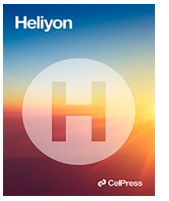
Heliyon 10 (2024) e25746

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon



Research article

Random forest regression for prediction of Covid-19 daily cases and deaths in Turkey

Figen Özen

Department of Electrical and Electronics Engineering, Haliç University, Istanbul, Turkey

ARTICLE INFO

Keywords:
Covid-19 pandemic
Machine learning
Ensemble learning
Random forest regressor
Bagging regressor
Boosting regressor
LSTM
ARIMA

ABSTRACT

During pandemic periods, there is an intense flow of patients to hospitals. Depending on the disease, many patients may require hospitalization. In some cases, these patients must be taken to intensive care units and emergency interventions must be performed. However, finding a sufficient number of hospital beds or intensive care units during pandemic periods poses a big problem. In these periods, fast and effective planning is more important than ever. Another problem experienced during pandemic periods is the burial of the dead in case the number of deaths increases. This is also a situation that requires due planning. We can learn some lessons from Covid 19 pandemic and be prepared for the future ones. In this paper, statistical properties of the daily cases and daily deaths in Turkey, which is one of the most affected countries by the pandemic in the World, are studied. It is found that the characteristics are nonstationary. Then, random forest regression is applied to predict Covid-19 daily cases and deaths. In addition, seven other machine learning models, namely bagging, AdaBoost, gradient boosting, XGBoost, decision tree, LSTM and ARIMA regressors are built for comparison. The performance of the models are measured using accuracy, coefficient of variation, root-mean-square score and relative error metrics. When random forest regressors are employed, test data related to daily cases are predicted with an accuracy of 92.30% and with an r^2 score of 0.9893. Besides, daily deaths are predicted with an accuracy of 91.39% and with an r^2 score of 0.9834. The closest rival in predictions is the bagging regressor. Nevertheless, the results provided by this algoritm changed in different runs and this fact is shown in the study, as well. Comparisons are based on test data. Comparisons with the earlier works are also provided.

1. Introduction

According to the data provided by World Health Organization, as of December 20, 2023, with 772,838,745 confirmed cases and 6,988,679 deaths, Covid-19, the biggest pandemic of the twenty-first century so far, has been effective in Turkey since early 2020. In the statistics of the World Health Organization, the number of confirmed cases for Turkey on December 20, 2023 was 17,004,677 and the number of deaths was 101,419. With these figures, Turkey ranks 12th country in the world in terms of the number of cases, and 20th country in terms of the number of deaths [1]. Covid-19 still continues to produce variants and, according to estimates, it does not seem possible to stop the spread and the deaths caused by it any time soon.

Support for diagnosis and prediction with machine learning methods is increasingly accepted in the medical world and many studies can be found in the literature [2–8], and [9]. Same approach can be applied to analyze the Covid-19 pandemic.

E-mail address: figenozen@halic.edu.tr.

https://doi.org/10.1016/j.heliyon.2024.e25746
Received 22 December 2023; Received in revised form 18 January 2024; Accepted 1 February 2024
Available online 8 February 2024

Since the beginning of the Covid-19 pandemic, various studies have been done related to the diagnosis and prediction of the future of the pandemic, based on time-series, image processing and mixed methods.

Image based machine learning is used extensively due to an abundance of image data since the beginning of the pandemic. Multi-view machine learning technique is used in Ref. [10] and 95.5% accuracy is obtained. With the convolutional support estimator network, 96.5% accuracy is accomplished [11]. Deep convolutional and recurrent neural networks result in 86.7% accuracy [12]. A less common method, namely, uncertainty vertex-weighted hypergraph learning achieves 89.8% accuracy [13]. The convolutional neural network, guided whale optimization, particle swarm optimization combination achieves 99.5% area under curve (AUC) [14]. A convolutional neural network is designed for the diagnosis of Covid-19 quickly in the chest X-ray images. The hyperparameter tuning is achieved using genetic algorithm and 5G technology is utilized. The accuracy obtained is 98.48% [15]. With the exploitation of attention-based deep 3D multiple instance learning, 97.9% accuracy, 99.0% AUC are obtained [16]. The obvious choice for image processing is convolutional neural network and its application results in 98.05% accuracy, 99.66% AUC [17]. Siamese network is used in Ref. [18], yielding 95.6% accuracy. In Ref. [19] ensemble deep convolutional neural network results in 99.7% accuracy. Explainable artificial intelligence with pseudo-labeling and consistency regularization yields 75.13% accuracy [20].

On the other hand, time-series data was scarce at the beginning of the pandemic but more data accumulated as time progressed. Sparse ensemble regressor is applied to time-series data in Ref. [21], but the results are given in terms of root mean square error (rmse) and mean absolute error (mae). These values don't provide a reasonable means to compare a variety of algorithms using different datasets, thus they are not repeated here. With the application of a neural network, 89.98% accuracy is obtained. Long short-term memory (LSTM) network is used in Ref. [22], providing no metric to compare. Bidirectional LSTM yields an r^2 score of 0.9997 [23], where r^2 score is a common metric to evaluate time-series machine learning algorithms and to be described later in materials and methods section of the paper. With multiple linear regression, r^2 score of 0.9992 is reached [24]. A variety of regressors are used in Ref. [25], and the best result is achieved using a decision tree, which yields mean absolute percentage error (mape) value of 0.8981. A deep autoencoder and ConvLSTM framework is used to estimate outbreak dates in Ref. [26]. Polynomial regression results in 93.0% accuracy in Ref. [27]. With exponential smoothing an r^2 score of 0.98 is obtained [28]. An uncommon method, namely, wavelet coupled random link functional model yields an r^2 score of 0.99 [29].

In some studies, mixed type of data are used. In Ref. [30], intensive symptom weight learning mechanism results in 97.17% accuracy. In Ref. [31], random forest achieves 82.0% accuracy. A classical method, namely logistic regression analysis creates a 96.2% AUC in Ref. [32]. Reinforcement learning is applied in Ref. [33], and an ensemble model in Ref. [34] results in 91.0% accuracy. Online analytical processing is applied with 80.0% confidence [35]. With local network analysis, accuracy value range is 92%–98% [36]. Harris hawks optimized fuzzy k-nearest neighbor method yields 94% accuracy [37]. In Ref. [38], explainable convolution LSTM network is used and the results are given in rmse values. Random forest classifier results in 82.0% AUC [39]. In Ref. [40] random forest, extreme gradient boosting and gradient boosting methods function together to provide the best prediction of the future health condition of the Covid-19 patients. Best performing method is selected upon a voting scheme and an accuracy of 97.24% is reached.

Research on Covid-19 data of Turkey is also available. In Ref. [41], Spearman's rank correlation coefficient calculation is made but the results are not comparable to other studies, since there is no other work with the same coefficient to compare. In Ref. [42], polynomial regression, least squares polynomial fit, cubic spline fit are used to analyze the characteristics of Covid-19 data [42]. lays the foundation of this study and 97.8% accuracy is obtained. In Ref. [43], box-cox exponential smoothing is applied. Arima and LSTM are applied to achieve 79.0–95.0% accuracy [44]. With arima, 99.8% accuracy is obtained [45]. In Ref. [46], poisson regression is applied to yield a pseudo r^2 score of 0.907, which cannot be compared to other results. Decision tree and polynomial regression result in 98.7% and 99.4% accuracies, respectively in Ref. [47]. Eight different machine learning classifiers are used in Ref. [48] with the AUC values greater than 92.0%. LSTM network yields an r^2 score of 0.895 in Ref. [49] and mape of 0.70 in Ref. [50].

Analyzing the previous works on daily cases and deaths in Turkey, it can be concluded that a thorough study on the nature and the characteristics of the data for the whole period of data availability is missing. This kind of analysis is needed because it will help to draw conclusions, serve as a benchmark, and be prepared for possible future pandemics. This research aims to fill the gap. The main contributions of this study are.

a) It provides a through statistical analysis of the daily cases and deaths in Turkey in the whole period when data was available.
b) It provides predictions of future values of the daily cases and deaths in Turkey with high accuracies using a random forest regressor.
c) It provides a comparison of the proposed model with the other state-of-the art ensemble and nonensemble methods, obtaining the best performance by the random forest model.
d) It provides a benchmark for future pandemic studies both in Turkey and in other countries since Turkey is one of the most affected countries in pandemics.
e) The novelty lies in a comprehensive analysis and prediction of the daily cases and deaths for the entire period of data. Such a thorough study does not exist for Turkey.
f) It provides a decision support system where future deaths and cases can be predicted very accurately so that in-time planning of allocation of hospital beds, intensive care units and ambulances can be achieved successfuly by the authorities, which helps to prevent the devastating spread as was the case in Covid-19 pandemic.

The organization of the paper is as follows: In Section 2, materials are discussed and statistical analysis of the data is illustrated. In Section 3, methods are described. In Section 4 performance criteria used in this study are introduced. In Section 5 experimental results are given and discussions are provided. In Section 6 conclusions are drawn.

2. Materials

The starting point of this study is the analysis of the Covid-19 data characteristics in Turkey by statistical methods. Then several models are built for predicting future values using machine learning methods. Python programming language is used both for the statistical analysis and in the machine learning work. Properties of the data and the methods used are described in the sections that follow.

2.1. Data description

The length of the time series used to estimate daily cases and daily deaths is 813 samples. The length is determined by the daily data provided by the Ministry of Health of Turkey and downloaded from the website of the World Health Organization [51]. The data starts on March 11, 2020 and ends on June 1, 2022. After this date, the data was provided weekly, and later on, every two weeks. The daily data are visualized in Figs. 1 and 4.

In Table 1, statistical properties of daily cases are given. The mean of daily cases is approximately 18,539 and the maximum number of infected people on a given day is 111,157.

The coefficient of variation is calculated by dividing the standard deviation by the mean. It gives information about the dispersion of the dataset [52]. Daily cases has a higher coefficient of variation (Table 2) in comparison to daily deaths (Table 4), namely 1.14 and 0.73, respectively. Therefore, the data related to daily cases has a wider spread than the data related to daily deaths.

OLS (Ordinary Least Squares) Regression results of both datasets (Figs. 2 and 5, respectively) indicate that, it is not possible to predict these datasets using OLS Regression model since F-statistic values are far from satisfying the null hypothesis, i.e., they are not close to 0. Besides, r^2 values are far from being satisfactory. They should be close to 1 for a good fit.

For normal distribution of errors, Prob(Omnibus) must be close to 1, but in both daily cases and deaths the values are zero. Therefore, it can be concluded that errors are not normally distributed, and hence, the null hypothesis is rejected.

Skewness and kurtosis values for both datasets prove that they are not normal datasets [53].

For both datasets, Akaike's Information Criteria (AIC) and Bayesian Information Criteria (BIC) are very large numbers, meaning that the model is not good enough to represent the characteristics under consideration.

