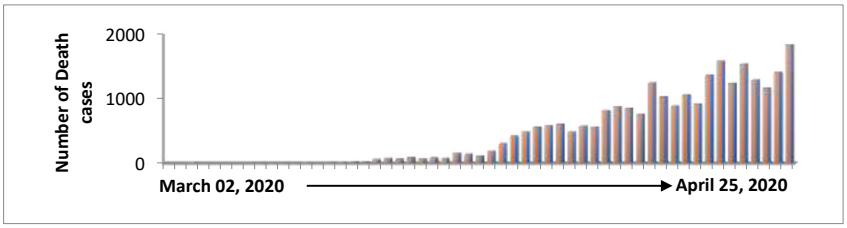
FIGURE 27.1 Number of recorded (A) Confirmed cases, (B) Recovered cases, and (C) Death cases of cumulative data for novel coronavirus disease 2019 in India.



(a)



(b)



(c)

FIGURE 27.2 Number of recorded (A) Confirmed cases, (B) Recovered cases, and (C) Death cases of daily data for novel coronavirus disease 2019 in India.

2.2 Preprocessing

Preprocessing techniques are used to remove inconsistencies from the dataset, which improves the quality of data. It is an essential step, especially in the case of an ML-based forecasting model for the smooth convergence of a learning curve [21]. Preprocessing techniques that were adopted are discussed next.

2.2.1 Normalization

Normalization is the process of smoothing nonlinear data in a scale of [0 1]. The dataset is normalized by dividing each sample with the highest value of the sample data. An equation is used for data normalization, which scales data in the range of [0 1]:

$$X_{new} = \frac{X}{X_{\max}}$$

in which X is the daily number of cases accrued for COVID-19 data, $X_{\max}$ is highest value of number of cases, and X_{new} is obtained normalize data.

2.2.2 Sliding window

Sliding window works well when we have a smaller number of samples for training and testing and accumulate all values of window size. It is a temporary approximation over the actual value of the time series data [22,23]. The size of the window and segment increases until we reached the least error approximation. Sliding window accumulate the historical time series data [24,25] to predict the next-day value. After selecting the first segment, the next segment is selected from the end of the first segment. The process is repeated until all-time series data are segmented. In this research, we use window size = 7 based on previous work and experience in developing ML models.

2.3 Feature extraction and feature selection

A straightforward approach and the most important aspect of the ML technique, called feature extraction and feature selection methods [26], is applied to the dataset [27] to extract features from existing the feature space and select the best features to develop a robust forecasting model. Because COVID-19 time series data consist of only two fields, namely, date and number of cases (confirmed, death, and recovered), to improve performance, we need to generate features based on well-known and tested statistical formulas, as explained in Table 27.2.

In this research, a new feature space was generated with the help of feature extraction. Features are extracted using technical indicators [28] such as moving average, exponential moving average (EMA), weighted moving average, relative strength index, standard deviation, variance, and rate of change, as shown in Table 27.2.

On the other hand, feature selection is a process to find relevant features after removing irrelevant features from the original feature space. In this proposed work, we have selected the best features using a ranking-based feature selection technique. Of eight features, the four best ones (cases [original feature], EMA, standard deviation, and variance) were selected. There reduced feature space data were used to forecast COVID-19—related cases and will also work well compared with original feature space data.

3. Machine learning techniques

A new library of python (PyCaret) provides the bulk of ML techniques. Based on an exhaustive search of ML algorithms, 22 ML algorithms were selected automatically by feeding COVID-19 time series data. Models were trained using 10-fold cross-validation to use all of the samples as training as well as testing owing to the small sample size of data. Many ML techniques works well for forecasting time series data, including Bayesian net, support vector machine, and perceptron, to list a few [24,29,30]. A simple but widely used ML technique is regression [31,32], which is a supervised ML technique that estimates a significant relationship between one dependent variable and two or more

Table 27.2 Description of technical indicators [28].

S.No.	Technical indicator	Formula	Description
1	Moving average (MA)	$\text{MA} = \frac{\sum_{i=0}^{n} x_i}{n}$, where x_i is the current value and n is number of values.	It is the average of time series data.
2	Exponential moving average (EMA)	**Multiplier**: $2/(n+1)$ **EMA**: {V − EMA(previous day)} * multiplier + EMA (previous day), where n is number of values and V is current value.	EMA reduces lag by applying more weight to recent value.
3	Weighted moving average (WMA)	$WMA_t = \frac{\sum_{i=n}^{1} x_i * \left(i-1\right)}{\sum_{i=n}^{1} \frac{(n*(n+1)}{2}}$, where x_i is the current value and n is number of values.	It is moving averages visualize the average value of time-series data over a specific period of time.
4	Relative strength index (RSI)	$RSI = 100 - \frac{100}{1 + \frac{AVGGain}{AVGloss}}$ *AVGGain* and *AVGloss* are the average percentage of gain and loss during look-back period.	It is a momentum indicator that measures the magnitude of recent changes in data.
5	Standard deviation	$\text{SD} = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}\left((x_i - \overline{x})^2\right)}$, where x_i is current value, $\overline{x}$ is mean value of x, and n is number of values.	It calculates the dispersion of a dataset relative to its mean and square root of variance.
6	Variance	$\text{Var} = \frac{\sum\left(\left(x_i - \overline{x}\right)\right)^2}{(n-1)}$, where x_i is current value, $\overline{x}$ is mean value of x, and n is number of values.	It determines the spread of data size compared with the mean value.
7	Rate of change (ROC)	$ROC = \left(\frac{C_p - C_{p-n}}{C_{p-n}}\right) * 100$, where C_p is the value of the most recent period and C_{p-n} represents the value that is n periods before the recent value.	It measures the percent change in data from one period to the next.

independent variables [30]. Regression analysis is a statistical methodology most often used for numeric prediction, although other methods also exist. Regression also encompasses the identification of distribution trends based on available data [27]. The objective of this technique in ML is to predict continuous values such as time series data. ML algorithms developed through statistical learning theory has been widely applied in nonlinear regression estimations. When the regression of Y on X is linear, sometimes the regression line does not pass through the origin. In such conditions, it is more appropriate to use the regression type estimator to estimate the expected value; these estimators are various ML algorithms. Various estimators of regression along with a few basic algorithms have been used here and are listed in Table 27.3.

Table 27.3 Machine learning algorithms (estimators) and their descriptions.

S.No.	Estimator	Description
1	CatBoost regressor	An ML algorithm that uses gradient boosting on decision trees.
2	Extra trees regressor	An extremely randomized tree regressor. It is used only within ensemble methods.
3	Random forest	An ensemble learning method for classification or regression.
4	Light gradient boosting machine	A high-performance gradient boosting framework built on a decision tree algorithm for classification, regression and various other machine learning tasks.
5	Extreme gradient boosting	Used for supervised learning tasks such as regression, classification, and ranking, based on gradient boosting framework.
6	Gradient boosting regressor	Uses the principle of ensemble decision tree regressor models.
7	Decision tree	A nonparametric supervised learning method used to create a model that predicts the value of a target variable by using simple decision rules influenced by the data features.
8	Ridge regression	A way to create a model when the number of predictor variables is more than the number of observations.
9	Lasso regression	A type of linear regression that uses shrinkage when data values are shrunk toward a midpoint, such as the mean.
10	Linear regression	A linear approach that shows the relationship between a dependent variable and one or more independent variables.
11	Least angle regression	A linear regression that selects the model to solve the problem of overfitting.
13	Bayesian ridge	Estimates a probabilistic model of the regression problem.
14	Theil-Sen regressor	Selects the median of the slopes of all lines for robustly fitting a line.
15	Random sample consensus	Works iteratively and estimates the parameter of a mathematical model from a set of observed data that contains outliers.
16	Huber regressor	A regression technique that is robust to outliers.
17	Passive aggressive regressor	Online learning algorithms for both classification and regression.
18	Orthogonal matching pursuit	A greedy compressed sensing recovery algorithm that selects the best-fitting column of the sensing matrix in every iteration.
19	AdaBoost regressor	A machine learning meta-algorithm that can be used in combination with many types of learning algorithms to improve performance.
20	K neighbors regressor	A simple algorithm to predict the target value based on a similarity measure on all available cases.
21	Elastic net	A linear regression model useful with multiple correlated features and likely to pick both.
22	Support vector machine	Supervised learning model associated with multiple learning algorithms for data analysis for classification and regression.

