

Received December 17, 2020, accepted January 5, 2021, date of publication January 11, 2021, date of current version January 20, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3050852

# A Novel Bayesian Optimization-Based Machine Learning Framework for COVID-19 Detection From Inpatient Facility Data

MD. ABDUL AWAL<sup>1</sup>, MEHEDI MASUD<sup>2</sup>, (Senior Member, IEEE),  
MD. SHAHADAT HOSSAIN<sup>3</sup>, ABDULLAH AL-MAMUN BULBUL<sup>1</sup>,  
S. M. HASAN MAHMUD<sup>4</sup>, AND ANUPAM KUMAR BAIRAGI<sup>5</sup>, (Member, IEEE)

<sup>1</sup>Electronics and Communication Engineering Discipline, Khulna University, Khulna 9208, Bangladesh

<sup>2</sup>Department of Computer Science, College of Computers and Information Technology, Taif University, Taif 21944, Saudi Arabia

<sup>3</sup>Department of Quantitative Sciences, International University of Business Agriculture and Technology, Dhaka 1230, Bangladesh

<sup>4</sup>School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China

<sup>5</sup>Computer Science and Engineering Discipline, Khulna University, Khulna 9208, Bangladesh

Corresponding author: Md. Abdul Awal (m.awal@ece.ku.ac.bd)

This work was supported by Taif University Researchers Supporting Project number (TURSP-2020/10), Taif University, Taif, Saudi Arabia.

**ABSTRACT** The whole world faces a pandemic situation due to the deadly virus, namely COVID-19. It takes considerable time to get the virus well-matured to be traced, and during this time, it may be transmitted among other people. To get rid of this unexpected situation, quick identification of COVID-19 patients is required. We have designed and optimized a machine learning-based framework using inpatient's facility data that will give a user-friendly, cost-effective, and time-efficient solution to this pandemic. The proposed framework uses Bayesian optimization to optimize the hyperparameters of the classifier and ADaptive SYNthetic (ADASYN) algorithm to balance the COVID and non-COVID classes of the dataset. Although the proposed technique has been applied to nine state-of-the-art classifiers to show the efficacy, it can be used to many classifiers and classification problems. It is evident from this study that eXtreme Gradient Boosting (XGB) provides the highest Kappa index of 97.00%. Compared to without ADASYN, our proposed approach yields an improvement in the kappa index of 96.94%. Besides, Bayesian optimization has been compared to grid search, random search to show efficiency. Furthermore, the most dominating features have been identified using SHapely Adaptive exPlanations (SHAP) analysis. A comparison has also been made among other related works. The proposed method is capable enough of tracing COVID patients spending less time than that of the conventional techniques. Finally, two potential applications, namely, clinically operable decision tree and decision support system, have been demonstrated to support clinical staff and build a recommender system.

**INDEX TERMS** COVID-19, ADASYN, Bayesian optimization, classification, inpatient's facility data.

## I. INTRODUCTION

The world is currently experiencing a pandemic situation due to the extensive spreading of the novel coronavirus disease namely, COVID-19. It is an acute respiratory syndrome triggered by the Severe Acute Respiratory Syndrome Coronavirus 2 (SARS-CoV-2), which was primarily detected in Wuhan under the Hubei province of China in late 2019.

The associate editor coordinating the review of this manuscript and approving it for publication was Bilal Alatas<sup>1</sup>.

Considering the alarming rate of infection and death from the COVID-19, World Health Organization (WHO) announced the COVID-19 as a pandemic disease in March 2020 [1]–[3]. As per the WHO report on the COVID-19 on August 04, 2020, about 18,142,718 people have been infected due to COVID-19 [4]. Among them, about 691,013 people died so far. Due to its high contagious nature, both the COVID-19 infection and death toll are rapidly increasing.

In most cases, this disease spreads from man to man via respiratory droplets, transmitted from individual to

individual via air or any other surfaces. This virus lives multiple hours to multiple days on a suitable surface at room temperature [5], [6]. As suggested by WHO, the COVID-patient should get himself isolated from others as early as possible to resist its transmission. The COVID-19 should be detected as early as possible, reducing life, livelihood, and the economy. But a critical issue is the broad maturation period of the COVID-19 that varies from 3 to 14 days. The usual symptoms of this disease include fever, cough, dyspnea, loss of smell, loss of taste, diarrhoea, etc. [7], [8]. People affected by COVID-19 should go through a fruitful, real-time, fast, and accurate screening scheme to ensure timely treatment, isolation, and safety for the patient.