As far as the decomposition of both datasets are concerned, seasonal values are oscillatory in character, showing that the difference between the actual data and the trend goes quickly from negative to positive and vice versa. The increase of the noisy structure in the original data is reflected in the residual characteristic of daily deaths starting from April 2021 (Fig. 6), whereas, even though a similar characteristic is seen around the same date, it is much more enhanced starting from January 2022 in daily cases (Fig. 3). Starting from April 2022, residual values settle down to smaller values for both datasets.

In Table 3, statistical properties of daily deaths are given. The mean of daily deaths is approximately 121 and the maximum number of deaths on a given day is 394.

From the above analyses of the datasets, it can be inferred that the datasets are both nonstationary. Therefore for predicting future values, complex algorithms need to be employed.

3. Methods

Due to the complexity of the time-series data related to Covid-19, simple regression models do not yield satisfactory results. The

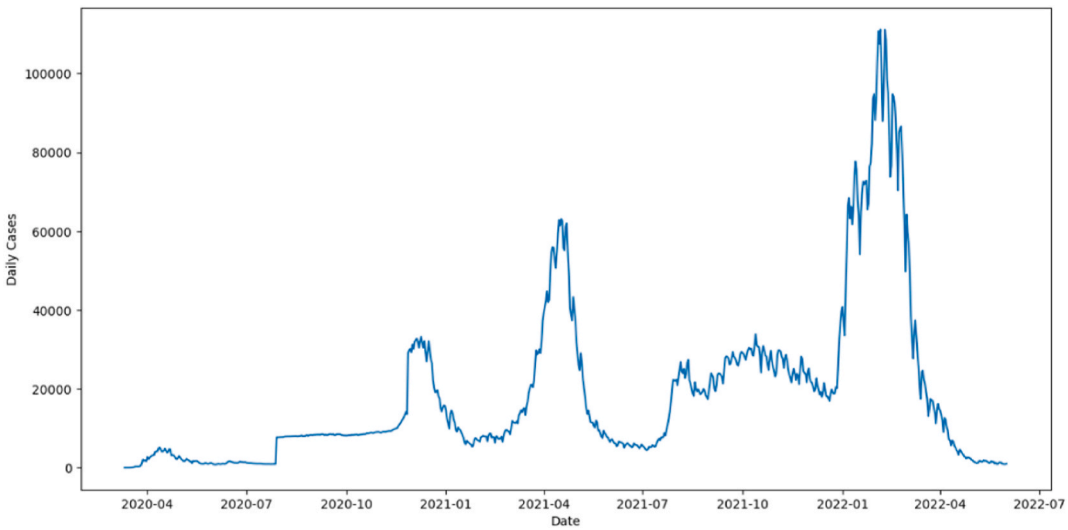


Fig. 1. Number of daily cases in Turkey.

Table 1
Statistical properties of daily cases.

	Daily Cases	Order
count	813	813
mean	18,539.66	406
standard deviation	21,165.14	234.84
minimum value	0	0
25%	5299	203
50%	9354	406
75%	24,856	609
maximum value	111,157	812

Table 2
Mean, standard deviation and coefficient of variation of daily cases.

	Mean	Standard Deviation	Coefficient of Variation
Daily Cases	18,626.74	21,166.97	1.14

OLS Regression Results

Dep. Variable:	y	R-squared (uncentered):	0.558
Model:	OLS	Adj. R-squared (uncentered):	0.558
Method:	Least Squares	F-statistic:	1026.
Date:	Thu, 18 Jan 2024	Prob (F-statistic):	3.74e-146
Time:	22:38:15	Log-Likelihood:	-9150.4
No. Observations:	813	AIC:	1.830e+04
Df Residuals:	812	BIC:	1.831e+04
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
x1	44.8066	1.399	32.024	0.000	42.060	47.553

Omnibus:	231.736	Durbin-Watson:	0.021
Prob(Omnibus):	0.000	Jarque-Bera (JB):	687.176
Skew:	1.403	Prob(JB):	6.05e-150
Kurtosis:	6.523	Cond. No.	1.00

Notes:
[1] R^2 is computed without centering (uncentered) since the model does not contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Fig. 2. Ordinary least squares regression model for daily cases.

model developed using random forest in this work, estimates the daily data with high accuracy and r^2 score values. The model is compared to other ensemble models (Bagging, Adaptive Boosting, Gradient Boosting, Extreme Gradient Boosting), to a decision tree regressor model, to a well-established algorithm for estimating time-series data, namely, long short-term network (LSTM), and also to an ARIMA model which constitutes a widely used one for forecasting nonstationary time-series.

The work done is summarized in Fig. 7. The datasets are retrieved from the World Health Organization (WHO) database [1]. The necessary statistical analysis and data preprocessing are performed and the datasets become ready for regressors. Eight different methods are applied and the results are compared on various criteria, which are described in Section 4 of this paper.

3.1. Ensemble learning

Random forest is a special type of ensemble learning algorithm. Ensemble learning is a machine learning paradigm which enhances the accuracy and robustness of predictions by combining the outputs of multiple models. It is applicable to and yields good results in classification and regression. In regression problems, which is the subject of this study, time-series data is used to forecast future values using previous values. Through ensemble learning, different models can be trained with different features, different algorithms, or different hyperparameters, and their predictions can be combined using techniques such as simple or weighted averaging.

Different approaches to ensemble learning are possible. One possibility is the bagging (bootstrap aggregating) method, which trains

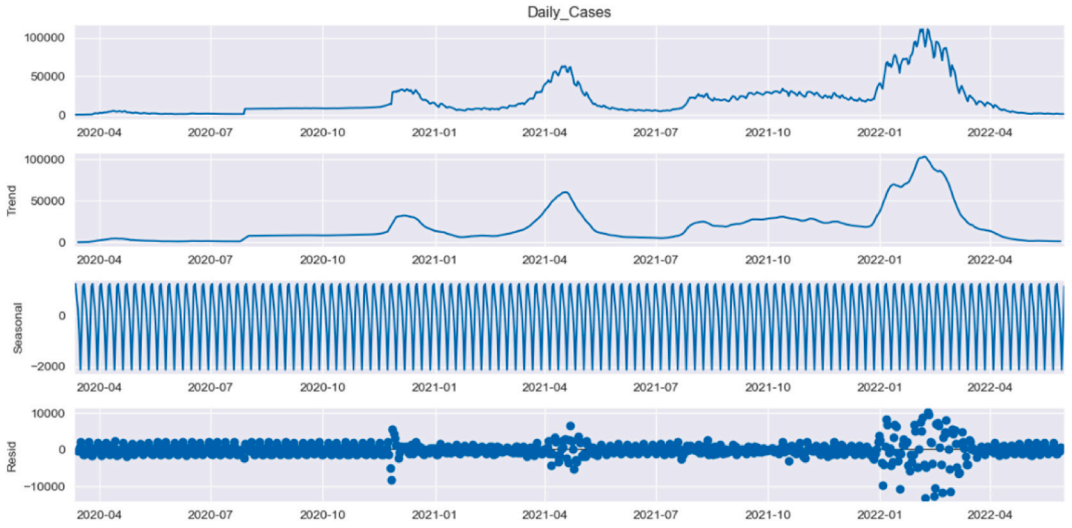


Fig. 3. Trend, seasonal and residual characteristics of daily cases.

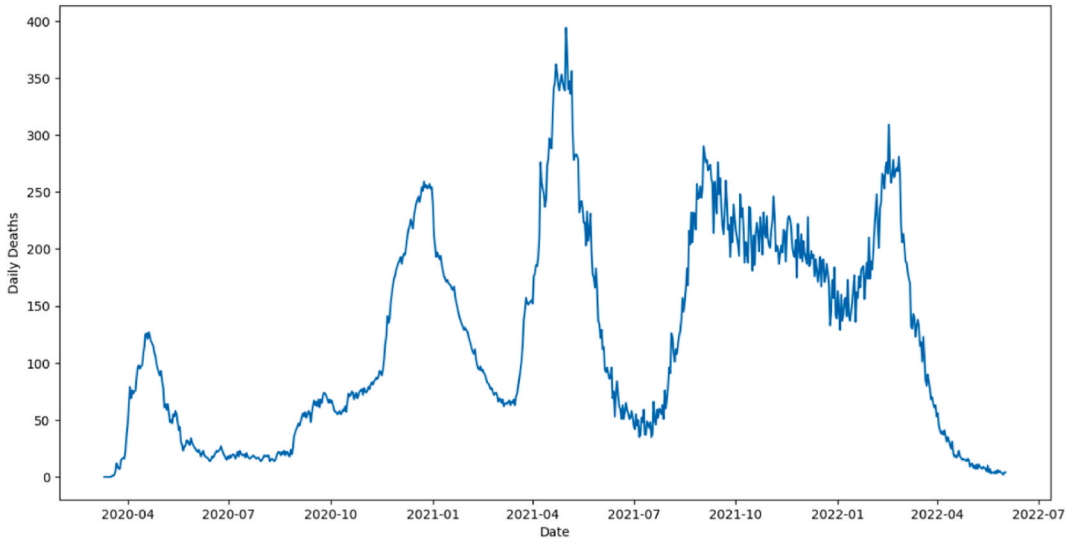


Fig. 4. Number of daily deaths in Turkey.

multiple models on different bootstrap samples of the training data and aggregates their predictions. The bagging algorithm employed in this work is shown in Fig. 8.

Another possibility is the boosting method, which trains weak models successively, where each model is trained to correct the errors created by the previous model. The boosting algorithm employed in this work is shown in Fig. 9.

A third possibility is the stacking technique. In this approach, multiple models are trained and their outputs are used as inputs to a higher-level model, which learns to combine the predictions of the sub-models, with the aim of increasing the overall success.

The ensemble regressors used in this work are bagging, adaptive boosting, gradient boosting, extreme gradient boosting and random forest regressors. A detailed derivation of these methods can be found in Ref. [54]. In addition, decision tree regressors, long short-term memory networks and autoregressive integrated moving average model are used to compare the results with the developed model.

3.2. *Bagging regressor*

Bagging (bootstrap aggregating) combines multiple lower-level models for enhancing the accuracy and the stability of the forecasts. Bootstrap methods have been applied for many decades in different contexts [55]. To build a bagging regressor, the first step is to randomly select a subset of the training data for each base model using bootstrap sampling, that is to say, selection with replacement.

OLS Regression Results

Dep. Variable:	y	R-squared (uncentered):	0.637
Model:	OLS	Adj. R-squared (uncentered):	0.636
Method:	Least Squares	F-statistic:	1423.
Date:	Thu, 18 Jan 2024	Prob (F-statistic):	1.01e-180
Time:	22:46:14	Log-Likelihood:	-4822.7
No. Observations:	813	AIC:	9647.
Df Residuals:	812	BIC:	9652.
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
x1	0.2575	0.007	37.725	0.000	0.244	0.271

Omnibus:	20.515	Durbin-Watson:	0.020
Prob(Omnibus):	0.000	Jarque-Bera (JB):	27.959
Skew:	-0.259	Prob(JB):	8.49e-07
Kurtosis:	3.746	Cond. No.	1.00

Notes:
[1] R^2 is computed without centering (uncentered) since the model does not contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Fig. 5. Ordinary least squares regression model for daily deaths.