4. Results and discussion

Experimental work was carried out using a Python library (PyCaret). The regression module of PyCaret is a supervised ML module used to forecast continuous values. It has over 22 ML algorithms and various plots to analyze the performance of models.

4.1 Experimental design

A flow diagram of experimental design is depicted in Fig. 27.3, showing the seven main components: data collection, data normalization, feature extraction, feature selection, data partition, model development, and model selection and validation.

As stated in Section 2.1, data for confirmed and recovered cases and deaths were collected. Data are in an integer numeric form and were normalized first to scale between [0 1] and then partitioned into training and testing samples. Owing to the small size of available data, 10-fold cross-validation [33] was used. The 10-fold cross-validation technique is employed for the dynamic partitioning of data and to improve the performance of the model. As stated in Section 2.3, feature extraction and feature selection were also employed. Feature selection was performed by the Python tool, and the four best features (actual data, EMA, standard deviation, and variation) were selected as the best of eight extracted features. ML models were then developed with 20 ML algorithms (estimators), as explained in Section 3, and performance was measured based on the MAPE value, from which the best four models were selected. These were extra tree regressor, extreme gradient boosting, linear regression, and AdaBoost regressor. Finally,

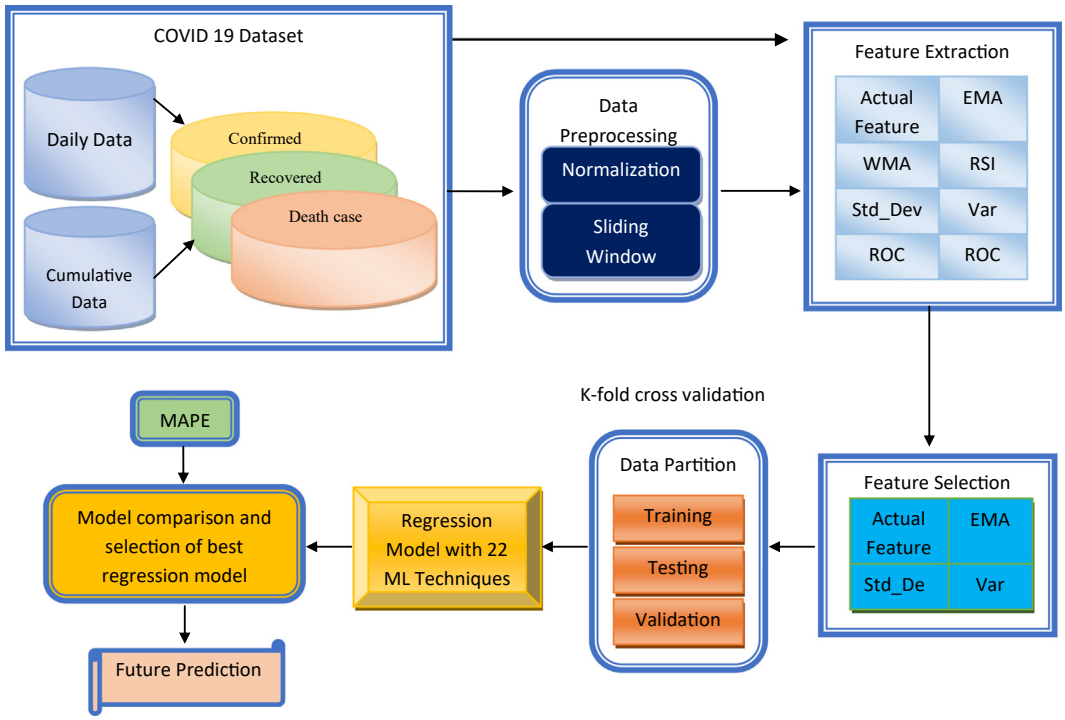


FIGURE 27.3 Process flow diagram for proposed research. *COVID-19*, novel coronavirus disease 2019; *EMA*, exponential moving average; *MAPE*, mean absolute percentage error; *ML*, machine learning; *ROC*, rate of change; *RSI*, relative strength index; *Std—Dev*, standard deviation; *Var*, variance; *WMA*, weighted moving average.

15-day-ahead forecasting was done through the best selected model (i.e., extra trees regressor model). The model was also verified with future data from Apr. 15 to 25, 2020 to verify the robustness of the model and was found to be satisfactory.

4.2 Regression-based model development

Tables 27.4–27.6 show the comparative analysis of these ML models with training, testing, and validation datasets for cumulative and daily datasets for confirmed and recovered cases and deaths in terms of MAPE. The extra tree regressor outperformed in all cases at training and testing as well as the validation stage. MAPE at testing stage was 1.377, 1.302, and 0.488, respectively, for confirmed and recovered cases and deaths based on the model developed with daily observations, whereas it was 0.498, 0.240, and 0.430, respectively, for confirmed and recovered cases and deaths based on the model developed with cumulative observations. This also proves that MAPE is in an acceptable range, especially when the models were validated with actual official data obtained from Apr. 15 to 25, 2020. At the validation stage, the model performed well with a MAPE of 6.262, 7.576, and 6.273 for confirmed and recovered cases and deaths, respectively, based on the model developed with daily observations, while it was 4.123, 5.422, and 4.553 for confirmed and recovered cases and deaths, respectively, based on the model developed with cumulative observations. However, there was a gap between estimated MAPE values (predicted by the model) and observed MAPE values (obtained officially), but it was obviously caused by the nonlinear nature of data as well as the availability of the small sample size of training and testing data.

The tables also show that of 4 ML algorithms (estimators), the extra tree regressor [34] performing best in models developed with cumulative as well as daily datasets in all three cases.

4.3 Performance analysis

The performance of the model was analyzed across different aspects, as discussed next.

Table 27.4 Comparative analysis of training, testing, and validation datasets of novel coronavirus disease 2019 with different estimators in terms of mean absolute percentage error.

		Based on cumulative data			Based on daily data		
S.No.	**Estimator**	**Training**	**Testing**	**Validation**	**Training**	**Testing**	**Validation**
1	Extra tree regressor	0.229	0.498	4.123	0.324	1.377	6.261
2	Extreme gradient boosting	0.432	0.547	5.997	0.475	2.482	8.887
3	Linear regression	0.561	0.750	6.912	0.740	3.111	12.448
4	AdaBoost regressor	0.595	1.072	7.002	1.850	4.482	12.900

Table 27.5 Comparative analysis of training, testing, and validation datasets of novel coronavirus disease 2019 with different estimators in terms of mean absolute percentage error.