Many pieces of research are going on to find out efficient and speedy COVID-19 detection schemes in different dimensions. The Reverse Transcription Polymerase Chain Reaction (RT-PCR) is a COVID-19 detection scheme that has shown its efficiency and has been practised worldwide. Using samples like the nasal or oral pharyngeal swab, this method can competently detect coronavirus and has attained the gold-standard banner. However, these testing kits fail to meet the mounting demand due to its limited supply, especially in developing countries [9]. Another drawback of this method is that it requires an extended period, ranging from one to two days. Moreover, the situation is even worse in rural areas, because people from remote areas get the results after two or more days, even after a week [10]. This extended period increases the vulnerability of the spreading of COVID-19 as the patient does not usually keep himself isolated from others before getting his result.

To optimize these limitations, the potentiality of Artificial Intelligence (AI) and Machine Learning (ML) algorithms in the analysis, characterization, and classification of different diseases have motivated the researchers to introduce AI and ML in COVID-19 detection. Numerous researches have already been carried out to design a COVID-19 detection model based on AI and ML [7]–[20]. Furthermore, Rajaraman and Antani [10] proposed a COVID-19 detection model based on deep learning (DL) algorithms. Using convolutional neural networks (CNNs), chest X-ray (CXR) data from patients are analyzed in this model to evaluate the presence of the SARS-CoV-2 virus. The model showed about 93% accuracy employing the VGG16 classifier. Another DL and CNNs based automatic COVID-19 detection model was proposed by Makris *et al.* [8]. Diagnosing the CXR data, the model exhibited about 95.9% and 95.00% accuracy engaging VGG16 and the VGG19 classifiers, respectively. A transfer learning-based model was presented by Abbas *et al.* [12] to trace COVID-19. This CNN based model diagnosed the CXR images of patients to check the COVID-19 presence, and the model attained about 97.5% accuracy. He *et al.* [7] presented a DL model for the automatic detection of COVID-19. This model employed the chest computed tomography (CT) images from patients to detect COVID-19. The anticipated 3D CNN model, MNas3DNet41, revealed about 87% accuracy. Jim *et al.* [11] presented an

automatic COVID-19 detection model based on sequential CNN. This model took the CT images in its input to detect COVID-19. The model attained almost 92.5% accuracy along with 94.2% sensitivity and 95.6% specificity. A lot of more automatic COVID-19 detection models have been proposed so far based on the computer-based diagnosis of the CT and CXR images.

Hence, all the anticipated models require CT or CXR data of patients as the key input parameter, only available from diagnostic centres. So, each patient or suspected patient has to visit the diagnostic centre in person to check the presence of COVID-19 in his body. Most of the families in developing countries do not have private transport. Besides, patients from rural areas have to travel a long distance to reach a diagnostic centre. Therefore, they have to use public transport to visit the diagnostic centre to check COVID-19. This will create high vulnerability to COVID-19 spreading, among others. From another point of view, a low percentage of people tested for COVID-19 gets COVID-positive results in most of the countries; as an example, as of July 30, 2020, the positive rate is about 1.30% in France, 22.20% in Bangladesh, 9.90% in Iran, 0.90% in Italy, 7.90% in the USA, 11.10% in India, 2.10% in Russia, and 0.40% in the UK [21]. Visiting the diagnostic or test centre, a large percentage of COVID-19 negative people may meet with COVID-19 positive patients, which will enhance the risk of getting contaminated by COVID-19 disease. So, an inpatient data-based COVID detection will be the best option to avoid these types of risks. Besides, this type of detection will be very user friendly, cost-effective, and time-efficient.

Considering all the above issues, we have proposed a fast and user-friendly model to detect the COVID-19 based on machine learning. A large volume of data on COVID-19 is available in different laboratories and test centres. The dataset comprises other features like age, temperature, pulse rate, systolic and diastolic pressure, fever, cough, loss of smell, runny nose, diabetics, loss of taste, asthma, etc., which are analyzed to design the automatic COVID-19 detection model. The most promising advantage of this model is that it is capable of detecting the COVID-19 within a few minutes as well as help the doctors take adequate precautionary measures while treating the COVID patients. Different classification algorithms such as Linear Discriminant Analysis (LDA), Quadratic-DA (QDA), Naive Bayes (NB), k-Nearest Neighbors (KNN), Decision Tree (DT), Random Forest (RF), eXtreme Gradient Boosting (XGB), Gradient Boosting (GB), Support Vector Machine (SVM), etc. are used to characterize the model. These classifiers have some hyper-parameters, and proper tuning of these hyper-parameter improves the performance of the classification using state-of-the-art global optimizers such as Bayesian optimization [22], Gradient-Based Optimizer (GBO) [23], Slime mould algorithm (SMA) [24], and Harris hawks optimization (HHO) [25] etc. The evaluation of different performance metrics such as accuracy, specificity, sensitivity, etc. for the anticipated model demonstrates higher efficiency in detecting COVID-19.