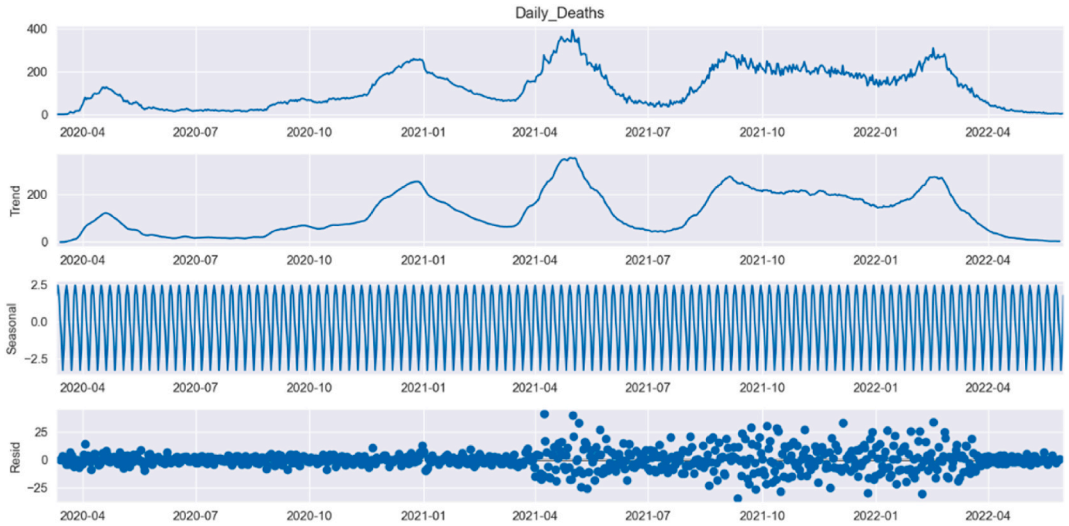


Fig. 6. Trend, seasonal and residual characteristics of daily deaths.

Table 3
Statistical properties of Daily deaths.

	Daily Deaths	Order
count	813	813
mean	121.73	406
standard deviation	89.93	234.84
minimum value	0	0
25%	46	203
50%	96	406
75%	195	609
maximum value	394	812

Table 4
Mean, standard deviation and coefficient of variation figures of daily deaths.

	Mean	Standard Deviation	Coefficient of Variation
Daily Deaths	122.31	89.71	0.73

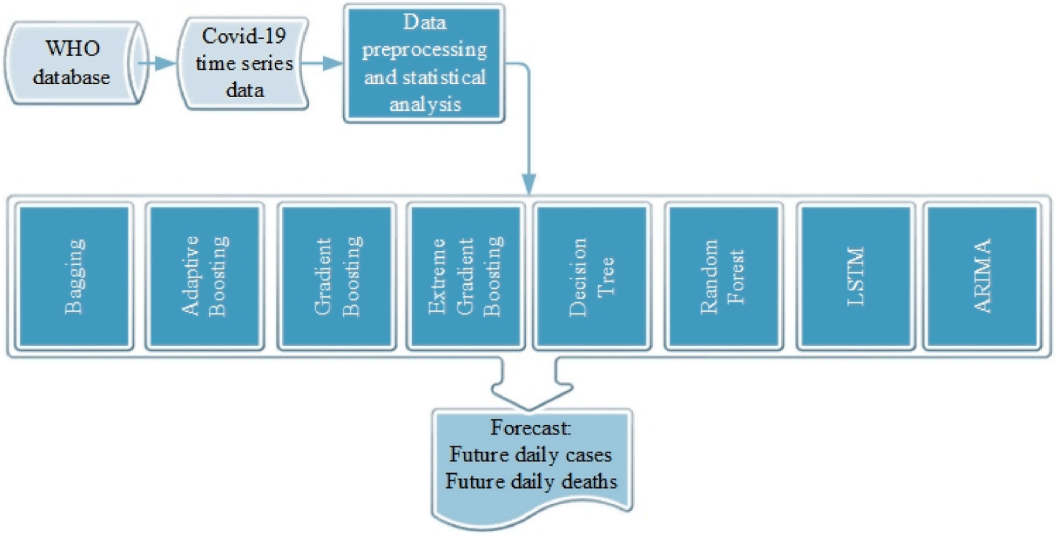


Fig. 7. The flowchart of the work done in the paper.

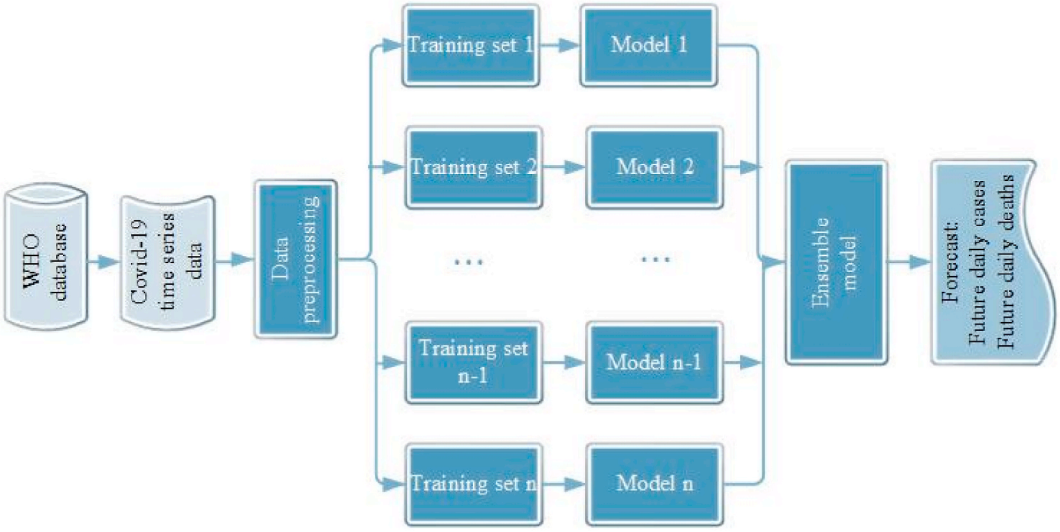


Fig. 8. Bagging regressor for Covid time-series prediction.

But this means that some training instances may be selected multiple times, while others may not be selected at all.

Next, a base model, which can be any fast and simple algorithm, is trained on each bootstrap sample. The base models should have low variance, since bagging is used to reduce variance [54].

Once all the base models are trained, the target variable is predicted by simple or weighted averaging of the predictions of all the models. If a weighted mean is used, the weights are selected in proportion to the performance of each model.

The bagging regressor has several advantages over individual base models for regression. It is less prone to overfitting and is more robust to noisy or irrelevant features.

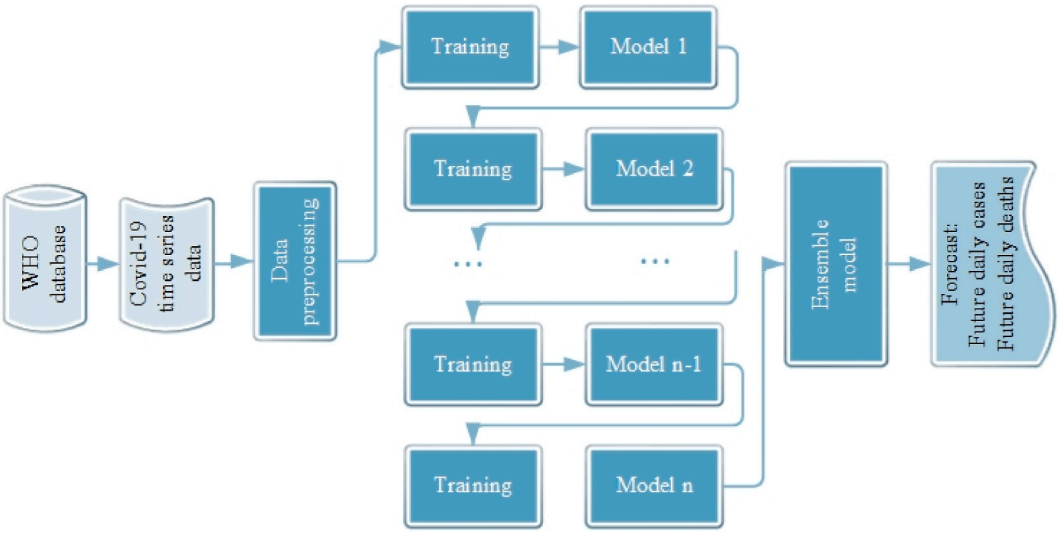


Fig. 9. Boosting regressor for Covid time-series prediction.

3.3. Adaptive boosting regressor

Adaptive Boosting (AdaBoost) can be used for regression tasks. It works by iteratively training base models on different versions of the data. At each iteration, the algorithm adjusts the weights of the training examples based on error rate from the lower-level model [54]. The weights of the examples are adjusted at each iteration to enhance the accuracy of the total model. AdaBoost is less prone to overfitting than other algorithms, however, it can be sensitive to noisy data and outliers. The base models in AdaBoost are typically decision trees.

The design parameters of the AdaBoost model, for example the number of base models, the learning rate, and the depth of the decision trees, are adjusted using a variety of techniques, such as grid search or random search.

3.4. Gradient boosting regressor

The Gradient Boosting algorithm works by iteratively adding weak base models to the ensemble. Each base model is trained using the residuals, namely the errors of the lower-level models. The aim is to enhance the prediction accuracy of the overall model [54].

The algorithm uses a predefined loss function, which is common for gradient-based approaches, and this loss function is error based. The Gradient Boosting algorithm then calculates the negative gradient or residual of the predefined loss function. This is used as the target variable for the next lower-level model in the ensemble.

The base models in Gradient Boosting are typically decision trees. To prevent overfitting, regularization techniques such as tree pruning and early stopping are commonly used.

It is also less prone to overfitting than other algorithms, but it can be computationally expensive.

3.5. Extreme gradient boosting regressor

Extreme Gradient Boosting (XGBoost) is an extension of the gradient boosting algorithm that uses a different objective function and regularization techniques to improve performance.

The objective function used by XGBoost for regression is the mean squared error. In addition, XGBoost uses regularization techniques to prevent overfitting and improve generalization performance. This includes L_1 and L_2 regularizations, which penalize the magnitude of the weights in the model, and a technique called 'tree pruning', which removes unnecessary branches in the decision trees.

XGBoost has inner mechanisms for handling outliers.

