A Comparison Between Adaptive Neuro-fuzzy Inference System and Autoregressive Integrated Moving Average in Predicting COVID-19 Confirmed Cases in Bangladesh

Sirajum Monir Parvez, Syed Shahir Ahmed Rakin, Md. Asadut Zaman, Istiaq Ahmed, Redwanul Alam Alif, Ania-Nin-Ania, and Rashedur M. Rahman

Abstract Since December 2019, the novel coronavirus (COVID-19) has become one of the most contagious diseases to have hit the world for several decades. From December 2019 till May 2020, this respiratory syndrome-like disease has quickly spread to all countries around the world and has taken more than 400 thousand lives. The WHO declared a global pandemic situation due to the virus from March 2020. The source of this virus is not known, especially since there are no well-placed standards for its diagnosis and treatment. Several factors are involved in the spread of the disease. There have been several studies to predict or forecast the number of new cases in upcoming dates. In our study, we tested the widely used ANFIS—Adaptive Neuro-Fuzzy Inference System and the ARIMA—Autoregressive Integrated Moving Average methods to predict the total number of COVID-19 cases in the

S. M. Parvez (✉) · S. S. A. Rakin · Md. Asadut Zaman · I. Ahmed · R. A. Alif · Ania-Nin-Ania · R. M. Rahman
Department of Electrical and Computer Engineering, North South University, Plot-15, Block-B, Bashundhara Residential Area, Dhaka, Bangladesh
e-mail: sirajum.monir@northsouth.edu
S. S. A. Rakin
e-mail: syed.rakin01@northsouth.edu
Md. Asadut Zaman
e-mail: asadut.zaman@northsouth.edu
I. Ahmed
e-mail: istiaq.ahmed18@northsouth.edu
R. A. Alif
e-mail: redwanul.alif@northsouth.edu
Ania-Nin-Ania
e-mail: aina.ania@northsouth.edu
R. M. Rahman
e-mail: rashedur.rahman@northsouth.edu

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021
S. Fong et al. (eds.), ICT Analysis and Applications, Lecture Notes in Networks and Systems 154, https://doi.org/10.1007/978-981-15-8354-4_73
upcoming days in Bangladesh. We tuned both the models with different configuration parameters, and made 3 distinct configurations for each. After that, we applied all the different configurations on the same dataset, and the results were compared against each other in terms of statistical performance measures such as Mean Absolute Percentage Error (MAPE), Root Mean Squared Relative Error (RMSRE), Root Mean Squared Relative Error (RMSRE).

Keywords COVID-19 · Adaptive neuro-fuzzy inference system (ANFIS) · Autoregressive integrated moving average (ARIMA) · Forecasting

1 Introduction

Coronavirus, a SAARS COV-2 like virus is responsible for causing a contagious flue like disease. The disease is named COVID-19 and has already created a global pandemic situation. First discovered in the Hubei city of the province Wuhan in China, this viral disease has now reached across every country around the world. As governments and health agencies struggle to prevent the spread of coronaviruses, they need every help. As no confirmed medication has been proposed by the WHO for COVID-19 yet, the most effective way to fight against the pandemic is to prevent it from spreading by predicting possible future cases and taking precautionary measures accordingly. In recent months, the trend and final dimension of the COVID-2019 pandemic have been forecasted by an increasing number of research works using various methods. Zhao et al. [13] in their work forecasted confirmed cases in China, and tried to estimate the number of unreported cases from Wuhan. The authors found the number to be at least 469 cases from 1 to 15 January 2020. Besides, this evidence had increased 21 folds after 17 January 2020. Al-qaness et al. [1] used the ANFIS prediction tool combined with Flower Pollination Algorithm (FPA) and the Salp Swarm Algorithm (SSA) to predict confirmed cases in China. Ardabili et al. [2] used stand-alone ANFIS to predict the cases. On the other hand, Perone et al. [10], Benvenuto et al. [4], Dehesh et al. [6], etc. adopted ARIMA for their prediction and observed good results. Kumar et al. [7] used ARIMA to predict the top 15 countries by April 2020 in terms of COVID infection. Pal et al. [9] used a Bayesian optimization framework to predict risk category of a country, where the proposed a shallow long short-term memory (LSTM) based neura network.