		Based on cumulative data			Based on daily data		
S.No.	Estimator	Training	Testing	Validation	Training	Testing	Validation
1	Extra tree regressor	0.135	0.240	5.411	0.393	1.302	7.576
2	Extreme gradient boosting	0.150	0.325	6.577	0.452	1.151	8.728
3	Linear regression	0.185	0.337	8.094	0.979	1.679	13.173
4	AdaBoost regressor	0.329	0.401	8.803	0.995	2.240	13.609

Table 27.6 Comparative analysis of training, testing and validation datasets of novel coronavirus disease 2019 with different estimators in terms of mean absolute percentage error.

		Based on cumulative data			Based on daily data		
S.No.	Estimator	Training	Testing	Validation	Training	Testing	Validation
1	Extra tree regressor	0.229	0.430	4.553	0.290	0.488	6.273
2	Extreme gradient boosting	0.404	0.512	6.209	0.519	0.706	7.818
3	Linear regression	0.491	0.568	8.333	0.819	0.804	9.663
4	AdaBoost regressor	0.654	0.822	11.827	0.852	0.936	10.862

4.3.1 Model validation

According to the results at the validation stage shown in the tables in Section 4.2, a comparison was made, as shown in Table 27.7–27.9 of actual (official) data as reported and predicted through the best model along with the percent error. These are also shown graphically in the form of bar graphs in Fig. 27.4.

4.3.2 Model analysis by plotting

Analysis of the model was also conducted graphically through residuals graphs and prediction error plotting. Plotting takes a trained model object and returns a plot based on the test dataset, as shown in Figs. 27.5–27.10.

A prediction error plot depicts actual targets against predicted values generated by our model. This shows the variance in the model. Using this plotting, we can diagnose regression models by comparing them against the slanting line of 45 degrees and identify whether the prediction exactly matches the model. A residual plot is a graphical representation that shows the relationship between a given independent variable and the response variable. A residual value is a measure of how much a regression lines best fits the dataset where few data points will fit the line and others will miss. In the residual plot, the X axis represents the residual values and the Y axis displays the independent variable.

Table 27.7 Actual data (officially recorded) and predicted data for confirmed cases.

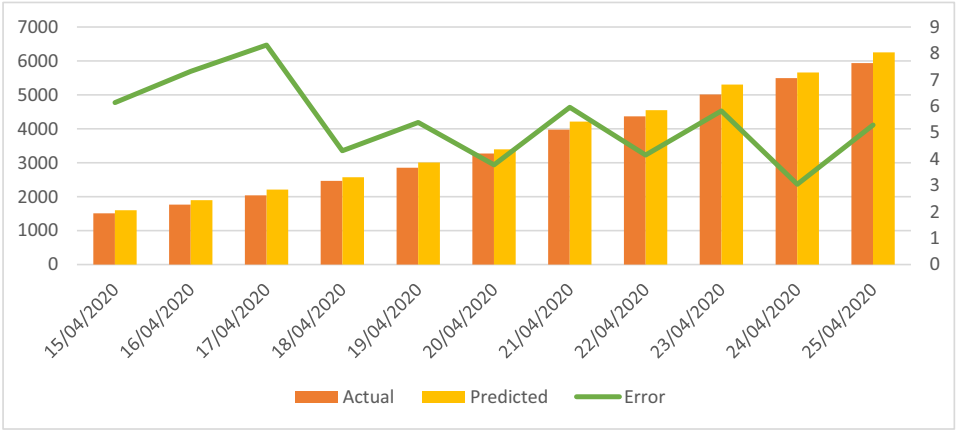
	Based on cumulative data			Based on daily data		
Date	**Actual**	**Predicted**	**Error (%)**	**Actual**	**Predicted**	**Error (%)**
Apr. 15, 2020	12,370	12,880	6.053	886	941	6.213
Apr. 16, 2020	13,431	14,244	4.324	1061	1108	4.462
Apr. 17, 2020	14,353	14,973	3.566	922	989	7.301
Apr. 18, 2020	15,724	16,284	5.598	1371	1449	5.659
Apr. 19, 2020	17,304	18,272	3.152	1580	1678	6.214
Apr. 20, 2020	18,543	19,127	4.628	1239	1296	4.608
Apr. 21, 2020	20,080	21,009	2.039	1537	1663	8.191
Apr. 22, 2020	21,372	21,807	4.712	1292	1375	6.394
Apr. 23, 2020	23,039	24,125	3.187	1167	1244	6.639
Apr. 24, 2020	24,447	25,226	5.469	1408	1490	5.844
Apr. 25, 2020	26,282	27,719	2.632	1835	1971	7.423

Table 27.8 Actual data (officially recorded) and predicted data for recovered cases.

	Based on cumulative data			Based on daily data		
Date	**Actual**	**Predicted**	**Error (%)**	**Actual**	**Predicted**	**Error (%)**
Apr. 15, 2020	1509	1602	6.139	144	156	8.538
Apr. 16, 2020	1767	1896	7.324	258	278	7.936
Apr. 17, 2020	2040	2210	8.318	273	291	6.456
Apr. 18, 2020	2466	2572	4.313	426	447	4.926
Apr. 19, 2020	2854	3008	5.387	388	421	8.452
Apr. 20, 2020	3273	3397	3.781	419	451	7.644
Apr. 21, 2020	3976	4213	5.958	703	764	8.735
Apr. 22, 2020	4370	4551	4.152	394	423	7.353
Apr. 23, 2020	5012	5304	5.821	642	686	6.833
Apr. 24, 2020	5496	5663	3.042	484	528	9.109
Apr. 25, 2020	5939	6253	5.289	442	474	7.356

Table 27.9 Actual data (officially recorded) and predicted data for deaths.

	Based on cumulative data			Based on daily data		
Date	**Actual**	**Predicted**	**Error (%)**	**Actual**	**Predicted**	**Error (%)**
Apr. 15, 2020	422	440	4.309	27	29	6.072
Apr. 16, 2020	448	460	2.758	26	27	3.246
Apr. 17, 2020	486	500	2.826	38	40	4.295
Apr. 18, 2020	521	550	5.636	35	37	5.277
Apr. 19, 2020	559	579	3.656	38	39	3.295
Apr. 20, 2020	592	620	4.713	33	35	4.692
Apr. 21, 2020	645	683	5.912	53	56	6.397
Apr. 22, 2020	681	710	4.263	36	38	4.373
Apr. 23, 2020	721	766	6.239	40	41	2.607
Apr. 24, 2020	780	815	4.493	59	62	4.362
Apr. 25, 2020	824	867	5.285	44	46	5.472



(a)

(b)

(c)

FIGURE 27.4 Bar graph comparing actual and predicted data along with error for (A) Confirmed and (B) Recovered cases and (C) Deaths.