3.6. Decision tree regressor

A decision tree recursively partitions the data set based on the values of the attributes, using a greedy approach that selects the best attribute at each step, namely, the attribute that produces the greatest reduction in variance, and this results in a split that best separates the data into homogeneous groups [56].

One of the main advantages of decision trees for regression is their interpretability. The tree structure allows for a visualization of the decision-making process, making it easy to understand how the model is making its predictions. Decision trees can also handle

missing values by imputing the most common value or by creating a separate branch for missing values.

On the other hand, decision trees suffer from overfitting, if the tree is too complex or when the data set is noisy. Overfitting occurs when the tree memorizes the training data. To avoid overfitting, various techniques such as pruning, regularization, and ensemble methods can be used.

3.7. Random forest regressor

Random forest uses bagging with random feature selection [56]. Therefore, Fig. 8 illustrates random forest regressor, as well. It uses trees in its predictions. The randomness in the structure is useful for decreasing the variance of the model and the prediction is made by averaging the forecasts of the trees employed.

The random selection of training instances and input features introduces diversity among the decision trees, allowing them to capture different aspects of the relationship between the input features and the target variable. This decreases the correlation between the trees, making the random forest more robust to noise or outliers.

3.8. Long short-term memory network

Long short-term memory (LSTM) networks are capable of handling long-term dependencies in sequential data, which can be important for accurately predicting the target variable. They also have a memory cell that can selectively remember or forget information, which helps to prevent the network from being overwhelmed by irrelevant or noisy input features.

The memory cell in an LSTM network is used to store information about the input sequence that is relevant for the prediction of the target variable. The memory cell can remember through input gate or forget information through forget gate using gating mechanisms that are learned during training. An example of a memory cell is shown in Fig. 10 [57]. In the figure, the abbreviation 'AF' stands for activation function. Different types of activation functions, such as relu, tanh, sigmoid can be used in the same cell. Multiplications are done by multiplying the corresponding elements of vectors.

The LSTM network also has an output gate that determines which information from the memory cell to output should be transferred as the prediction of the target variable. This output gate can also be used to regulate the amount of information that is propagated through the network, which helps to prevent the vanishing or exploding gradient problem that can occur in deep neural networks [58].

The relationships in Fig. 10 are calculated as in Eq.s 1–4:

$$f_t = AF\left(h_{t-1}, x_t, b_f\right) \tag{1}$$

$$c_t = c_{t-1} \times f_t + AF(h_{t-1}, x_t, b_i) \times AF(h_{t-1}, x_t, b_o) \tag{2}$$

$$y_t = h_t = AF(h_{t-1}, x_t, b_o) \times AF(c_t) \tag{3}$$

$$AF(h_{t-1}, x_t, b_k) = AF(Hh_{t-1} + Xx_t + b_k) \tag{4}$$

where H, X and b_k represent vectors of parameters. H and X represent weight vectors, on the other hand b_k represents bias vector and it is equal to b_f, b_i or b_o. Also Eq. (1) expresses the input to forget gate, namely f_t and it is the output of a nonlinear activation function at the same time. The variable c_t is the input to next LSTM unit and transfers the state of the cell state under consideration. Likewise, c_{t-1} transfers the information related to the state of the previous LSTM unit to the cell in Fig. 10. The expression of c_t is given by Eq. (2) where AF stands for an activation function with the purpose of adding nonlinearity to the model to be able to express more general cases in addition to linear problems and is described by Eq. (4). Finally, Eq. (3) describes the output of the given cell. In all the equations the hidden state coming from the previous cell (h_{t-1}) acts as an input, as well. A multi-layer LSTM network is shown in Fig. 11.

To train an LSTM network for regression, an appropriate loss function, defined using the error between the predicted and the actual

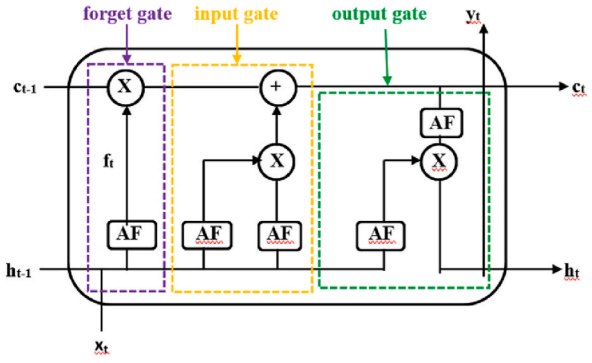


Fig. 10. A typical LSTM cell.

variable, is minimized. The network parameters, including the weights and biases of the gates and memory cell, are updated using an appropriate optimization algorithm, such as backpropagation.

LSTM networks are preferred for many current applications.

3.9. Auto Regressive integrated moving average (ARIMA) model

ARIMA is a common technique used for time-series forecasting. It is an extended version of classical Auto Regressive Moving Average (ARMA) technique. The future of stationary time-series can be predicted using ARMA models, whereas this is not possible when the time-series is nonstationary. The problem is solved by introducing differencing into ARMA structure, hence to make the time-series stationary. The name is modified as ARIMA, where I stands for "Integrated." The ARIMA model is specified using three arguments (p, d, q) where p is the number of AR terms, q is the number of MA terms and d is the number of differences [59]. The equation that describes the prediction of the time-series is shown in Eq. (5):

$$\left(1-\sum_{i=1}^{p}a_iB^i\right)(1-B)^d y_t = c+\left(1+\sum_{j=1}^{q}b_jB^j\right)e_t \tag{5}$$

where B is the backshift operator defined by Eq. (6):

$$B^k y_t = y_{t-k} \tag{6}$$

c is a constant, e_t is the error between the actual and the predicted values, y_t is the predicted value of the time series at the instant t.

4. Performance criteria

The performance metrics selected for measuring the validity of the models and also for comparison are accuracy, coefficient of variation [52] and root-mean-square error [60] scores.

Per cent accuracy is defined by Eq. (7):

$$\%\ accuracy = \left(1-\sum_i \frac{|y_i-\widehat{y}_i|}{|y_i|}\right)*100 \tag{7}$$

The coefficient of variation, r^2 score, is defined by Eq. (8):

$$r^2\ score = 1-\frac{\sum_i (y_i-\widehat{y}_i)^2}{\sum_i (y_i-\overline{y})^2} \tag{8}$$

Rmse, namely, root-mean-square score, is defined by Eq. (9):

$$rmse\ score = \sqrt{\frac{\sum_{i=1}^{N}(y_i-\widehat{y}_i)^2}{N}} \tag{9}$$

where y_i is the actual value, $\widehat{y}_i$ is the predicted value, $\overline{y}$ is the mean value of the actual values and N is the total number of samples.

The best possible value for the per cent accuracy is 100. But in machine learning applications, this is not preferred as training accuracy since it would mean memorizing training values and a possible failure in the prediction of the test values.

The best possible value for the coefficient of variation is 1 but the values close to 1 are also acceptable. Negative values can also be encountered, but in these cases, the model used for prediction is not satisfactory and should not be used [61]. Another metric, namely explained variance score, produces similar results, therefore it is not used to avoid the duplication of the results [62].

Since rmse score is an indication of the error, it is better to have small values but it is not possible to define an admissable range, simply because it is very much dependent upon the characteristics of the problem under consideration. Depending on the data used, rmse score can be computed either for training or test.

5. Results and discussion

Daily cases are predicted using random forest, bagging, XG boosting, decision tree regressors, long short-term network and ARIMA model. The other methods, namely adaptive and gradient boosting techniques, did not provide satisfactory results and they are not included here. As can be seen from Table 5, the best results are obtained by random forest regressor. On the other hand, bagging regressor produces very similar results, as well. To investigate the results further, relative errors of both regressors, defined by Eq. (10):

$$relative\ error(i) = \frac{y_i-\widehat{y}_i}{y_i} \tag{10}$$

Fig. 11. A simple LSTM network.

Table 5
Simulation results of predicting daily cases.

Method	Train accuracy	Test accuracy	r^2 score train	r^2 score test
XG boosting regressor	85.59	90.26	0.9999	0.9826
Decision tree regressor	100.00	90.48	1.0000	0.9806
Bagging regressor	95.91	92.17	0.9971	0.9893
Random forest regressor	95.25	92.30	0.9980	0.9886

where y_i is the actual value, $\hat{y}_i$ is the predicted value for the ith sample, are drawn in Fig. 12, and the respective rmse values are calculated. The rmse value of random forest regressor is less than that of bagging regressor, namely 2122.92 and 2347.76, respectively. Therefore, random forest regressor should be preferred. In Table 5, the train accuracy of the decision tree regressor is 100% whereas test accuracy is 90.48%. This indicates overfitting, which is not uncommon in the case of decision trees.

The predictions of both bagging and random forest regressors are shown in Fig. 13, where actual values are also shown. The values of the regressors for each test sample are close, therefore they cannot be observed on the graph individually. They drift from the actual values at certain samples and these cases are shown clearly on the graph.

To compare the results with the results of another well-established algorithm, namely LSTM network, predictions of the last 30% of the sample values are made. Test data accuracy is found to be 89.39%, in comparison to random forest test data accuracy, which is 92.30%. Moreover r^2 score of test data in the case of LSTM network is 0.9833 (Fig. 14) in comparison to r^2 score of test data of random forest, which is 0.9886 (Table 5). Besides LSTM model takes a much longer time to train and test.

In addition, relative error of LSTM network is worse, which is reflected in its rmse figure of 4079.20 (Fig. 15) in comparison to the rmse figure of random forest regressor of 2122.92 (Fig. 12).

The details of the model used as the LSTM network in simulations are summarized in Table 6.

To be able to compare the results obtained by a well-known time-series forecasting technique, an ARIMA model is built. Its results for predicting daily cases is similar to LSTM. Test data accuracy is slightly better than LSTM (90.25% versus 89.39%) but r^2 score (0.9790 versus 0.9833) and rmse value (4217.6376 versus 4079.1954) are worse. The prediction of test data for daily cases using ARIMA model is shown in Fig. 16.

Daily deaths are estimated using random forest, bagging, AdaBoost, gradient boosting, XG boosting, decision tree regressors, long short-term network and an ARIMA model. As can be seen from Table 7, the best results are obtained by random forest regressors. Bagging regressor produces the second to best results. To compare the results of the two regressors further, relative errors of both of them are drawn in Fig. 17 and the respective rmse values are calculated. The rmse figure of random forest regressor is less than that of bagging regressor, 11.0805 and 12.4090, respectively (Fig. 17). Therefore, random forest regressor should be preferred. In Table 7, the train accuracy of the decision tree regressor is 100%, as in the case of daily cases, whereas test accuracy is 90.02%. This indicates overfitting, therefore it is not a good idea to prefer the decision tree regressor.