In countries like Bangladesh, the fight against the VIRUS is even more difficult concerning the poor infrastructure and the scarcity of tools. In our study, we vowed on finding a forecasting tool that will be best suited for predicting the number of COVID patients in Bangladesh. This data will help the concerned authority to take adequate precautionary measures to stall the spread of the infection. We have applied ANFIS and ARIMA to predict confirmed cases for upcoming days using Bangladesh’s COVID dataset for confirmed cases [12]. We have implemented the ANFIS model with three separate configurations for the prediction, and also made use of the ARIMA model with three different configurations to be applied on the
same dataset. Later we have compared output from all the different configurations based on a few performance parameters and proposed the best method out of them, which can be effectively utilized.

2 Methods

2.1 Fuzzy Inference System and ANFIS

Fuzzy Inference Systems take inputs and process them based on the prespecified rules to produce the outputs. Both the inputs and outputs are real-valued, whereas internal processing is based on fuzzy rules and fuzzy arithmetic. An ANFIS is a mixture of an adaptive neural network (ANN) and a fuzzy inference system. It is being used by many scholars due to its rapid learning capacity and the ability to capture the nonlinear structure of a process. Aside being used to forecast COVID cases recently, ANFIS has numerous applications in the past for other predictions as well. For instance, Mohaddes et al. [8] used ANFIS to forecast Iran’s agricultural product export, Zheng et al. [14] in their work used for short-term wind power prediction. Benmouiza et al. [3] used ANFIS with subtractive clustering and greed partitioning for an hour-ahead solar radiation forecasting. ANFIS architecture uses both artificial neural networks and fuzzy logic. The parameters of the fuzzy inference system are determined by the neural network. ANFIS can approximate to any degree of accuracy any real continuous function of a compact set of parameters. Since the system is based on a fuzzy inference system which reflects incredible information, it should always be translated into fluid IF-THEN rules.

- Rules—the if-then rules have to be determined somehow. This is mostly done by ‘knowledge acquisition’ from an expert. It is a time-consuming process that is fraught with problems.
- Membership Functions: Researchers and Data Scientists use membership functions for determining a full-fledged fuzzy set. For the case of Gaussian functions, there should be parameters set into the Gaussian function.

ANFIS stands for Adaptive Neuro-Fuzzy Inference System; as the name suggests, it is an adaptive network which has a structured network of nodes and directional links. ANFIS involves the use of a database that contains the required rules and membership functions for learning. The adaptive network of ANFIS consists of a set of rules for learning the data, such as backpropagation. As the name suggests, it is called adaptive for having parameters that could potentially bring alterations to the node of the output. Through the networks, a relationship between input nodes and output nodes can be established and further evaluated. These adaptive networks used in ANFIS have many ways to be implemented for different and unique purposes. In our case, we will be using the method initially brought up by Jang, which is the ANFIS method shown below. The fixed nodes in the diagram are symbolized by
circular nodes, while the learned parameters are given shape through square nodes (Fig. 1).

If \( x \) is \( A_i \) and \( y \) is \( B_i \) THEN \( f = p_i x + q_i y + r_i \)
If \( x \) is \( A_{i+1} \) and \( y \) is \( B_{i+1} \) THEN \( f = p_{i+1} x + q_{i+1} y + r_i \)

There is a forward and a backward pass for the training of the network. We look at the forward transfer on each layer in turn. The vector input is spread by network layer by layer through the forward pass. The error is returned in the backward transfer to the backpropagation through the network in the same way. It can be anything to membership. For instance, the following function represents a Gaussian membership function.

\[
\mu(x) = e^{-\left(\frac{x - \mu_i}{\alpha_i}\right)^2}
\]  

(1)

where the generalized Gaussian membership functions denoted by \( \mu, A_i \) and \( B_i \) define the membership values of \( \mu \). \( a_i, r_i \) denotes the premise parameters set.