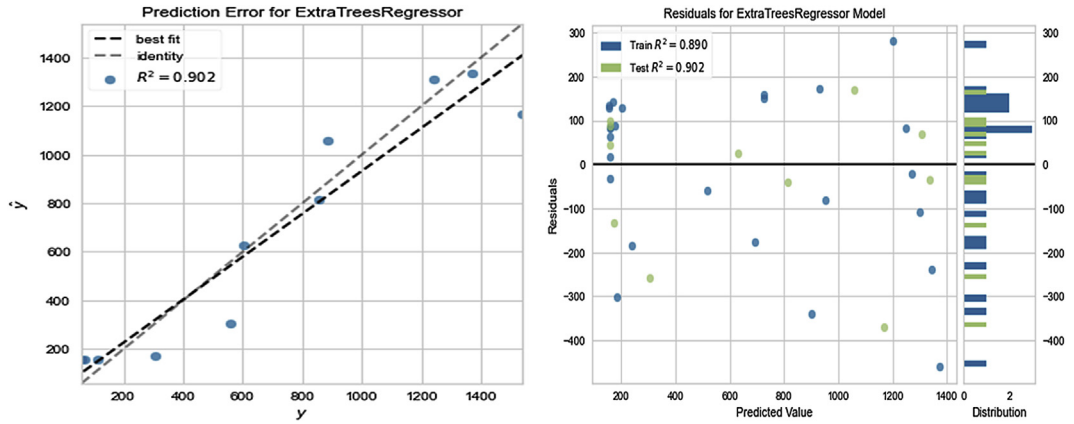


FIGURE 27.5 Confirmed cases in daily dataset: (A) Prediction error for extra Tree regressor (*left*); (B) Residual for extra tree regressor (*right*).

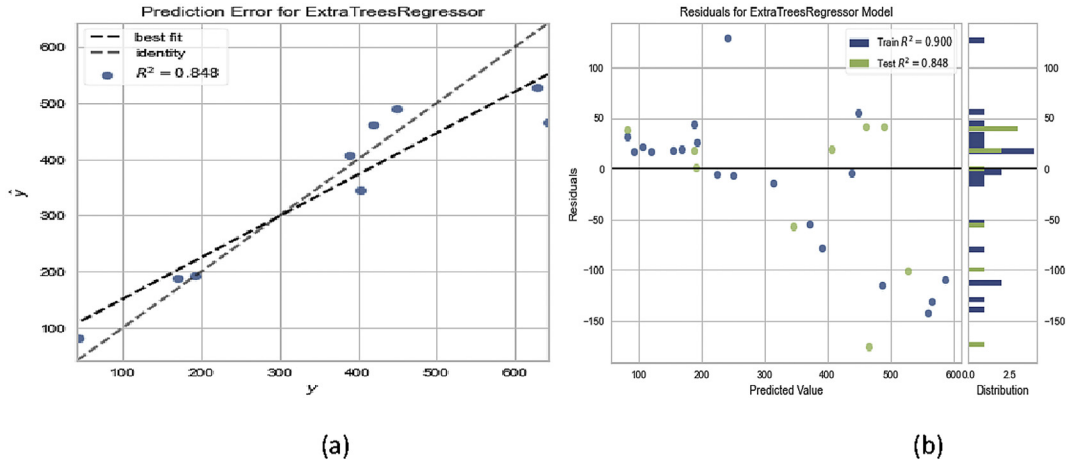


FIGURE 27.6 Recovered cases in daily dataset: (A) Prediction error for extra tree regressor (*left*); (B) Residual for extra tree regressor (*right*).

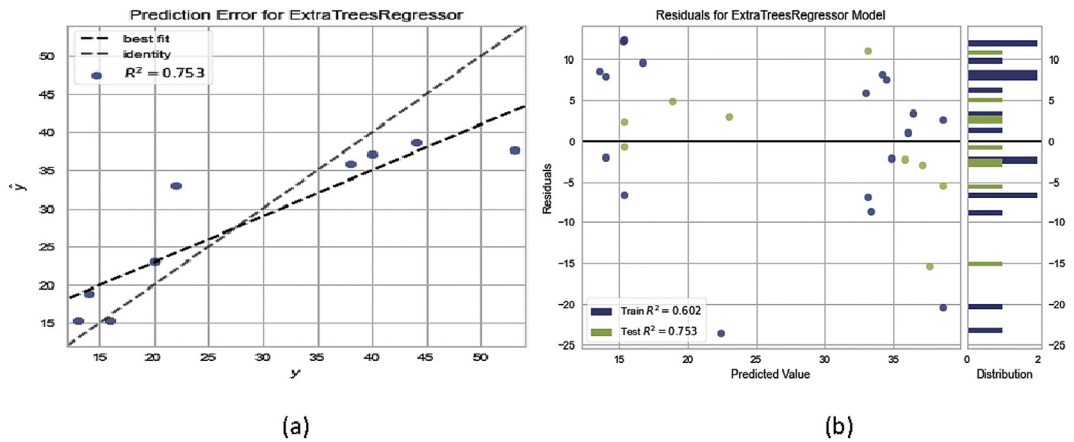


FIGURE 27.7 Deaths in daily dataset: (A) Prediction error for extra tree regressor (*left*); (B) Residual for extra tree regressor (*right*).

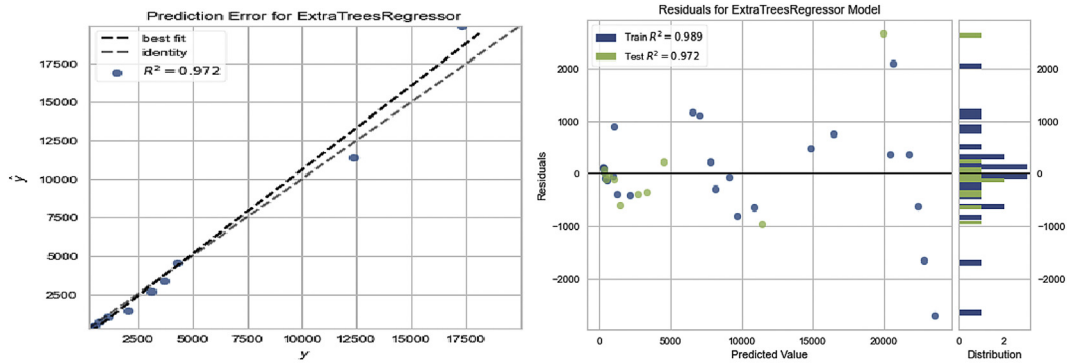


FIGURE 27.8 Confirmed cases in cumulative dataset: (A) Prediction error for extra tree regressor (*left*); (B) Residual for extra tree regressor (*right*).

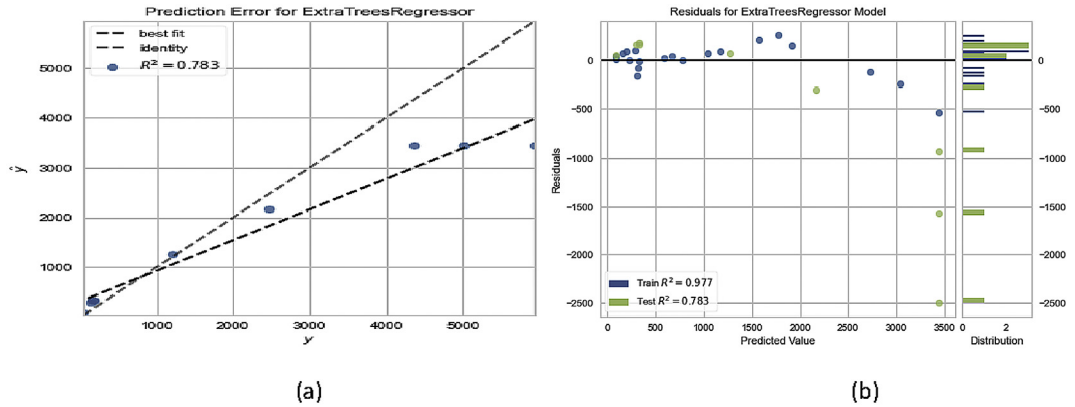


FIGURE 27.9 Recovered cases in cumulative dataset: (A) Prediction error for extra tree regressor (*left*); (B) Residual for extra tree regressor (*right*).