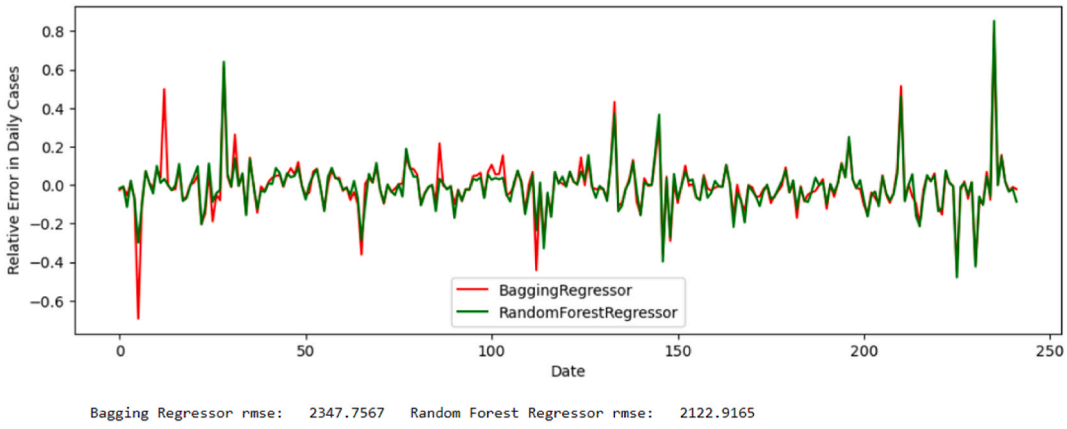


Fig. 12. Relative error of bagging and random forest regressors for daily cases.

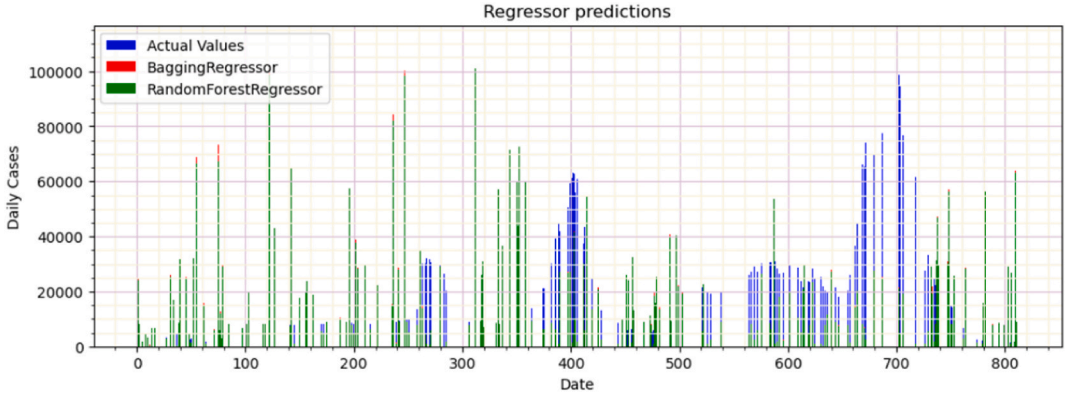


Fig. 13. Predicting daily cases using bagging and random forest regressors.

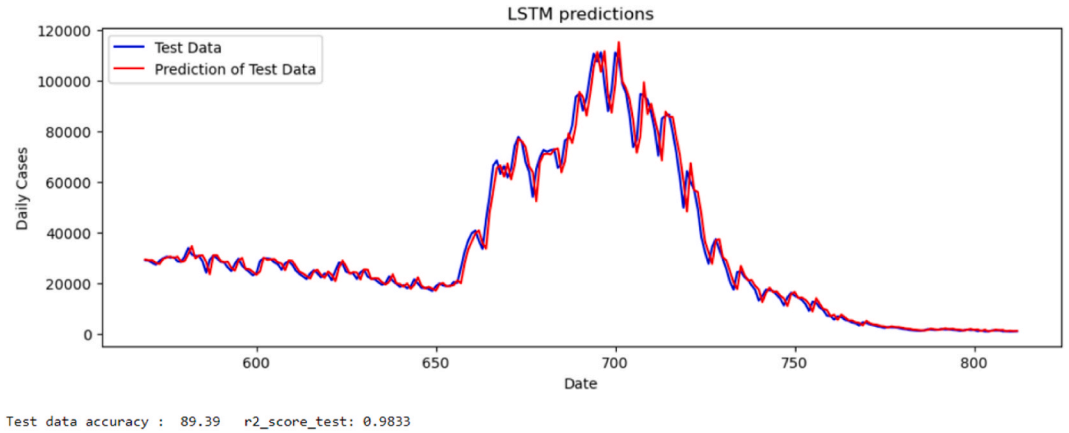


Fig. 14. LSTM prediction of daily cases.

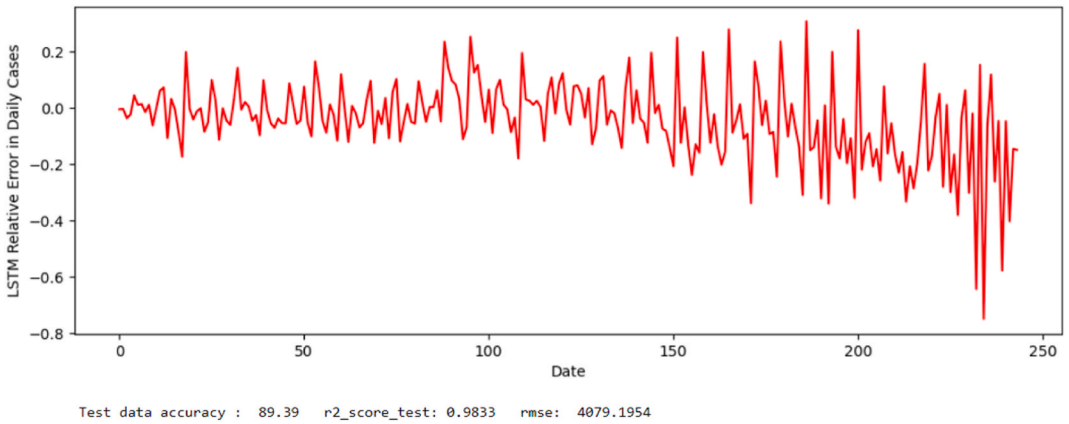


Fig. 15. Relative error of LSTM network for daily cases.

The predictions of both bagging and random forest regressors are shown in Fig. 18, where actual values are also shown. The values of regressors for each test sample are close to the actual values, therefore they cannot be observed on the graph individually.

To compare the results with the results of LSTM network, predictions of the last 30% of sample values are made with LSTM network, similar to daily cases. Test data accuracy is found to be 82.57% (Fig. 19), in comparison to random forest test data accuracy, which is 91.39% (Table 7). On the other hand r^2 score of test data in the case of LSTM network is 0.9867 (Fig. 19), therefore slightly better than r^2 score of test data of random forest, which is 0.9834 (Table 7). Since the test accuracy of LSTM model is much lower,

Table 6
LSTM model parameters.

Model: "sequential_29"		
Layer (type)	Output Shape	Param #
lstm_29 (LSTM)	(1, 200)	161600
Dense_29 (Dense)	(1, 1)	201
Total params: 161,801		
Trainable params: 161,801		
Non-trainable params: 0		

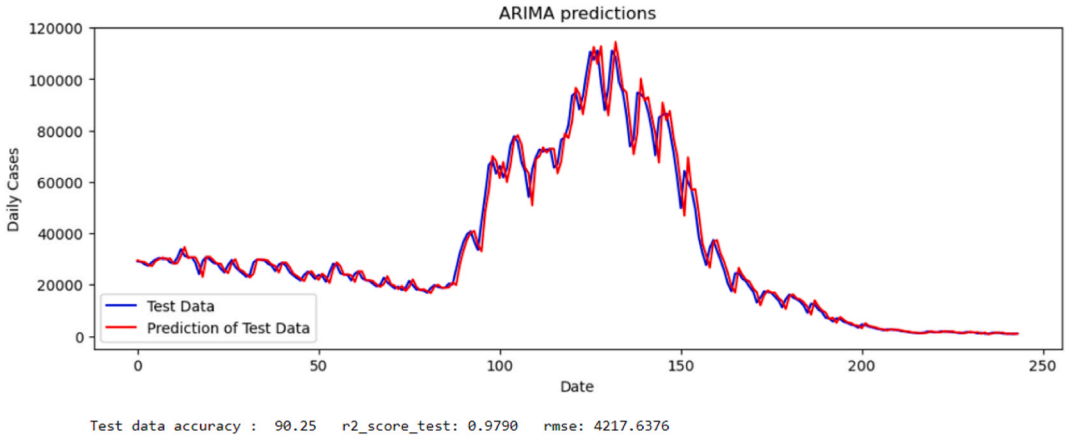


Fig. 16. Predicting daily cases using ARIMA model.

Table 7
Simulation results of predicting daily deaths.

Method	Train accuracy	Test accuracy	r^2 score train	r^2 score test
Adaboost regressor	38.61	21.46	0.7723	0.7538
Gradient boosting regressor	91.22	86.57	0.9913	0.9787
XG boosting regressor	97.95	89.31	0.9996	0.9738
Decision tree regressor	100.00	90.02	1.0000	0.9772
Bagging regressor	96.71	90.99	0.9971	0.9797
Random forest regressor	96.98	91.39	0.9980	0.9834

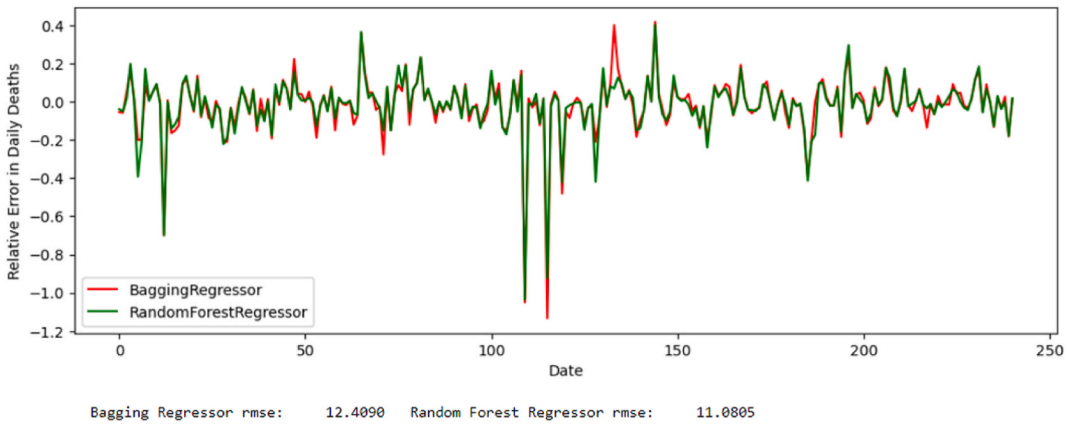


Fig. 17. Relative error of bagging and random forest regressors for daily deaths.