The result/output of each layer is fed to the next layer as input with a weight assigned to it and in the final layer summation of all the nodes outputs are generated as final output, and is represented by

\[
\sum \tilde{\omega}_i f_i
\]  

(2)
2.2 Autoregressive Integrated Moving Average (ANFIS)

*Autoregressive Integrated Moving Average* (ARIMA) is a prediction method that projects future values of a series based entirely on their inertia. It primarily uses short-term projections, which requires at least 40 historical data points. This works very well if the data display a steady or stable trend with a minimum of outliers over time. Often referred to as Box-Jenkins method at the time when the data are relatively long and the association between past observations are stable, ARIMA is generally superior to exponential smoothing techniques. ARIMA is preferred for prediction using time-series data and it performs best when there is seasonality in the dataset. ARIMA has been used by many to predict time series data, for example, in 2018 Wadi et al. [11] used Amman Stock Exchange’s 8 years data from 2010 to 2018 for closed time series prediction. Rebane et al. [15] used ARIMA for cryptocurrency price prediction and compared the results of the model with a deep multi-layer Seq 2 Seq RNN model.

ARIMA models are always expressed with the help of a few parameters, and the model is expressed as ARIMA \((p, d, q)\). In this case, \(p\) means the order of self-regression, \(d\) means the degree of trend variance, and \(q\) means the average of movements. The autocorrelation function (ACF) graph and partial autocorrelation (PACF) graph is utilized to find the initial number of ARIMA models. The difference in normality and stationery is then evaluated for ARIMA models. After that, they are checked for accuracy by observing their MAPE, MAE, and RMSE values to determine the finest model to forecast. The model for forecasting the number of future confirmed COVID-19 cases is represented as,

\[
ARIMA(p, d, f): X_t = \alpha_1 X_{t-1} + \alpha_2 X_{t-2} + \beta_1 Z_{t-1} + \beta_2 Z_{t-2} + Z_t
\]  

(3)

where,

\[
Z_t = X_t - X_{t-1}
\]  

(4)

Here, \(X_t\) is the predicted number of confirmed COVID-19 cases at \(t\)th day, \(\alpha_1, \alpha_2, \beta_1,\) and \(\beta_2\) are parameters whereas \(Z_t\) is the residual term for \(t\)th day. In previous cases, the pattern of potential effects can be predicted and a time-series study is performed to that end.

3 Experiment

This section presents the description of the used dataset, the performance measures, the parameter setting for all methods, the experiment results, and discussions.
3.1 Data Description

The main dataset of this study is extracted from the Worldometers.com (https://www.worldometers.info/coronavirus/country/bangladesh/), a reference website providing counters and real-time statistics on diverse topics. It is owned by data company Dadax Limited [12]. It contains the daily confirmed cases, deaths, and recovery reports in Bangladesh from 16 March to date, as shown in Table 1. The COVID 19 crisis