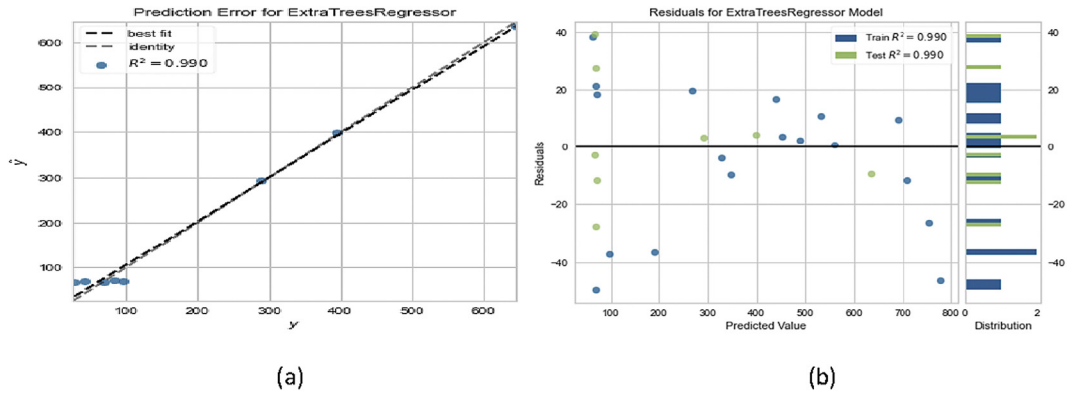


FIGURE 27.10 Deaths in cumulative dataset: (A) Prediction error for extra tree regressor (*left*); (B) Residual for extra tree regressor (*right*).

4.3.3 N-days-ahead forecasting of novel coronavirus disease 2019 pandemic

The primary objective of the proposed research is to predict future values of confirmed and recovered cases and deaths in India. The model was used to predict 15-day-ahead forecasting from Apr. 26 to May 10, 2020 using both models. The results are shown in Figs. 27.11–27.13, respectively, for confirmed and recovered cases and deaths. Figs. 27.11A, 27.12A, and 27.13A are smooth compared with Figs. 27.11B, 27.12B, and 27.13B because the previous figures are based on cumulative data and produce better results for confirmed and recovered cases and deaths. The first part of the graphs compare actual and predicted data at the testing stage, whereas the second part compares actual recorded data and predicted data at the validation stage and thee third part

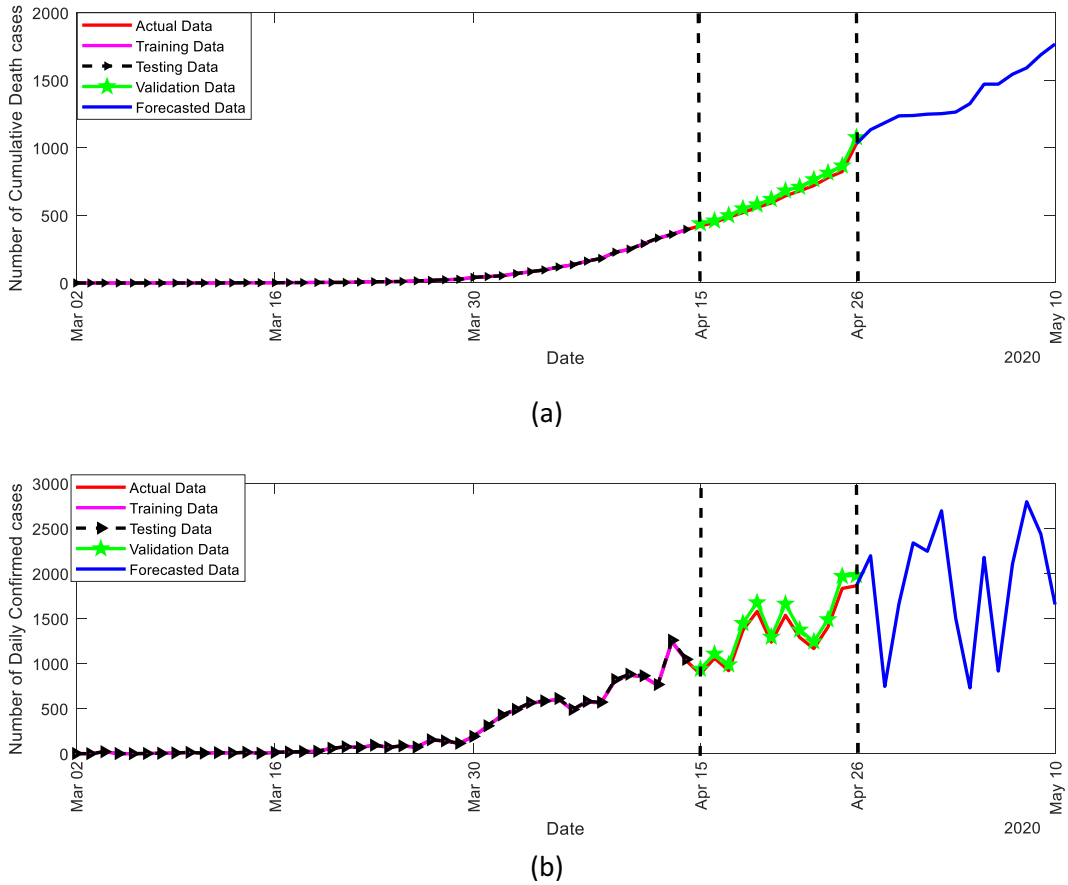
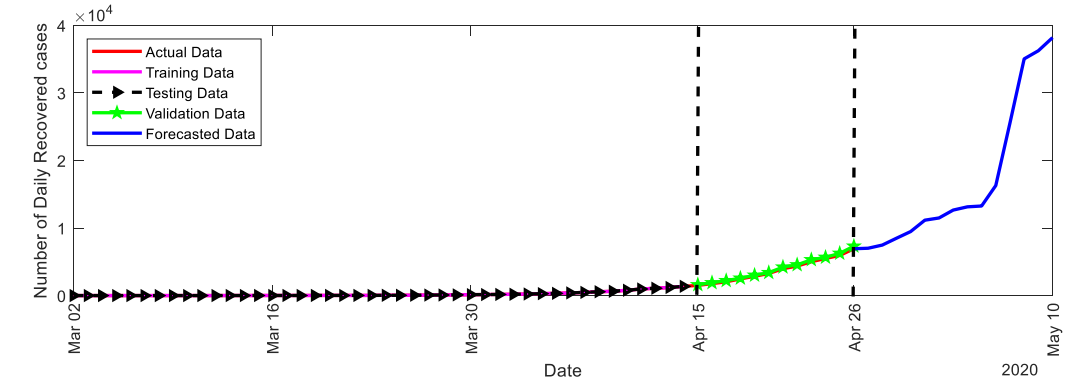
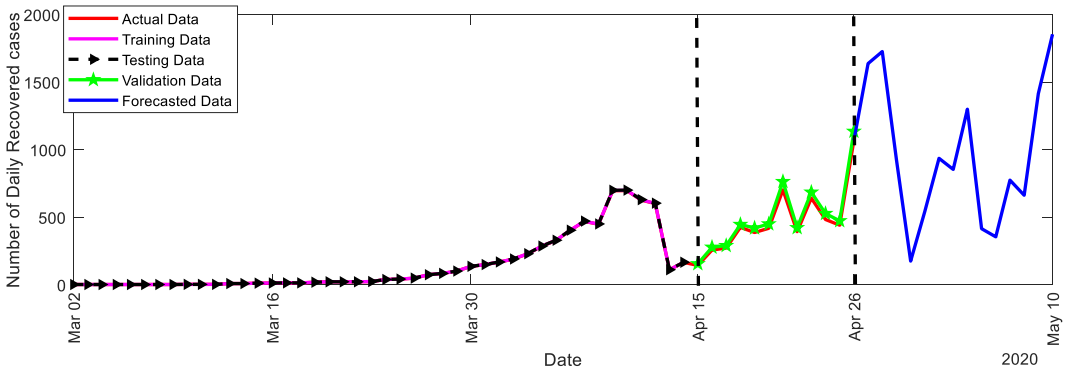


FIGURE 27.11 Fifteen-day-ahead forecast of novel coronavirus disease 2019 pandemic. Confirmed cases based on (A) Cumulative data; (B) Daily data.