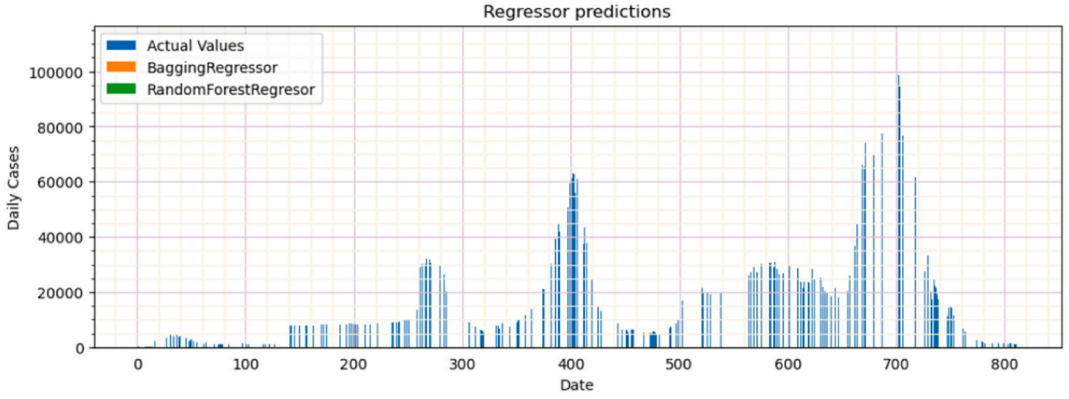


Fig. 18. Predicting daily deaths using bagging and random forest regressors.

random forest model is better.

In addition, the relative error of LSTM network is worse, which is reflected in its rmse figure of 16.1173 (Fig. 20) in comparison to rmse figure of random forest regressor, which is 11.0805 (Fig. 17).

The results are also compared to an ARIMA model. Test data accuracy is better than LSTM (87.89% versus 82.57%), but r^2 score is worse (0.9660 versus 0.9867) and rmse value (15.8248 versus 16.1173) is lower. The prediction of test data for daily deaths using ARIMA model is shown in Fig. 21.

Parameters of the developed models are selected using a grid search technique and are as follows.

- AdaBoost uses 1000 estimators and the learning rate is 1.0
- Bagging regressor uses 10 decision trees.
- Gradient boosting uses 100 estimators with maximum depth of 3 and the learning rate of 0.1
- Random forest uses 100 estimators.
- Decision tree uses a maximum depth of 20.
- ARIMA model uses (1, 1, 1) architecture.
- LSTM parameters are shown above in Table 6.

The simulations in this study are made using 12th Gen Intel Core i7-12650H Processor at 2.3 GHz with 16 GB of RAM and NVIDIA GeForce RTX 3060. The programming language used is Python and the environment is Jupyter Notebook.

5.1. A note on bagging regressor

The bagging regressor is used for prediction of both daily cases and deaths. In simulation, each time the ensemble methods are applied, it is observed that the bagging regressor produces slighlty varying results. This comes from the fact that bagging regressor selects data randomly and with replacement, therefore each run results in another selection with replacement. To analyze the effect of this kind of randomness on test accuracy, r^2 score and rmse values for test of daily death data, the program is run 100 times and the

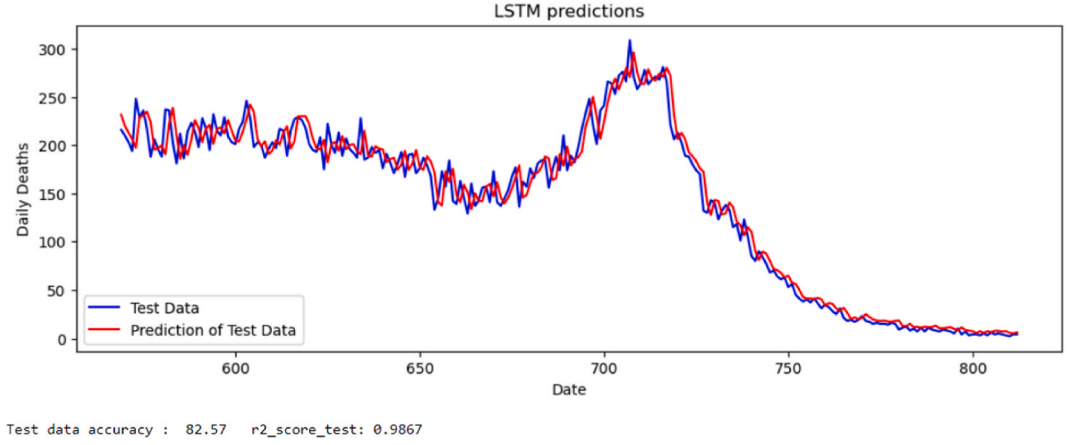


Fig. 19. LSTM prediction of daily deaths.

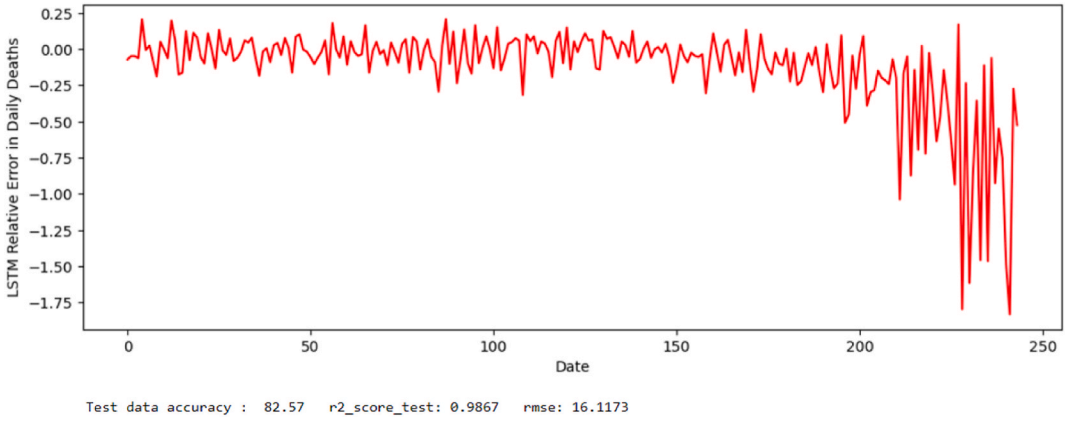


Fig. 20. Relative error of LSTM network for daily deaths.

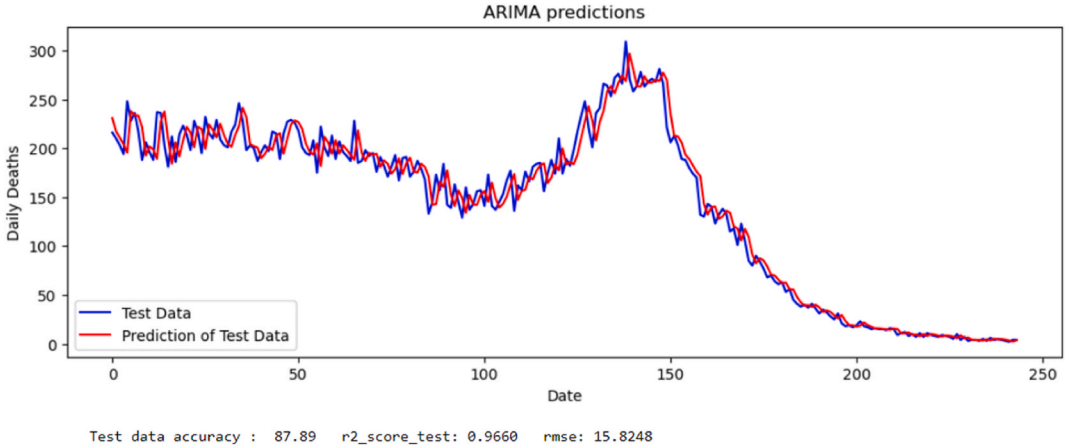


Fig. 21. ARIMA prediction of daily deaths.

results obtained are plotted in Fig. 22, Figs. 23 and 24.

In Fig. 22, it is shown that test accuracy fluctuates between 90.08% and 91.77%, with an average of 91.10%.

In Fig. 23, it is shown that r^2 score for test data fluctuates between 0.9793 and 0.9841, with an average of 0.9819.

In Fig. 24, it is seen that rmse value for test data fluctuates between 10.8301 and 12.6166, with an average of 11.6712.

The comparison of this study to the other works found in the literature is not easy simply because the experiments do not use the

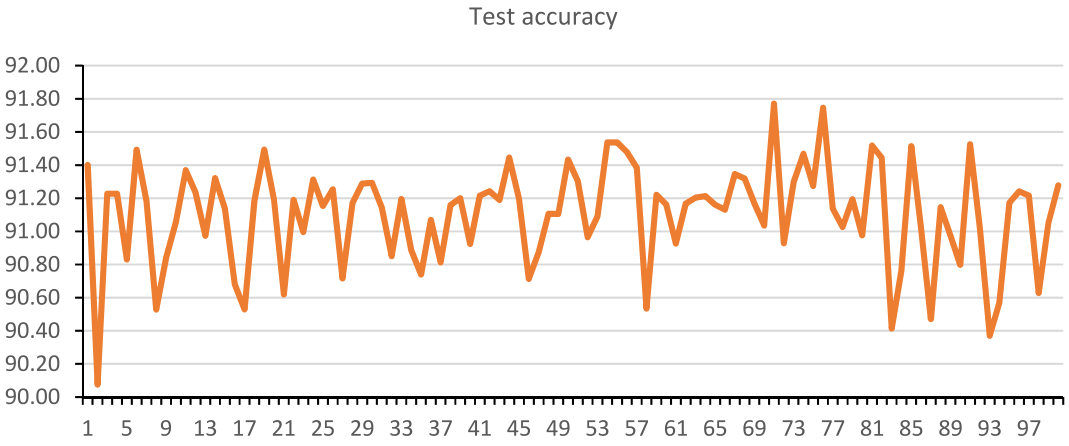


Fig. 22. Fluctuations in test accuracy values of bagging regressor.

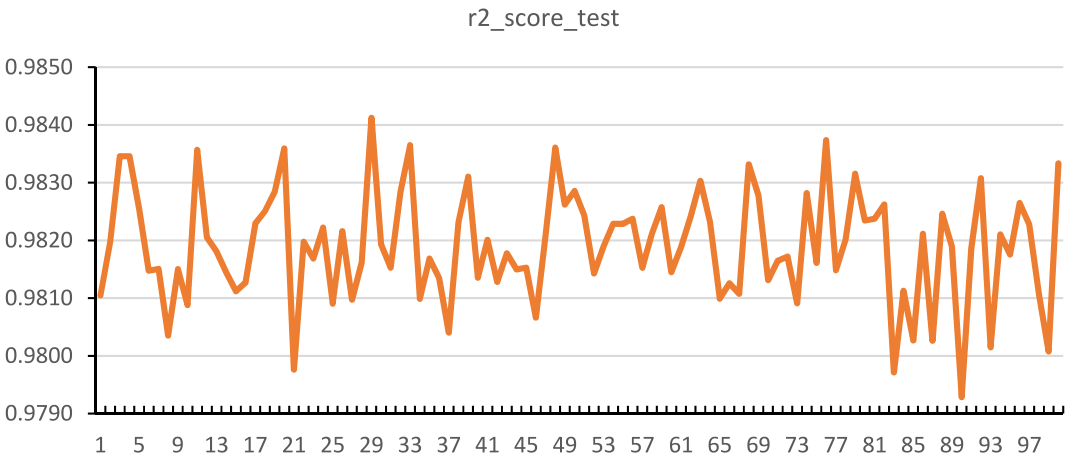


Fig. 23. Fluctuations in r^2 score values of bagging regressor.