| Date (D/M/Y) | New case | Total case | Date (D/M/Y) | New case | Total case | Date (D/M/Y) | New case | Total case |
|-------------|----------|------------|-------------|----------|------------|-------------|----------|------------|
| 16/3/20     | 3        | 3          | 14/4/20     | 209      | 1005       | 13/5/20     | 1162     | 17,795     |
| 17/3/20     | 0        | 3          | 15/4/20     | 219      | 1224       | 14/5/20     | 1041     | 18,836     |
| 18/3/20     | 4        | 7          | 16/4/20     | 341      | 1565       | 15/5/20     | 1202     | 20,038     |
| 19/3/20     | 4        | 11         | 17/4/20     | 266      | 1831       | 16/5/20     | 930      | 20,968     |
| 20/3/20     | 2        | 13         | 18/4/20     | 306      | 2137       | 17/5/20     | 1273     | 22,241     |
| 21/3/20     | 4        | 17         | 19/4/20     | 312      | 2449       | 18/5/20     | 1602     | 23,843     |
| 22/3/20     | 3        | 20         | 20/4/20     | 492      | 2941       | 19/5/20     | 1251     | 25,094     |
| 23/3/20     | 6        | 26         | 21/4/20     | 434      | 3375       | 20/5/20     | 1617     | 26,711     |
| 24/3/20     | 6        | 32         | 22/4/20     | 390      | 3765       | 21/5/20     | 1773     | 28,484     |
| 25/3/20     | 0        | 32         | 23/4/20     | 414      | 4179       | 22/5/20     | 1694     | 30,178     |
| 26/3/20     | 5        | 37         | 24/4/20     | 503      | 4682       | 23/5/20     | 1873     | 32,051     |
| 27/3/20     | 4        | 41         | 25/4/20     | 309      | 4991       | 24/5/20     | 1532     | 33,583     |
| 28/3/20     | 0        | 41         | 26/4/20     | 418      | 5409       | 25/5/20     | 1975     | 35,558     |
| 29/3/20     | 0        | 41         | 27/4/20     | 497      | 5906       | 26/5/20     | 1166     | 36,724     |
| 30/3/20     | 1        | 42         | 28/4/20     | 549      | 6455       | 27/5/20     | 1541     | 38,265     |
| 31/3/20     | 2        | 44         | 29/4/20     | 641      | 7096       | 28/5/20     | 2029     | 40,294     |
| 1/4/20      | 3        | 47         | 30/4/20     | 564      | 7660       | 29/5/20     | 2523     | 42,817     |
| 2/4/20      | 2        | 49         | 1/5/20      | 571      | 8231       | 30/5/20     | 1764     | 44,581     |
| 3/4/20      | 5        | 54         | 2/5/20      | 552      | 8783       | 31/5/20     | 2545     | 47,126     |
| 4/4/20      | 9        | 63         | 3/5/20      | 665      | 9448       | 1/6/20      | 2381     | 49,507     |
| 5/4/20      | 18       | 81         | 4/5/20      | 668      | 10,116     | 2/6/20      | 2911     | 52,418     |
| 6/4/20      | 35       | 116        | 5/5/20      | 786      | 10,902     | 3/6/20      | 2695     | 55,113     |
| 7/4/20      | 41       | 157        | 6/5/20      | 790      | 11,692     | 4/6/20      | 2423     | 57,536     |
| 8/4/20      | 54       | 211        | 7/5/20      | 706      | 12,398     | 5/6/20      | 2828     | 60,364     |
| 9/4/20      | 112      | 323        | 8/5/20      | 709      | 13,107     | 6/6/20      | 2635     | 62,999     |
| 10/4/20     | 94       | 417        | 9/5/20      | 636      | 13,743     | 7/6/20      | 2743     | 65,742     |
| 11/4/20     | 58       | 475        | 10/5/20     | 887      | 14,630     | 8/6/20      | 2735     | 68,477     |
| 12/4/20     | 139      | 614        | 11/5/20     | 1034     | 15,664     | 9/6/20      | 3171     | 71,648     |
| 13/4/20     | 182      | 796        | 12/5/20     | 969      | 16,633     |             |          |            |
definitely has increased the visibility of the website. It is one of Google’s highest-ranking coronavirus search results. Such time-series data are collected by the State, local governments, and health authorities from monitoring the ongoing epidemic.

Worldometers.com has been monitoring coronavirus cases in real-time since late January, as they have been identified following research. However, due to the broad lack of testing, the data in the outbreak picture is necessarily limited. We used data from 16 March 2020 to 30 May, 2020, to train the model. Data from 31 May 2020 to 9 June 2020 is used to test. The death cases were not considered as they do not have any relevance in the forecasting for confirmed cases. Therefore, data have been filtered to remove the death case column. Then we have done comparative analysis between ANFIS and ARIMA based on cumulative confirmed case forecasting. The key indicators for determining the quality of performance are the root mean squared error. Mean absolute error and mean percentage error.

3.2 Performance Measure and Parameter Settings

The quality of the proposed method is evaluated using a set of performance metrics as follows:

- **Root Mean Square Error (RMSE)**

\[
RMSE = \sqrt{\frac{1}{N_s} \sum_{i=1}^{N_s} (YP_i - Y)^2}
\]  

- **Mean Absolute Error (MAE)**

\[
MAE = \frac{1}{N_s} \sum_{i=1}^{N_s} |YP_i - Y_i|
\]

- **Mean Absolute Percentage Error (MAPE)**

\[
MAPE = \frac{1}{N_s} \sum_{i=1}^{N_s} \left| \frac{YP_i - Y_i}{YP_i} \right|
\]