The contribution and key topics covered by this study are as follows:

- The proposed model can be easily tested on inpatients or inhouse facilities discussed in Section II. Therefore, the patient needs not to visit the clinic to test the COVID-19.
- We have designed a machine learning framework using Bayesian optimization adapted by the ADASYN algorithm to detect COVID-19 which is presented in Section II.D and II.E.
- The state-of-the-art machine learning technique is optimized using our method and compared with other commonly used Grid-search and random search techniques; see Section III.H.
- The proposed method uses the ADASYN algorithm to balance the model, and the effect of ADASYN has also been demonstrated in III.A.
- Using SHapely Adaptive exPlanations (SHAP) analysis, important features are calculated, and the SHAP values are explained to interpret the model in Section III.F.
- A clinically operable decision tree is built that will be helpful for the clinical staff stated in Section IV.A. A decision support system has also been developed to assist the recommender system illustrated in Section IV.B.

The remainder of the paper is organized as follows. In Section II, we discuss the materials and methods used in this work. We present the experimental results in Section III. In Section IV, we present a systematic discussion and comparison of the work with other approaches. Finally, we draw some conclusions in Section V.

## II. MATERIALS AND METHODS

### A. DATA SOURCE

The clinically-driven information on individuals who have undergone through RT-PCR test was collected from the [26]. The dataset, containing 11169 person's data with 2.82% of patients' COVID positive and 97.18% COVID negative tests from the United States, was prepared by Carbon Health (CH) and Braid Health (BH). The CH started RT-PCR testing of a coronavirus in early April 2020. The dataset is compliant with the Health Insurance Portability and Accountability Act (HIPAA) privacy rule's de-identification standard. Five clinical teams worked under the CH. The dataset prepared by the CH covered multiple physiognomies on patients, including Epidemiological (Epi) Factors, comorbidity, vital signs, lab technician-assessed symptoms, patient-stated symptoms. Whereas, two clinical teams gathered the dataset under the BH, which assembled the radiological information containing verdicts, CXR impressions, CXR labels, and CXR link. We haven't used radiological information as most of the patient doesn't have radiological details. The integration of radiological information is beyond the scope of this study, hence excluded from the analysis. The dataset consisted of both positive and negative test results for patients having both one or more symptoms and zero symptoms. In addition to

COVID-19 test results, the complete dataset, available on the GitHub website, contains multiple features of patients such as pulse rate, temperature, age, higher danger introducer occupation, higher danger contacts, diabetics, cancer, asthma, smoker, systole, diastole, diarrhoea, fatigue, fever, losing smell, losing taste, runny nose, headache, muscle pain, pain in the throat, cough, shortness of breath, etc. The vignette of the entire data set has been illustrated through a tabular sketch shown in Figure 1.

From the pictorial depiction (Figure 1), it is much clearer that there are two types of data (numeric and boolean), where the boolean variables are more than three times that of the numeric data. Moreover, the highest age of the patients in this data set is 90 years old, and the extreme values of both systolic and diastolic pressures were dramatically higher than the natural ones. It can be further added that *days\_since\_symptom\_onset* has about 68% missing data, while the percentage of missing data in the entire data set is around 17. Besides the tabular display, as shown in Figure 1, the graphical example the green bars in Figure 2 efficiently reveals that the variables *cough*, *diabetes*, *chd*, *htn*, *cancer*, *asthma*, *COPD*, *autoimmune\_dis*, and *smoker* have no missing data. In contrast, the variable *days\_since\_symptom\_onset* has the highest missing values compared to others.