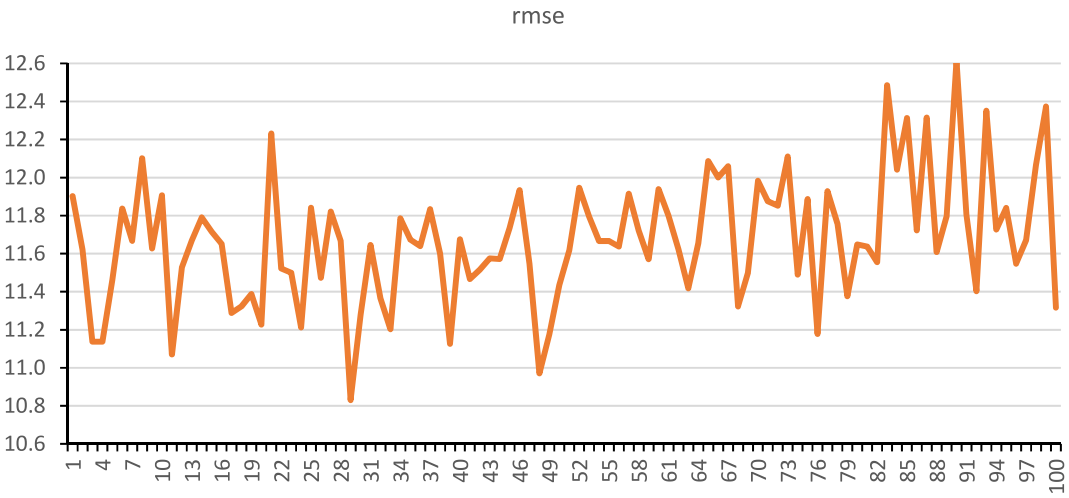


Fig. 24. Fluctuations in test rmse values of bagging regressor.

same datasets. Even though some use the same dataset(s) in some of the works, the period of time is different. Therefore, only a rough comparison can be made. In Table 8, two of the similar works are compared with the results of this work. In Ref. [44], the data of 315 days are used, predictions are made using LSTM network for 15 days. The accuracy reached for daily cases (85.15%) is worse than the accuracy of the work in this paper (89.19% with LSTM network and 92.30% with random forest regressor). On the other hand accuracy reached for daily deaths is comparable to the result of this work.

In [49], data of 378 days is used and 30 days are estimated using LSTM network. Only r^2 score for daily cases is provided. The LSTM network in this paper is 9.83% better and the random forest regressor is 10.50% better when the r^2 scores are compared.

6. Conclusion

This study demonstrates the superiority of the use of random forest based prediction of Turkey's Covid-19 daily data for cases and deaths. The statistical properties of the datasets are also studied and it is concluded that the datasets exhibit nonstationary character

Table 8
Comparison of predicting daily cases and deaths.

Ref	Data (days)	Test (days)	Daily cases	Daily deaths	r^2 score (Daily cases)
[44]	315	15	85.15%	**91.40%**	–
[49]	378	30	–	–	0.895
LSTM network in this paper	813	244	89.39%	82.57%	0.983
Random Forest regressor in this paper	813	244	**92.30%**	**91.39%**	**0.989**

and linear methods are not applicable.

The findings show that random forest and bagging regressors outperform other models developed in this study and it is concluded that random forest is better due to fluctuations in the performance of the bagging regressor in different runs of the program. When random forest regressors are employed, test data related to daily cases are predicted by 92.30% accuracy and with an r^2 score of 0.9893, in addition, daily deaths are predicted with 91.39% accuracy and with an r^2 score of 0.9834. With the same regressor, training samples are predicted with much higher accuracies and r^2 scores, namely daily cases are predicted with an accuracy of 95.25% and with an r^2 score of 0.9980, and daily deaths are predicted with an accuracy of 96.98% and with an r^2 score of 0.9980. Therefore, it can be concluded that random forest regressor is an efficient and powerful method to predict future values of Covid-19 daily cases and deaths for the data of Turkey.

In this work, the limitations are such that the effects of curfew, quarantine periods, compulsory use of face masks in public are not taken into account. On the other hand, they have direct impacts on the characteristics under consideration. Another limitation is in the use of data. It would be better to make predictions using other factors such as the number of patients in the intensive care units and number of recovering patients. These affect new cases and death numbers. This kind of approach could not be applied due to the fact that the data provided by the source is not complete for many periods of time and only daily deaths and daily cases are consistently reported. Also, this study can be extended to compare the results of data from various countries.

As the future work, stacking ensemble algorithms can be applied to the same datasets. Besides, use of face masks, effects of curfew and quarantine periods can be taken into account.

In future pandemics, the analysis of this work can provide a reference for both the spread of the disease and the occurrence of deaths in countries with high populations like Turkey.

Funding

No funds, grants, or other support was received.

Data availability statement

The data required to reproduce the above findings can be downloaded from WHO Coronavirus (COVID-19) Dashboard as referenced in the paper.

CRediT authorship contribution statement

Figen Özen: Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Formal analysis, Data curation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] WHO Covid 19. https://covid19.who.int/, 2023. (Accessed 20 December 2023).
[2] P. Mirmohammadi, A. Rasooli, M. Ashtiyani, M.M. Amin, M.R. Deevband, Automatic recognition of acute lymphoblastic leukemia using multi-SVM classifier, Curr Sci 115 (8) (2018) 1512–1518, https://doi.org/10.18520/cs/v115/i8/1512-1518.
[3] N. Bibi, M. Sikandar, I.U. Din, A. Almogren, S. Ali, IOMT-based automated detection and classification of leukemia using deep learning, J Healthc Eng 2020 (2020), https://doi.org/10.1155/2020/6648574.
[4] D. Shen, G. Wu, H.-I. Suk, Deep learning in medical image analysis, Annu Rev Biomed Eng 19 (1) (Jun. 2017) 221–248, https://doi.org/10.1146/annurev-bioeng-071516-044442.
[5] V. Balakrishnan, Y. Kehrabi, G. Ramanathan, S.A. Paul, C.K. Tiong, Machine learning approaches in diagnosing tuberculosis through biomarkers - a systematic review, Prog Biophys Mol Biol (May 2023), https://doi.org/10.1016/j.pbiomolbio.2023.03.001.
[6] Y. Liu, S. Mazumdar, P.A. Bath, An unsupervised learning approach to diagnosing Alzheimer's disease using brain magnetic resonance imaging scans, Int J Med Inform 173 (May 2023), https://doi.org/10.1016/j.ijmedinf.2023.105027.
[7] Y. Miyachi, O. Ishii, K. Torigoe, Design, implementation, and evaluation of the computer-aided clinical decision support system based on learning-to-rank: collaboration between physicians and machine learning in the differential diagnosis process, BMC Med Inform Decis Mak 23 (1) (Dec. 2023), https://doi.org/10.1186/s12911-023-02123-5.
[8] K. Noguchi, I. Saito, T. Namiki, Y. Yoshimura, T. Nakaguchi, Reliability of non-contact tongue diagnosis for Sjögren's syndrome using machine learning method, Sci Rep 13 (1) (Dec. 2023), https://doi.org/10.1038/s41598-023-27764-4.
[9] T. Haferlach, W. Walter, Challenging gold standard hematology diagnostics through the introduction of whole genome sequencing and artificial intelligence, in: International Journal of Laboratory Hematology, John Wiley and Sons Inc, 2023, https://doi.org/10.1111/ijlh.14033. Apr. 01.
[10] H. Kang, et al., Diagnosis of Coronavirus disease 2019 (COVID-19) with structured latent multi-view representation learning, IEEE Trans Med Imaging 39 (8) (Aug. 2020) 2606–2614, https://doi.org/10.1109/TMI.2020.2992546.
[11] M. Ahishali, et al., Advance warning methodologies for COVID-19 using chest X-ray images, IEEE Access 9 (2021) 41052–41065, https://doi.org/10.1109/ACCESS.2021.3064927.
[12] A.G. Dastider, F. Sadik, S.A. Fattah, An integrated autoencoder-based hybrid CNN-LSTM model for COVID-19 severity prediction from lung ultrasound, Comput Biol Med 132 (May 2021), https://doi.org/10.1016/j.compbiomed.2021.104296.
[13] D. Di, et al., Hypergraph learning for identification of COVID-19 with CT imaging, Med Image Anal 68 (Feb. 2021), https://doi.org/10.1016/j.media.2020.101910.

[14] E.S.M. El-Kenawy, A. Ibrahim, S. Mirjalili, M.M. Eid, S.E. Hussein, Novel feature selection and voting classifier algorithms for COVID-19 classification in CT images, IEEE Access 8 (2020), https://doi.org/10.1109/ACCESS.2020.3028012.

[15] M.R. Hassan, W.N. Ismail, A. Chowdhury, S. Hossain, S. Huda, M.M. Hassan, A framework of genetic algorithm-based CNN on multi-access edge computing for automated detection of COVID-19, Journal of Supercomputing 78 (7) (May 2022) 10250–10274, https://doi.org/10.1007/s11227-021-04222-4.

[16] Z. Han, et al., Accurate screening of COVID-19 using attention-based deep 3D multiple instance learning, IEEE Trans Med Imaging 39 (8) (Aug. 2020) 2584–2594, https://doi.org/10.1109/TMI.2020.2996256.

[17] D.M. Ibrahim, N.M. Elshennawy, A.M. Sarhan, Deep-chest: multi-classification deep learning model for diagnosing COVID-19, pneumonia, and lung cancer chest diseases, Comput Biol Med 132 (May 2021), https://doi.org/10.1016/j.compbiomed.2021.104348.

[18] M. Shorfuzzaman, M.S. Hossain, MetaCOVID: a Siamese neural network framework with contrastive loss for n-shot diagnosis of COVID-19 patients, Pattern Recognit 113 (May 2021), https://doi.org/10.1016/j.patcog.2020.107700.