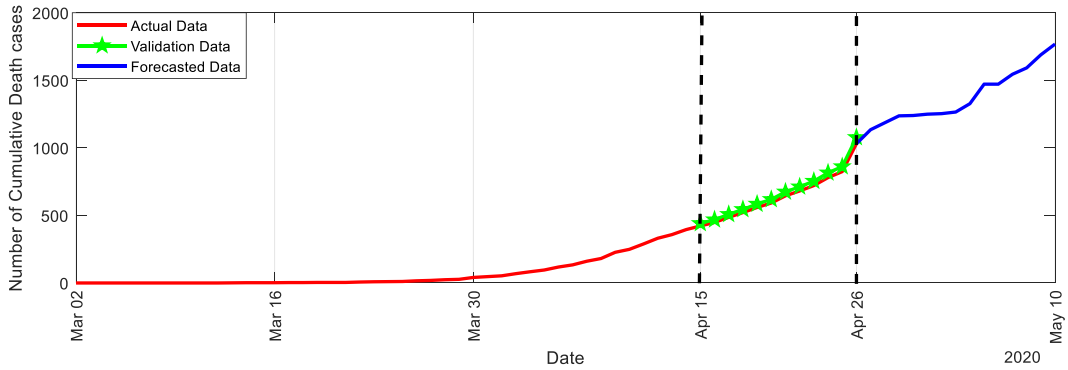
(a)



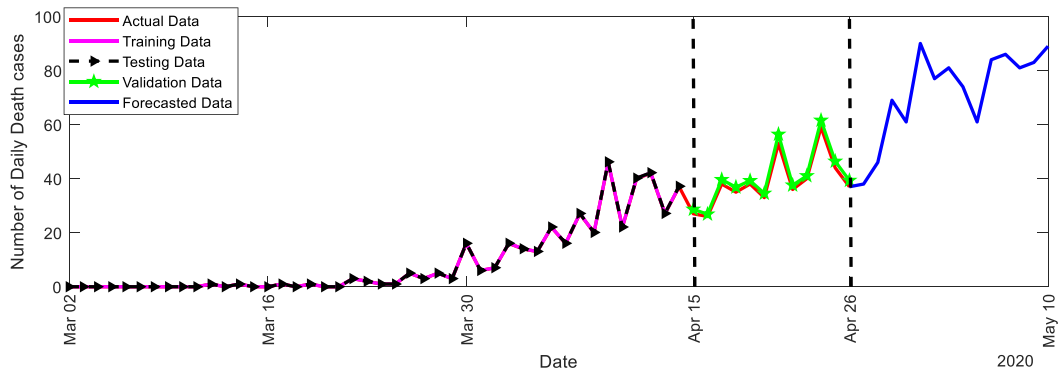
(b)

FIGURE 27.12 Fifteen-day-ahead forecast of novel coronavirus disease 2019 pandemic. Recovered cases based on (A) Cumulative data; (B) Daily data.

shows the 15-day-ahead forecast. The actual data almost overlaps the predicted data at the testing and validation stages. The forecasted data indicate the 15-day-ahead forecasted values from Apr. 26 to May 10, 2020, which can be verified in a real sense as the time arrives. These graphical results indicate that cases will have increase in the near future and will have reached 72,028 confirmed cases, 38,178 recovered cases, and 1768 deaths by May 10, 2020 (shown in Fig. 27.14 for all three cases). Fig. 27.14 depicts a pandemic graph for confirmed and recovered cases and deaths in the case of forecasted models developed with both datasets. Fig. 27.14A shows that the number of confirmed cases increases rapidly; on the other hand, because of available medical facilities and



(a)



(b)

FIGURE 27.13 Fifteen-day-ahead forecast of novel coronavirus disease 2019 pandemic. Deaths based on model built with (A) Cumulative data; (B) Daily data.

special attention given to treating COVID-19 patients, recovered cases are subsequently increasing and deaths are almost stable for the period of forecasting (i.e., Apr. 26, 2020 to May 10, 2020). The same trend is observed in the case of a model developed with daily data, as shown in Fig. 27.14B, but with some more variations. The comparative results of both models are shown in Fig. 27.15A and B for confirmed and recovered cases and deaths.

Because validation of the model was done on the basis of data during lockdown, when the sufficient precautions are being taken by Indian government, hence forecasting is biased and binding upon the situation of lockdown. Though more detailed