### B. DATA PRE-PROCESSING

The overall workflow of our study is presented in Figure 3. For data pre-processing, the dataset has been imputed using Multivariate Imputation by Chained Equations (MICE) algorithm [27]. After completing scaling, we used the ADASYN algorithm to balance out COVID and non-COVID datasets. ADAptive SYNthetic (ADASYN) algorithm [28] is an over-sampling method where COVID positive is a rare instance. It helped us generate synthetic data, solving the over-fitting problem. In contrast, the under-sampling process is not the right choice in COVID classification. The majority class (i.e. COVID-no class) is downsampled to the amount minority class (i.e., COVID-yes). This process will reduce the amount of data that drastically cause data inefficiency, and it loses the vital information of COVID-no class. Our COVID data set is not a big dataset, and downsampling could mislead the diagnosis and detection. Compared to other correlated over-sampling methods such as AdaBoost in conjunction with Over/Under-Sampling and Jittering of the data (JOUS-Boost), Synthetic Minority Over-sampling TEchnique (SMOTE), SMOTE-Boost and, DataBoost-IM (DataBoost IMbalanced) algorithm, ADASYN can balance the imbalanced dataset, for example, COVID-19 dataset by reducing the bias introduced by the imbalanced data distribution [28]. Besides, ADASYN shifts the decision boundary to harder to learn examples which ultimately improves the classification accuracy. These two objectives, i.e. (i) bias reduction and (ii) introducing harder to learn neighbourhoods examples, are accomplished through the dynamic weight adjustment and adaptive learning procedure [28].

Factor	Variable	Variable Types	Distinct Count	For Numeric Data			The ratio of Boolean Data (0/1)	Missing Data	Missing (%)
				Maximum	Minimum	Mean			
Epidemiologic al (Epi) Factors	age	Numeric	94	90		42.18		0	0
	high_risk_covid_occupation [Whether patient in a profession with a high risk of exposure.]	Boolean	2				8933/2067	169	1.5
	high_risk_interactions	Boolean	2				6763/2905	1501	13.4
Comorbidities	diabetes	Boolean	2				10924/245	0	0
	chd [Chorony Heart Disease]	Boolean	2				11006/163	0	0
	Htn [Hypertension]	Boolean	2				10195/974	0	0
	cancer	Boolean	2				11079/90	0	0
	asthma	Boolean	2				10513/656	0	0
	Copd [Chronic Obstructive Pulmonary Diseases]	Boolean	2				11139/30	0	0
	autoimmune_dis	Boolean	2				11092/77	0	0
smoker	Boolean	2				10392/777	0	0	
Vitals	temperature	Numeric	66	39.4	34.95	36.88		4627	41.4
	pulse	Numeric	96	150	40	78.73		4644	41.6
	Sys [Systolic]	Numeric	91	235	75	126.04		4618	41.3
	Dia [Diastolic]	Numeric	80	135	36	79.71		4618	41.3
	Rr [Respiratory Rate]	Numeric	20	32	10	15.05		5407	48.4
	Sats [Oxygen Saturation]	Numeric	14	100	76	98.11		4783	42.8
Symptoms	Ctab [Lungs Clear To Auscultation Bilaterally]	Boolean	2				4881/995	5293	47.4
	labored_respiration	Boolean	2				7005/18	4146	37.1
	rhonchi	Boolean	2				3631/880	6658	59.6
	wheezes	Boolean	2				3795/1030	6344	56.8
	days_since_symptom_onset	Numeric	20	300	1	10.14		7596	68.0
	cough	Boolean	2				9034/2120	15	0.1
	fever	Boolean	2				7894/743	2532	22.7
	sob [Shortness of breath]	Boolean	2				9756/1207	206	1.8
	diarrhea	Boolean	2				10396/586	187	1.7
	fatigue	Boolean	2				9318/1675	176	1.6
	headache	Boolean	2				9415/1571	183	1.6
	loss_of_smell	Boolean	2				10753/226	190	1.7
	loss_of_taste	Boolean	2				10744/235	190	1.7
	runny_nose	Boolean	2				10016/966	187	1.7
muscle_sore	Boolean	2				9901/1086	182	1.6	
sore_throat	Boolean	2				9666/1320	183	1.6	

FIGURE 1. Characteristics of the Sample.