We used ANFIS and ARIMA to predict the outcomes from the data set. We tuned the ANFIS model with different parameter settings to bring variation in the results, so did we do with the ARIMA model as well. We prepared 3 ANFIS models and
Table 2  Parameter settings for the three separate ANFIS configurations

| Settings                  | ANFIS-1                          | ANFIS-2                          | ANFIS-3                          |
|---------------------------|----------------------------------|----------------------------------|----------------------------------|
| Cluster type              | Grid partitioning                | Grid partitioning                | Subtractive clustering           |
| Input                     | Gaussian MF                      | Generalized bell-shaped MF       | 0.7                              |
| Output                    | Linear MF                        | Linear MF                        | 0.3                              |
| No. of membership functions | 3                                | 5                                | –                                |
| Epochs                    | 50                               | 100                              | 50                               |
| Step size                 | 1.10                             | 1.10                             | 1.30                             |

3 ARIMA models to with minor changes in the parameters. Then we computed the result to determine which model from the ANFIS and the ARIMA performed best individually. After that the best models from both cases were analyzed to observe the performance difference between ANFIS and ARIMA. Table 2 shows the parameter settings for the 3 distinct ANFIS models.

We first created an ANFIS model with 5 Gaussian membership functions and the input space was divided by Grid partitioning. The step size increase rate was set to its default 1.10 and the dataset was trained in 50 full cycles. For ease of understanding, we called this configuration to be ANFIS-1. The input membership function was then tuned and was changed to Generalized Bell-shaped MF for the second configuration, which was ANFIS-2. In this case the total number of membership functions were 5, and the Epoch was set to 100, while the step increase size remained the same. Figure 2a and b represents input membership functions for both these configurations.

For the third configuration, we went for a different clustering option by choosing subtractive clustering. The configuration is mentioned in Table 2.

On the other hand, for the ARIMA model we created 3 partitions by changing only the value for order of differencing, $d$. Its value for ARIMA-1 was set to 1 and for ARIMA-2, and for ARIMA-3, the value was set to 2 considering the data to be seasonal in nature this time. The other parameters remained the same. The parameter configurations are depicted in Table 3.

When we analyze the data to extract insights, we find that our ANFIS-3 configuration has the lowest percentage of error. It obtains Mean Absolute Percentage Error value of 2.46%. Among the ARIMA configurations, our ARIMA-3 model earns a value of 3.26%.
4 Results and Analysis

All 3 ANFIS models and 3 ARIMA models performed quite well in prediction. The results with respect to the original testing value is given in Table 4.

After analyzing the forecasted values, it is found that the ANFIS-3 configuration has the lowest percentage of error. It obtained Mean Absolute Percentage Error value of 2.46%. Among the three ARIMA configurations, the ARIMA-3 model earned a value of 3.26%. Performance parameters of the above results are mentioned in Table 5.
Table 3  Three separate ARIMA models’ parameter settings

| Parameter                                      | ARIMA-1 | ARIMA-2 | ARIMA-3 |
|-----------------------------------------------|---------|---------|---------|
| The order (no. of time lags) of the autoregressive (“AR”) model | 1       | 1       | 2       |
| The order of the moving average (“MA”) model  | 1       | 1       | 2       |
| The maximum value of $p$                      | 3       | 4       | 5       |
| The maximum value of $q$                      | 3       | 4       | 5       |
| Period of seasonal differencing               | 3       | 3       | 3       |
| The order of first-differencing               | 1       | 2       | 2       |
| The order of seasonal differencing            | 1       | 1       | 1       |

Table 4  Forecasted results from all the configurations

| Date (D/M/Y) | Actual cases | ANFIS -1 | ANFIS-2 | ANFIS-3 | ARIMA-1 | ARIMA-2 | ARIMA-3 |
|--------------|--------------|----------|---------|---------|---------|---------|---------|
| 31/5/20      | 47,126       | 45,684   | 46,008  | 46,438  | 46,526  | 46,574  | 46,655  |
| 1/6/20       | 49,507       | 47,429   | 47,791  | 48,385  | 48,466  | 48,595  | 48,807  |
| 2/6/20       | 52,418       | 49,173   | 49,552  | 50,331  | 50,401  | 50,642  | 51,000  |
| 3/6/20       | 55,113       | 50,917   | 51,276  | 52,266  | 52,330  | 52,717  | 53,267  |
| 4/6/20       | 57,536       | 52,659   | 52,948  | 54,184  | 54,254  | 54,819  | 55,615  |
| 5/6/20       | 60,364       | 54,399   | 54,555  | 56,074  | 56,173  | 56,948  | 58,007  |
| 6/6/20       | 62,999       | 56,135   | 56,082  | 57,929  | 58,086  | 59,105  | 60,476  |
| 7/6/20       | 65,742       | 57,869   | 57,516  | 59,740  | 59,994  | 61,289  | 63,029  |
| 8/6/20       | 68,477       | 59,600   | 58,849  | 61,498  | 61,898  | 63,499  | 65,628  |
| 9/6/20       | 71,648       | 61,328   | 60,073  | 63,193  | 63,796  | 65,738  | 68,308  |