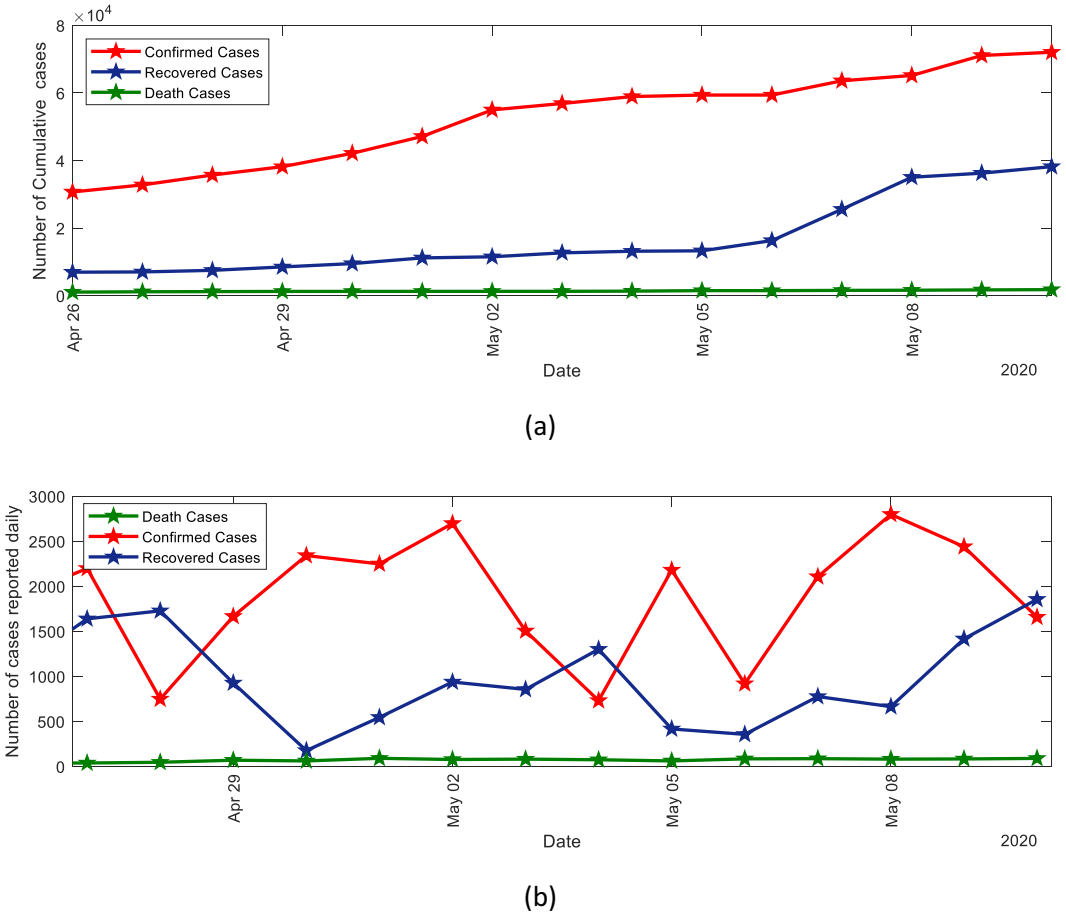


FIGURE 27.14 Fifteen-day-ahead forecast of novel coronavirus disease 2019 pandemic. Confirmed and recovered cases and deaths based on the model developed using (A) Cumulative data and (B) Daily data.

data is needed to improve the forecasting of model for COVID-19 [8]. It may slightly change if any other decisions about lockdown will be taken by the government in near future. Also, the growth of this pandemic will be slowed in recent days only when people take precautions according to advisories that are issued by the Ministry of Health and Family Welfare of the Government of India from time to time. At this point, it is advisable for the lockdown to continue or for limited privileges to be given in specific areas of the country to fight COVID-19.

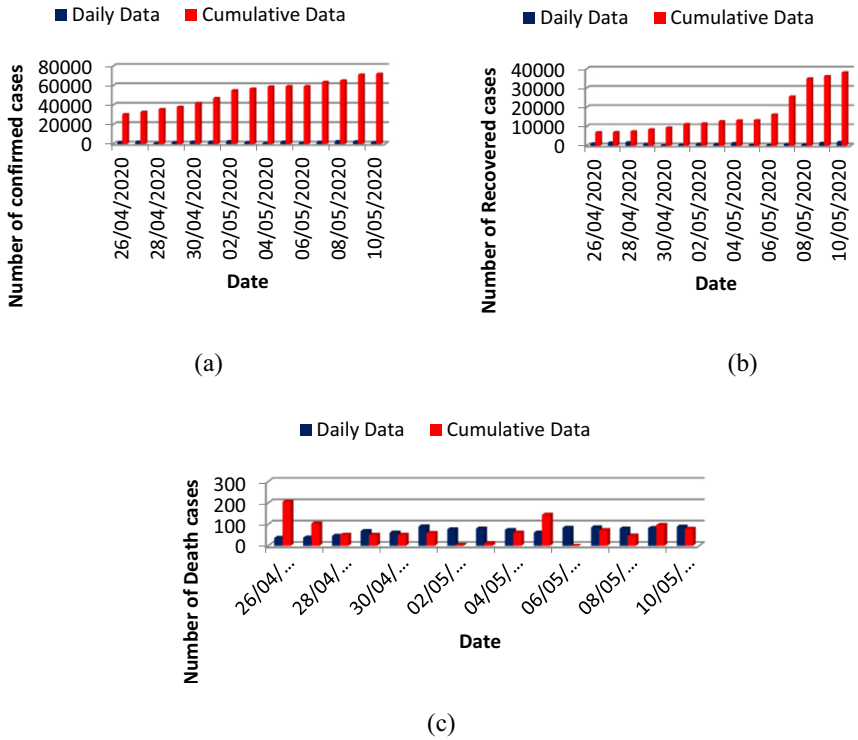


FIGURE 27.15 Comparative graph of models developed with cumulative and daily data for (A) Confirmed and (B) Recovered cases; and (C) Deaths.

5. Conclusion

Forecasting the future trend of COVID-19 pandemic is a critically important job for researchers to assist the government in making appropriate decisions to protect human lives. This research explores the development of ML-based models for 15-day-ahead forecasting based on cumulative and daily data for confirmed and recovered cases and deaths reported in India. Traditional statistical techniques are unable to incorporate variations of data of a nonlinear nature despite the application of statistical and mathematical techniques for the future value of forecasting COVID-19. This study used ML-based techniques to build forecasting models because these techniques are self-capable of producing comparatively better outputs. Regression techniques to develop robust forecasting models with the concept of the sliding window, feature extraction, and feature selection were used. The findings forecast COVID-19 in India with the models developed using cumulative and daily datasets for three cases, showing that confirmed and recovered cases and deaths will increase in the near future. Model validation for data from Apr. 15 to 25, 2020 show that the model works well with an acceptable range of MAPE values. The nature of training data as linear or nonlinear has an important role in

building ML models. Cumulative data are linear in nature whereas daily data are slightly nonlinear in nature; therefore, the developed model using cumulative data outperforms the model developed with daily data. Despite having limited data for training, the models perform well. It is expected that this research work will provide a basis for the development of more an accurate forecasting model and that it might also be used to arrive at a strategy to control the COVID-19 pandemic. In the future, ML models with a greater amount of data can be developed and forecasting might be done for other countries with ML techniques such as deep learning. Also, long-term furcating for 30−60 days might be performed.

References

[1] E. Massad, M.N. Burattini, L.F. Lopez, F.A.B. Coutinho, Forecasting versus projection models in epidemiology: the case of the SARS epidemics, Med. Hypotheses [Internet] (2005) 17−22. Available from: http://doi.org/10.1016/j.mehy.2004.09.029.

[2] Y. Bai, Z. Jin, Prediction of SARS epidemic by BP neural networks with online prediction strategy, Chaos Solitons Fractals [Internet] 26 (2) (2005) 559−569. Available from: http://10.0.3.248/j.chaos.2005.01.064.

[3] B. Krishnakumar, S. Rana, COVID 19 in India: strategies to combat from combination threat of life and livelihood, J. Microbiol. Immunol. Infect. [Internet] 53 (3) (2020). Available from: http://doi.org/10.1016/j.jmii.2020.03.024.

[4] Ministry of Health and Family Welfare, GoI, 2020 [Internet]. Available from: https://www.mohfw.gov.in/.

[5] S. Galeshchuk, Neural networks performance in exchange rate prediction, Neurocomputing [Internet] 172 (2016) 446−452. Available from: http://doi.org/10.1016/j.neucom.2015.03.100.

[6] N.H. Zainuddin, M.S. Lola, M.A. Djauhari, F. Yusof, M.N.A. Ramlee, A. Deraman, et al., Improvement of time forecasting models using a novel hybridization of bootstrap and double bootstrap artificial neural networks, Appl. Soft Comput. J. [Internet] 84 (2019) 105676. Available from: http://doi.org/10.1016/j.asoc.2019.105676.

[7] K. Roosa, Y. Lee, R. Luo, A. Kirpich, R. Rothenberg, J.M. Hyman, et al., Real-time forecasts of the COVID-19 epidemic in China from February 5th to February 24th, 2020, Infect. Dis. Model. [Internet] 5 (2020) 256−263. Available from: http://doi.org/10.1016/j.idm.2020.02.002.