The Mathematical explanation behind the ADASYN algorithm is given below:

For illustration, if  $m_l$  and  $m_s$  represent the majority and minority classes, respectively, then the Degree of imbalance of the two classes can be figured as follows:

$$d = \frac{m_s}{m_l} \tag{1}$$

If  $d < d_x$  (where  $d_x$  is the preset threshold for the maximum tolerated imbalance) then the total number of the synthetic minority can be estimated as follows:

$$G = (m_l - m_s)d \tag{2}$$

Here  $d = 1$  means there is a total balance between two classes. If  $r_i = \frac{m_l}{k}$ , where  $k$  is the number of neighbours of each minority, and  $\hat{r} = \sum r_i$  such that  $\sum \hat{r} = 1$ , then the amount of synthetic data to generate for each neighbourhood can be calculated as:

$$G_i = G\hat{r} \tag{3}$$

If  $x_i$  and  $x_u$  are two minority examples within the same neighbourhood, where  $x_u$  is randomly selected, then the new synthetic example,  $s_i$  can be enumerated using the followings:

$$s_i = x_i + (x_u - x_i)\lambda, \tag{4}$$

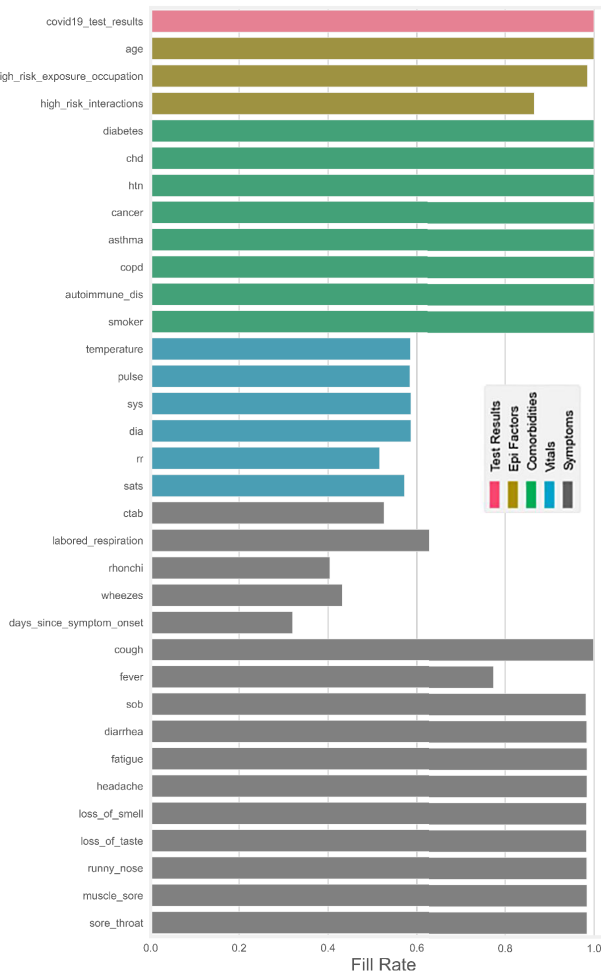


FIGURE 2. Fill rate for all Variables.

where  $x_u - x_i$  is the difference vector in  $n$ -dimensional spaces, and  $\lambda$  is a random number over  $[0, 1]$ .

### C. CLASSIFICATION MODELS

These nine classifiers such as Linear Discriminant Analysis (LDA), Quadratic Linear Discriminant Analysis (QLDA), Naive Bayes (NB), KNN, DT, RF, XGB, GB, and SVM, have been utilized in the proposed machine learning framework. Among nine classifiers LDA, QLDA, NB, KNN, DT and, SVM are common classifiers and also used in COVID-19 classification. RF, XGB and GBC are recent state-of-the-art classifiers. For example, XGB is recently applied to interpret the mortality prediction in COVID-19 patient and proposed a clinically operable simple tree-based tool which can be suitable to take the right decision from an expert point of view [56]. Considering the above rationale, we have used both commonly used classifiers as well as recently updated classifiers in this study. This will allow us to compare the classification performance in different classifiers. Moreover, RF, XGB and GBC classifiers can be explained through SHAP analysis which is very useful to clinical engineers. Finally, it can be seen from the results

that the XGB performed better in most of the classification metrics, and we used SHAP to explain the XGB to interpret the COVID-19 detection.

#### 1) LINEAR DISCRIMINANT ANALYSIS (LDA)

The LDA, introduced by Ronald Aylmer Fisher in 1936 [29], is a productive classification technique. It sorts-out  $n$ -dimensional spaces into 2-dimensional spaces that split-up by hyper-plane. The core objective of LDA is to trace the mean function for each class, and the function is projected on the directions that optimize between-groups variance and reduces within-group variance. The LDA is generated from the conditional distribution of the data  $P(X|Y = k)$  for each class  $k$ , and it optimizes by taking the class  $k$  when the measurements are made on standalone variables for each observation are continuous quantities [30], [31].