Table 5  Comparison of all ANFIS and ARIMA configurations based on the performance parameters

| Method     | RMSE     | MAE     | MAPE (%) |
|------------|----------|---------|----------|
| ANFIS      |          |         |          |
| ANFIS-1    | 6231.39  | 5573.7  | 8.96     |
| ANFIS-2    | 6512.42  | 5628.0  | 8.95     |
| ANFIS-3    | 4751.37  | 1484.5  | 2.46     |
| ARIMA      |          |         |          |
| ARIMA-1    | 4513.39  | 3900.6  | 6.20     |
| ARIMA-2    | 3518.29  | 3100.4  | 4.96     |
| ARIMA-3    | 2199.41  | 2013.8  | 3.26     |

Table 5 shows a relative comparison between the ANFIS and ARIMA models performance parameters. It shows both model’s performance measures in terms of Root Mean Square Error, Mean Absolute Error, Mean Absolute Percentage Error. If we look closely, we see that out of the three configurations we did for the ARIMA
models, all of them performed way better than ANFIS-1 and ANFIS-2, while ANFIS-3 was the best among all the six. ANFIS-1 and ANFIS 2 both used Grid partitioning for input spacing. On the other hand, ANFIS-3 used Subtractive clustering technique. This clearly indicates the high-level of performance due to the change in clustering type or partitioning choice. Though ANFIS-2 utilized 100 epochs, a different membership function with a higher number of membership functions, its performance could not be updated significantly. In fact, it’s only 0.01% improved than ANFIS-3. Meanwhile all the ARIMA models performed well. With the increase in the number of orders of differencing, the performance went well. With the addition of seasonality consideration in the data, the performance improved significantly for the ARIMA-3 configurations.

Figure 3a and b represents the ANFIS 3 configurations’ prediction curve, and the increase in the percentage of cumulative cases. Figure 3a depicts graph of 10 days’

Fig. 3 a Graph representing prediction by ANFIS-3 configuration. Blue dashed line depicts trained data, and blue cross denoted part indicates tested data, while the red stars show predicted values. b Percentage increase in cases from ANFIS-3 model
Fig. 4 a ARIMA 3 configuration. The blue line indicates cases in the trained data, while the amber line shows the validation data for pre-ARIMA and the red line shows the forecasted data for post-ARIMA. b Representation of the percentage increase in cases from ARIMA-3 model (from 31 May till 09 June) prediction in terms of number of cumulative cases. On the other hand, Fig. 4a and b represents the graph of the same indicators produced by the ARIMA 3 configuration.

5 Future Scope of Work

For Prediction of COVID-19 Cases in Bangladesh, we can also use Holt-Winters Exponential Smoothing. It is another suitable forecasting model data scientists use to predict values such as stock market values and other factors that change on a day-to-day basis. The Holt-Winters Exponential Smoothing is also called Triple Exponential Smoothing, as it adds seasonality factor to the existing time series. It has two types of seasonality: Additive and Multiplicative. Additive Seasonality is used in the case of linear exponentiality and Multiplicative Seasonality is used for exponential seasonality [5]. More data for the COVID-19 cases will be added concerning Bangladesh for further analysis as there have been cases every day in this country. Through the addition of data, we can further analyze COVID-19 cases in
Bangladesh. Along with the new addition of data, we can improve the results through ARIMA, ANFIS, and Holt-Winters Smoothing.