[19] Y.H. Bhosale, K.S. Patnaik, "PulDi-COVID: chronic obstructive pulmonary (lung) diseases with COVID-19 classification using ensemble deep convolutional neural network from chest X-ray images to minimize severity and mortality rates,", Biomed Signal Process Control 81 (Mar. 2023) https://doi.org/10.1016/j.bspc.2022.104445.

[20] M. Li, et al., Explainable COVID-19 infections identification and delineation using calibrated pseudo labels, IEEE Trans Emerg Top Comput Intell 7 (1) (Feb. 2023) 26–35, https://doi.org/10.1109/TETCI.2022.3189054.

[21] S. Benítez-Peña, et al., On sparse ensemble methods: an application to short-term predictions of the evolution of COVID-19, Eur J Oper Res 295 (2) (Dec. 2021) 648–663, https://doi.org/10.1016/j.ejor.2021.04.016.

[22] P. Wang, X. Zheng, G. Ai, D. Liu, B. Zhu, Time series prediction for the epidemic trends of COVID-19 using the improved LSTM deep learning method: case studies in Russia, Peru and Iran, Chaos Solitons Fractals 140 (Nov. 2020), https://doi.org/10.1016/j.chaos.2020.110214.

[23] F. Shahid, A. Zameer, M. Muneeb, Predictions for COVID-19 with deep learning models of LSTM, GRU and Bi-LSTM, Chaos Solitons Fractals 140 (Nov. 2020), https://doi.org/10.1016/j.chaos.2020.110212.

[24] R. Kumari, et al., Analysis and predictions of spread, recovery, and death caused by COVID-19 in India, Big Data Mining and Analytics 4 (2) (Jun. 2021) 65–75, https://doi.org/10.26599/BDMA.2020.9020013.

[25] Z. Malki, E.S. Atlam, A.E. Hassanien, G. Dagnew, M.A. Elhosseini, I. Gad, Association between weather data and COVID-19 pandemic predicting mortality rate: machine learning approaches, Chaos Solitons Fractals 138 (Sep) (2020), https://doi.org/10.1016/j.chaos.2020.110137.

[26] Y. Karadayi, M.N. Aydin, A.S. Öğrenci, Unsupervised anomaly detection in multivariate spatio-temporal data using deep learning: early detection of covid-19 outbreak in Italy, IEEE Access 8 (2020) 164155–164177, https://doi.org/10.1109/ACCESS.2020.3022366.

[27] E. Gambhir, R. Jain, A. Gupta, U. Tomer, Regression analysis of COVID-19 using machine learning algorithms, in: 2020 International Conference on Smart Electronics and Communication, IEEE, Trichy, India, 2020, pp. 65–71, https://doi.org/10.1109/ICOSEC49089.2020.9215356.

[28] F. Rustam, et al., COVID-19 future forecasting using supervised machine learning models, IEEE Access 8 (2020) 101489–101499, https://doi.org/10.1109/ACCESS.2020.2997311.

[29] B.B. Hazarika, D. Gupta, Modelling and forecasting of COVID-19 spread using wavelet-coupled random vector functional link networks, Applied Soft Computing Journal 96 (Nov. 2020), https://doi.org/10.1016/j.asoc.2020.106626.

[30] L. Fang, X. Liang, "ISW-LM: an intensive symptom weight learning mechanism for early COVID-19 diagnosis,", Comput Biol Med 146 (Jul. 2022) https://doi.org/10.1016/j.compbiomed.2022.105615.

[31] J. Leitner, A. Behnke, P.-H. Chiang, M. Ritter, M. Millen, S. Dey, Classification of patient recovery from COVID-19 symptoms using consumer wearables and machine learning, IEEE J Biomed Health Inform 27 (3) (Jan. 2023) 1271–1282, https://doi.org/10.1109/jbhi.2023.3239366.

[32] S. Bao, et al., A diagnostic model for serious COVID-19 infection among older adults in Shanghai during the Omicron wave, Front Med (Lausanne) 9 (Dec. 2022), https://doi.org/10.3389/fmed.2022.1018516.

[33] R. Padmanabhan, N. Meskin, T. Khattab, M. Shraim, M. Al-Hitmi, Reinforcement learning-based decision support system for COVID-19, Biomed Signal Process Control 68 (Jul. 2021), https://doi.org/10.1016/j.bspc.2021.102676.

[34] D. Goodman-Meza, et al., A machine learning algorithm to increase COVID-19 inpatient diagnostic capacity, PLoS One 15 (9 September) (Sep. 2020), https://doi.org/10.1371/journal.pone.0239474.

[35] C.K. Leung, Y. Chen, C.S.H. Hoi, S. Shang, A. Cuzzocrea, Machine learning and OLAP on big COVID-19 data, in: Proceedings - 2020 IEEE International Conference on Big Data, Institute of Electrical and Electronics Engineers Inc., Dec. 2020, pp. 5118–5127, https://doi.org/10.1109/BigData50022.2020.9378407. *Big Data 2020*.

[36] C.A. Reis Pinheiro, M. Galati, N. Summerville, M. Lambrecht, Using network analysis and machine learning to identify virus spread trends in COVID-19, Big Data Research 25 (Jul. 2021), https://doi.org/10.1016/j.bdr.2021.100242.

[37] H. Ye, et al., Diagnosing Coronavirus disease 2019 (COVID-19): efficient harris hawks-inspired fuzzy K-nearest neighbor prediction methods, IEEE Access 9 (2021) 17787–17802, https://doi.org/10.1109/ACCESS.2021.3052835.

[38] N. Yudistira, S.B. Sumitro, A. Nahas, N.F. Riama, Learning where to look for COVID-19 growth: multivariate analysis of COVID-19 cases over time using explainable convolution–LSTM, Appl Soft Comput 109 (Sep. 2021), https://doi.org/10.1016/j.asoc.2021.107469.

[39] G. Raman, et al., Machine learning prediction for COVID-19 disease severity at hospital admission, BMC Med Inform Decis Mak 23 (1) (Dec. 2023) 46, https://doi.org/10.1186/s12911-023-02132-4.

[40] A. Gumaei, et al., A decision-level fusion method for COVID-19 patient health prediction, Big Data Research 27 (Feb. 2022), https://doi.org/10.1016/j.bdr.2021.100287.

[41] M. Şahin, "Impact of weather on COVID-19 pandemic in Turkey,", Science of the Total Environment 728 (Aug. 2020) https://doi.org/10.1016/j.scitotenv.2020.138810.

[42] F. Özen, Estimation of daily cases, deaths, serious patients and recovering Pa-tients of covid-19 in Turkey with machine learning methods, Journal of Advanced Research in Natural and Applied Sciences (Jul. 2022), https://doi.org/10.28979/jarnas.1055917.

[43] B. Ergul, A. Altinyavuz, E. GundoganAsik, B. Kalay, Statistical evaluation of the COVID-19 outbreak data as of april around the world and in Turkey, Anadolu Kliniği Tıp Bilimleri Dergisi (Apr. 2020), https://doi.org/10.21673/anadoluklin.719629.

[44] A.A. Karcioğlu, S. Tanişman, H. Bulut, Türkiye'de COVID-19 Bulaşısının ARIMA Modeli ve LSTM Ağı Kullanılarak Zaman Serisi Tahmini, European Journal of Science and Technology, Jan. 2022, https://doi.org/10.31590/ejosat.1039394.

[45] S. Akay, H. Akay, Time Series Model for Forecasting the Number of Covid-19 Cases in Turkey, Türkiye Halk Sağlığı Dergisi, Apr. 2021, https://doi.org/10.20518/tjph.809201.

[46] B. Tasdelen, D. Dericiyildirim, Predicting COVID-19 cases in Turkey with Poisson regression and the effect of preventions on incidence rate ratio estimation, Turkiye Klinikleri Journal of Biostatistics 12 (3) (2020) 293–302, https://doi.org/10.5336/biostatic.2020-77595.

[47] O. Çağlar, F. Özen, A comparison of Covid-19 cases and deaths in Turkey and in other countries, Network Modeling Analysis in Health Informatics and Bioinformatics 11 (1) (Dec. 2022), https://doi.org/10.1007/s13721-022-00389-9.

[48] S. Ustebay, A. Sarmis, G.K. Kaya, M. Sujan, A comparison of machine learning algorithms in predicting COVID-19 prognostics, Intern Emerg Med 18 (1) (Jan. 2023) 229–239, https://doi.org/10.1007/s11739-022-03101-x.

[49] G. Guclu, A. Al-Dulaimi, Estimating and analyzing the spread of covid-19 in Turkey using long short-term memory, in: ISMSIT 2021 - 5th International Symposium on Multidisciplinary Studies and Innovative Technologies, Proceedings, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 17–26, https://doi.org/10.1109/ISMSIT52890.2021.9604594.

[50] S.S. Helli, Ç. Demirci, O. Çoban, A. Hamamci, Short-term forecasting COVID-19 cases in Turkey using long short-term memory network, in: TIPTEKNO 2020 - Tip Teknolojileri Kongresi - 2020 Medical Technologies Congress, TIPTEKNO 2020, Institute of Electrical and Electronics Engineers Inc., Nov. 2020, https://doi.org/10.1109/TIPTEKNO50054.2020.9299235.

[51] "WHO Coronavirus (COVID-19) Dashboard," Accessed 1 December 2022.

[52] R.E. Walpole, R.H. Myers, S.L. Myers, K. Ye, Probability and Statistics for Engineers and Scientists, eighth ed., Pearson Education International, NJ, 2007.
[53] T.T. Soong, Probability and Statistics for Engineers, John Wiley & Sons, Ltd., West Sussex, 2004.
[54] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning, second ed., Springer, 2017.
[55] J.S. Maritz, T. Lwin, Empirical Bayes Methods, second ed., Chapman & Hall, London, 1995.
[56] S. Theodoridis, K. Koutroumbas, Pattern Recognition, fourth ed., Academic Press, 2010.
[57] Y. Wei, J. Jang-Jaccard, W. Xu, F. Sabrina, S. Camtepe, M. Boulic, LSTM-Autoencoder-Based anomaly detection for indoor air quality time-series data, IEEE Sens J 23 (4) (Feb. 2023) 3787–3800, https://doi.org/10.1109/JSEN.2022.3230361.
[58] C.C. Aggarwal, Neural Networks and Deep Learning, Springer, 2018.
[59] G.E.P. Box, G.M. Jenkins, G.C. Reinsel, G.M. Ljung, Time Series Analysis Forecasting and Control, fifth ed., Wiley, 2015.
[60] R. Yates, D.J. Goodman, Probability and Stochastic Processes, second ed., John Wiley & Sons, Inc., 2005.
[61] L. Igual, S. Segui, Introduction to Data Science: A Python Approach to Concepts, Techniques and Applications, Springer, 2017, https://doi.org/10.1007/978-3-319-50017-1.
[62] R.E. Walpole, R.H. Myers, S.L. Myers, K. Ye, Probability and Statistics for Engineers and Scientists, eighth ed., Pearson Education International, NJ, 2007.