#### 2) QUADRATIC LINEAR DISCRIMINANT ANALYSIS (QLDA)

QLDA, an extension of LDA is exploited in machine learning and statistical analysis to classify two or more groups by quadratic discernible using distance-based classification techniques. There is no hypothesis like LDA that the covariance matrix for every class is identical. When the regularity hypothesis is true, the best prospective test for the hypothesis that an assumed measurement is from a given class is the likelihood ratio test. QLDA can be found from the conditional distribution like LDA of the data  $P(X|Y = k)$  like LDA, and it maximizes by selecting the class  $k$  [30], [31]. More precisely, for LDA and QLDA,  $P(X|Y = k)$  is resulting as a multivariate Gaussian distribution with the following equation:

$$P(Y = k) = \left( (2\pi)^{d/2} \left| \sum_k \right|^{1/2} \right)^{-1} \exp \left( -0.5(X - \mu_k)^t \sum_k^{-1} (X - \mu_k) \right), \quad (5)$$

where  $d$  is the number of features [32]. It needs to estimate the class priors  $P(y = k)$  for using LDA and QDA model as classifiers, e.g. the proportion of instances of class  $k$  from the training data, the means  $\mu_k$  and the covariance matrix.

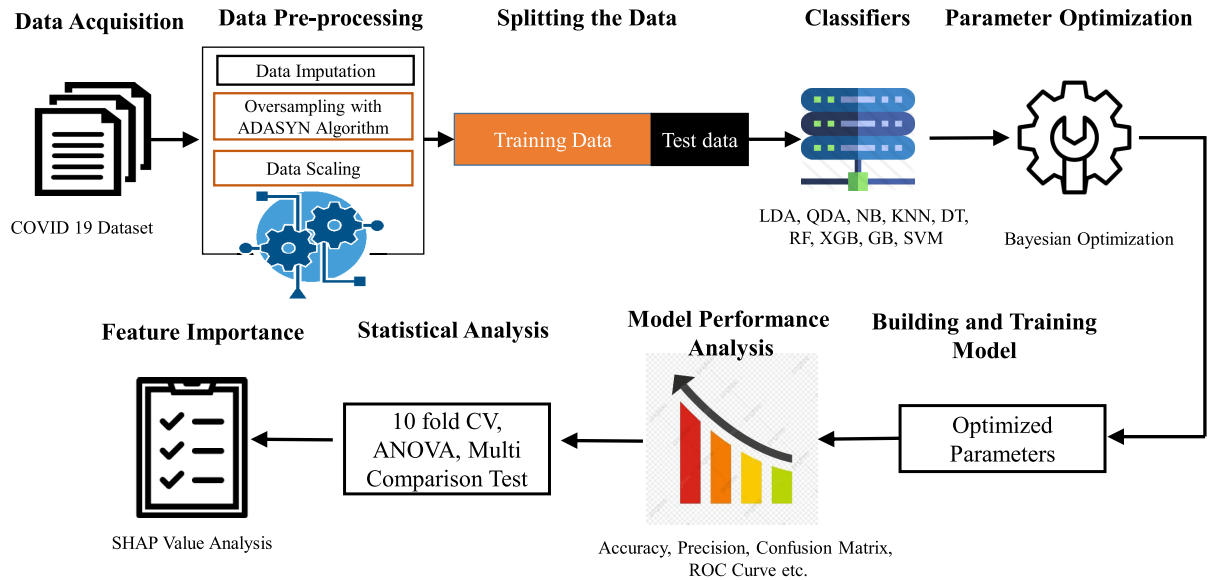
#### 3) NAIVE BAYES (NB)

NB classifier is authoritative and mainly useful in the large dataset. It is used in both machine learning and medical science, especially the diagnosis of different diseases like COVID-19. It is a Bayes' theorem, based on probabilistic classifier objects with the strong independent supposition between the features. It generates conditional probability models that allocate class labels to a given problem [33]. Say,

$$P(\text{Patient}|\text{Covid Positive}) = \frac{P(\text{Covid Positive}|\text{Patient}) \times P(\text{Patient})}{P(\text{Covid Positive})},$$

where,  $P(\text{Patient}|\text{Covid Positive})$ , a conditional probability is the likelihood of the patient occurring that s/he is affected