6 Conclusion

This work focused on utilizing two widely used forecasting methods, the Adaptive Neuro-Fuzzy Inference System and the Autoregressive Moving Average method, to see their performance in predicting COVID 19 cases in Bangladesh. This paper also compared both the models with respect to a set performance measures and tried to identify the best-suited method for the said prediction. Both of the models have used cumulative cases and percentage change in cumulative cases against dates. ANFIS and ARIMA was tuned with 3 different parameter settings. ANFIS with subtractive clustering configuration performed the best and showed the least amount of error in test data. For ARIMA, we had to start from the basic required parameters to see how the predictions could turn out. Before training the data in both models, some manual effort was involved for calculating percentage change in cumulative COVID-19 cases in Bangladesh. To summarize, it has been found that both the models are good in prediction but requires further improvement. Other coexisting algorithms can be studied and included with each of the methods mentioned above to optimize their performance further.

References

1. Al-qaness, M. A., Ewees, A. A., Fan, H., & Abd El Aziz, M. (2020). Optimization method for forecasting confirmed cases of COVID-19 in China. Journal of Clinical Medicine, 9(3), 674.
2. Ardabili, S. F., Mosavi, A., Ghamisi, P., Ferdinand, F., Varkonyi-Koczy, A. R., Reuter, U., Rabczuk, T., Atkinson, P. M. (2020, Apr 19). Covid-19 outbreak prediction with machine learning. Available from SSRN 3580188.
3. Benmouiza, K., & Cheknane, A. (2019). Clustered ANFIS network using fuzzy c-means, subtractive clustering, and grid partitioning for hourly solar radiation forecasting. Theoretical and Applied Climatology, 137(1–2), 31–43.
4. Benvenuto, D., Giovanetti, M., Vassallo, L., Angeletti, S., & Ciccozzi, M. (2020). Application of the ARIMA model on the COVID-2019 epidemic dataset. Data in brief, 29, 105340.
5. Brownlee, J. (2020). A gentle introduction to exponential smoothing for time series forecasting in Python. Available from https://machinelearningmastery.com/exponential-smoothing-for-time-series-forecasting-in-python/. Accessed June 21, 2020.
6. Dehesh, T., Mardani-Fard, H. A., & Dehesh, P. (2020, Jan 1). Forecasting of COVID-19 confirmed cases in different countries with ARIMA models. medRxiv.
7. Kumar, P., Kalita, H., Patairiya, S., Sharma, Y. D., Nanda, C., Rani, M., et al. (2020, Jan 1). Forecasting the dynamics of COVID-19 pandemic in top 15 countries in April 2020: ARIMA model with machine learning approach. medRxiv.
8. Mohaddes, S. A., & Fahimifard, S. M. (2015). Application of adaptive neuro-fuzzy inference system (ANFIS) in forecasting agricultural products export revenues (case of Iran’s agriculture sector). Journal of Agricultural Science and Technology, 17, 1–10.
9. Pal, R., Sekh, A. A., Kar, S., & Prasad, D. K. (2020, Mar 31). Neural network based country wise risk prediction of COVID-19. arXiv preprint arXiv:2004.00959.
10. Perone, G. (2020, Jan 1). An ARIMA model to forecast the spread and the final size of COVID-2019 epidemic in Italy. medRxiv.
11. Wadi, S. A., Almasarweh, M., Alsaraireh, A. A., & Aqaba, J. (2018). Predicting closed price time series data using ARIMA model. Modern Applied Science, 12(11).
12. Worldometer. (n.d.). Covid 19 situation report Bangladesh. Retrieved from www.worldometers.info.
13. Zhao, S., Musa, S. S., Lin, Q., Ran, J., Yang, G., Wang, W., et al. (2020). Estimating the unreported number of novel coronavirus (2019-nCoV) cases in China in the first half of January 2020: A data-driven modelling analysis of the early outbreak. Journal of Clinical Medicine, 9, 388.
14. Zheng, D., Eseye, A. T., Zhang, J., & Li, H. (2017). Short-term wind power forecasting using a double-stage hierarchical ANFIS approach for energy management in microgrids. Protection and Control of Modern Power Systems, 2(1), 13.
15. Rebane, J., Karlsson, I., Denic, S., & Papapetrou, P. (2018). Seq2Seq RNNs and ARIMA models for cryptocurrency prediction: A comparative study. SIGKDD Fintech, 18, 2–6