[8] D. Benvenuto, M. Giovanetti, L. Vassallo, S. Angeletti, M. Ciccozzi, Application of the ARIMA model on the COVID-2019 epidemic dataset, Data Brief [Internet] 29 (2020) 105340. Available from: http://doi.org/10.1016/j.dib.2020.105340.

[9] K. Abdulmajeed, M. Adeleke, L. Popoola, Online Forecasting of Covid-19 Cases in Nigeria Using Limited Data, Data in Brief [Internet], 2020, p. 105683. Available from: http://doi.org/10.1016/j.dib.2020.105683.

[10] S. Tuli, S. Tuli, R. Tuli, S.S. Gill, Predicting the Growth and Trend of COVID-19 Pandemic using Machine Learning and Cloud Computing, Internet of Things [Internet], 2020, p. 100222. Available from: http://doi.org/10.1016/j.iot.2020.100222.

[11] S.F. Ardabili, A. Mosavi, P. Ghamisi, F. Ferdinand, A.R. Varkonyi-Koczy, U. Reuter, et al., COVID-19 outbreak prediction with machine learning, SSRN Electron. J. [Internet] 8 (2020 April). Available from: http://doi.org/10.2139/ssrn.3580188.

[12] C. Zhou, F. Su, T. Pei, A. Zhang, Y. Du, B. Luo, et al., COVID-19: challenges to GIS with big data, Geogr. Sustain. [Internet] 1 (1) (2020 March) 77–87. Available from: http://doi.org/10.1016/j.geosus.2020.03.005.

[13] D. Fanelli, F. Piazza, Analysis and forecast of COVID-19 spreading in China, Italy and France, Chaos Solitons Fractals [Internet] 134 (2020) 109761. Available from: http://doi.org/10.1016/j.chaos.2020.109761.

[14] A. Tobías, Evaluation of the lockdowns for the SARS-CoV-2 epidemic in Italy and Spain after one month follow up, Sci. Total Environ. [Internet] 725 (2020) 138539. Available from: http://doi.org/10.1016/j.scitotenv.2020.138539.

[15] N. Chintalapudi, G. Battineni, F. Amenta, COVID-19 virus outbreak forecasting of registered and recovered cases after sixty day lockdown in Italy: a data driven model approach, J. Microbiol. Immunol. Infect. [Internet] 53 (3) (2020) 396–403. Available from: http://doi.org/10.1016/j.jmii.2020.04.004.

[16] W.W. Koczkodaj, M.A. Mansournia, W. Pedrycz, A. Wolny-Dominiak, P.F. Zabrodskii, D. Strzaška, et al., 1,000,000 cases of COVID-19 outside of China: the date predicted by a simple heuristic, Glob. Epidemiol. [Internet] (2020) 100023. Available from: http://doi.org/10.1016/j.gloepi.2020.100023.

[17] A. Tomar, N. Gupta, Prediction for the spread of COVID-19 in India and effectiveness of preventive measures, Sci. Total Environ. [Internet] (2020) 728. Available from: http://doi.org/10.1016/j.scitotenv.2020.138762.

[18] S.J. Fong, G. Li, N. Dey, R.G. Crespo, E. Herrera-Viedma, Composite Monte Carlo decision making under high uncertainty of novel coronavirus epidemic using hybridized deep learning and fuzzy rule induction, Appl. Soft Comput. [Internet] 93 (2020) 106282 (December 2019) Available from: http://doi.org/10.1016/j.asoc.2020.106282.

[19] B. Weng, L. Lu, X. Wang, F.M. Megahed, W. Martinez, Predicting short-term stock prices using ensemble methods and online data sources, Expert Syst. Appl. [Internet] 112 (2018) 258–273. Available from: http://doi.org/10.1016/j.eswa.2018.06.016.

[20] Coronavirus Outbreak in India [Internet]. Available from: www.covid19india.org.

[21] N.M. Nawi, W.H. Atomi, M.Z. Rehman, The effect of data pre-processing on optimized training of artificial neural networks, Proc. Technol. [Internet] 11 (2013) 32–39. Available from: http://doi.org/10.1016/j.protcy.2013.12.159.

[22] Y. BenYahmed, A. Abu Bakar, A. RazakHamdan, A. Ahmed, S.M.S. Abdullah, Adaptive sliding window algorithm for weather data segmentation, J. Theor. Appl. Inf. Technol. 80 (2) (2015) 322–333.

[23] M. Vafaeipour, O. Rahbari, M.A. Rosen, F. Fazelpour, P. Ansarirad, Application of sliding window technique for prediction of wind velocity time series, Int. J. Energy Environ. Eng. 5 (2–3) (2014) 1–7.

[24] L. Mozaffari, A. Mozaffari, N.L. Azad, Vehicle speed prediction via a sliding-window time series analysis and an evolutionary least learning machine: a case study on San Francisco urban roads, Eng. Sci. Technol. Int. J. [Internet] 18 (2) (2015) 150–162. Available from: http://doi.org/10.1016/j.jestch.2014.11.002.

[25] T. Xiong, Y. Bao, Z. Hu, Multiple-output support vector regression with a firefly algorithm for interval-valued stock price index forecasting, Knowledge-Based Syst. [Internet] 55 (2014) 87–100. Available from: http://doi.org/10.1016/j.knosys.2013.10.012.

[26] C.E. Galván-Tejada, L.A. Zanella-Calzada, H. Gamboa-Rosales, J.I. Galván-Tejada, N.M. Chávez-Lamas, M.D.C. Gracia-Cortés, et al., Depression episodes detection in unipolar and bipolar patients: a methodology with feature extraction and feature selection with genetic algorithms using activity motion signal as information source, Mobile Inf. Syst. [Internet] (2019) 1–12. Available from: http://doi.org/10.1155/2019/8269695.

[27] J. Han, M. Kamber, J. Pei, Data Mining Concepts and Techniques, Third., Elsevier, USA, 2012, pp. 451–452.

[28] R. Handa, H.S. Hota, S.R. Tandan, Stock market prediction with various technical indicators using neural network techniques, Int. J. Res. Appl. Sci. Eng. Technol. 3 (1) (2015) 604−608.

[29] B. Leo, Bagging predictors, Mach. Learn. [Internet] 24 (2) (1996) 123−140. Available from: http://doi.org/10.1007/BF00058655.

[30] D.K. Sharma, H.S. Hota, R. Handa, Prediction of foreign exchange rate using regression techniques, Rev. Bus. Technol. Res. 14 (1) (2017) 29−33.

[31] P. Jiang, J. Chen, Displacement prediction of landslide based on generalized regression neural networks with K-fold cross-validation, Neurocomputing [Internet] 198 (2016), 40−7. Available from: http://doi.org/10.1016/j.neucom.2015.08.118.

[32] H.L. Siew, M.J. Nordin, Regression techniques for the prediction of stock price trend, in: ICSSBE 2012 − Proceedings, 2012 International Conference on Statistics in Science, Business and Engineering: Empowering Decision Making with Statistical Sciences, 2012, pp. 99−103.

[33] T.T. Wong, Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation, Pattern Recognit. [Internet] 48 (9) (2015) 2839−2846. Available from: http://doi.org/10.1016/j.patcog.2015.03.009.

[34] M.W. Ahmad, J. Reynolds, Y. Rezgui, Predictive modelling for solar thermal energy systems: a comparison of support vector regression, random forest, extra trees and regression trees, J. Clean. Prod. [Internet] 203 (2018), 810−21. Available from: http://doi.org/10.1016/j.jclepro.2018.08.207.