**FIGURE 3.** The overall workflow of the classification of COVID-19. The first phase is collecting raw data followed by pre-processing, where the raw data is imputed, scaled, and most importantly, the imbalanced data is balanced using ADASYN algorithm. Secondly, the pre-processed data are split into the train and test set used by different classifiers to measure the classification performance. In the next step, Bayesian optimization has been implemented to tune the hyperparameters of the classifiers. This optimized classification model is then tested, and different performance metrics (accuracy, precision, Confusion matrix, ROC, 10-fold cross-validation, ANOVA, and multi-comparison test) have been used for evaluation. Finally, the important features have been efficiently traced using SHAP analysis.

with Covid;  $P(\text{Covid Positive}|\text{Patient})$  is also a conditional probability: the likelihood of the positive COVID occurring that is truly a patient;  $P(\text{Patient})$  is the prior probability of a patient;  $P(\text{Covid Positive})$  is the overall probability of observing COVID positive.

#### 4) K-NEAREST NEIGHBOURS (KNN)

KNN is straightforward simplest algorithms in supervised machine learning technique [34] uses data and classify new data points based on similarity measures with the distance function, be able to apply to solve both classification and regression difficulty. It uses an integer number as 1, -1, or 0 for symbolizing the productivity (labels) of a classification algorithm outputs. KNN is a memory-based classifier; for example, it remembers all the training data-points to predict test data by computing the similarity between an input sample and each training instance. For a given new data point  $x_0$ , it finds the  $k$  training points  $x_r, r = 1, \dots, k$  closest in distance to  $x_0$  and then classify using majority vote among the  $k$  neighbors [32]. For selecting  $k$ , it conducts the KNN algorithm respective times with various values of  $k$  and opts for the  $k$  that reduces the number of errors accurately.

#### 5) DECISION TREE (DT)

DT is a hierarchical flow chart like structure that generate some decision rules. The DT creates a model that predicts the target variable by learning the decision rule from the data feature [35]. The main hyper-parameters of DT are *criterion*, *max\_depth*, *max\_features*. In DT, “Gini” or “entropy” is used as a *criterion*. In contrast, the *max\_depth* is utilized to limit the number of nodes in the tree, and the *max\_features*

represents the number of features to consider while searching for the optimal split. By properly tuning the hyper-parameters of DT (i.e., *criterion*, *max\_depth*, *max\_features*) applied on the COVID training dataset, the classification performance will be efficiently magnified.

#### 6) RANDOM FOREST (RF)

RF is an ensemble learning technique for classification that uses several DTs on different sub-samples of the dataset to improve the classification performance and to control over-fitting [36]. The main hyper-parameters of RF are *criterion*, *max\_depth*, *max\_features*, *n\_estimators*. The *criterion*, *max\_depth*, and *max\_features* have already been discussed in DT. Besides, *n\_estimators* represent the number of DTs in the forest. The performance of RF can be increased by properly tuning the hyper-parameters of RF through optimization.

#### 7) GRADIENT BOOSTING CLASSIFIER (GBC)

GBC is also an ensemble classifier that combines different weak learners (typically DT) into a single strong learner in a forward stage-wise fashion by optimizing the differentiable loss function [37]. Generally, ‘deviance’ or ‘exponential’ is used as a loss function where ‘deviance’ refers to deviance (logistic regression) for classification with probabilistic outputs. For thrashing, ‘exponential’ gradient boosting recaptures the AdaBoost algorithm. Other controlling parameters of GBC contained different parameters such as *n\_estimators*, *learning\_rate*, and *max\_depth* where *n\_estimators* indicate individual boosting stages to accomplish; *learning\_rate* reduces the performance of each tree [32].

**TABLE 1. Classifiers and their controlling parameters or hyperparameters.**

Classifiers (Clf)	Parameters, $z \in Z$	Hyper-parameters name
RF	4	Criterion, max_depth, max_features, n_estimators
RBF-SVM	2	Cost(C) and Gamma( $\gamma$ )
DT	3	Criterion, max_depth, max_features
NB	2	Alpha
GBC	5	Learning_rate, loss, max_depth, max_features, n_estimators
XGB	7	N_estimators, learning_rate, n_jobs, max_depth, Gamma, min_child_weight, colsample_by_tree
KNN	1	Number of Neighbours

8) eXtreme GRADIENT BOOSTING (XGB)

XGB is designed based on the principles gradient boosting framework. It can be used for supervised learning tasks such as regression, classification, and ranking; similarly, it generates a prediction model in the form of an ensemble of weak prediction models. The model in a stage-wise approach is compassed with it as other boosting methods do, and it generalizes them by approving optimization of a random differentiable loss function. The gradient descent is used by ‘Gradient Boosting’ to generate new trees based on all previous trees. It supervises the objective function toward the minimum direction [38]. An objective function has a form, and it divides into training loss and regularization. The mathematical equation has been added as follows:

$$obj(\theta) = L(\theta) + \Omega(\theta), \tag{6}$$

where  $\theta$  denotes the parameters,  $\Omega$  symbolizes the regularization term, and  $L$  is the training loss. The main hyper-parameters of XGB are *N\_estimators*, *learning\_rate*, *n\_jobs*, *max\_depth*, *Gamma*, *min\_child\_weight*, *colsample\_by\_tree*. The hyper-parameters such as *N\_estimators*, *learning\_rate*, *max\_depth* have already been discussed. Besides, *n\_jobs* are relevant to the number of parallel threads used to run XGB; *Gamma* represents the loss required to make a further partition on a leaf of the tree. The *min\_child\_weight* denotes the minimum sum of feature example, i.e., instance weight needed in a child, and *colsample\_by\_tree* is used for the subsampling of columns.

9) SUPPORT VECTOR MACHINE CLASSIFIER (SVC)

SVC is one of the most powerful supervised classifiers and used mostly for data classification in medical diagnosis [39], [40]. It aims to build a decision boundary in such a way that it is as far as possible from the closest data points from each of the classes, which are known as support vectors. For non-linear problems like COVID detection, a Radial Basis function (RBF) kernel is used. For RBF-SVC, the controlling hyper-parameters are Cost(C) and Gamma( $\gamma$ ). The Cost(C) represents the regularization parameter that controls the misclassification of the training instances. Gamma( $\gamma$ ) controls the ‘spread’ of RBF kernel and, therefore, the decision region. The lower value of

Gamma( $\gamma$ ) will broaden the decision region and vice versa. The proper value of C and  $\gamma$  will increase the classification performance, which can be achieved by optimization.

**D. REQUIREMENT OF OPTIMIZATION**

Most of the classifiers used in our entire study have some hyperparameters. The classifier itself is the function of hyper-parameters, and these parameters control the hyper-plane. As an exemplification, XGB requires 7 Hyperparameters, while KNN and DT have one parameter each [Table 1]. Classifier performance indices, e.g., classification accuracy, error, specificity, sensitivity, etc. depend on the proper choice of these parameters. This is an optimization problem, whose general framework can be written as:

$$\lim_{z \in Z} J(Clf(z); Z), \tag{7}$$

where  $z \in Z$  denotes the hyper-parameters  $z_1, z_2, z_3, \dots, z_n$  belongs to  $Z$ . *Clf* denotes the classifiers, e.g. RF, SVM, DT, NB, etc. and  $J(\cdot)$  represents the objective function. This objective function is the user-defined function where users can use different classifier metrics such as classification error or accuracy or other metrics described in the following section of statistical evaluation of classification measures. The general framework of the optimization problem can be interpreted as minimizing the classification objective  $J(\cdot)$  as a function of classifier’s hyperparameters  $z \in Z$ . In this study; mean of the the 10-fold cross-validation error is used as an objective function. We chose one of the state-of-the-art optimization algorithms named Bayesian optimization. This algorithm used a stochastic process, namely, as a Bayesian process, and it tried to find the optimal parameters in a smaller number of iterations saving both memory and time [41].

Although various meta-heuristic algorithms such as GWO, GBO, SMA, and HHO etc. successfully integrated into many applications [42]–[44], hyper-parameter optimization in expensive-to-evaluate objective function e.g., 10-fold cross-validation loss, used in this study, makes it more complicated [45]. Besides, meta-heuristic algorithms require a set of input parameters that need to be found out to obtain an improved performance as the performance of the meta-heuristic algorithms are very sensitive to the

input parameters. Furthermore, comparison among various meta-heuristic algorithms is only valid if the proper input parameters have been set, which requires domain knowledge [46]. Bayesian optimization is used to set the parameters of the meta-heuristic algorithm [45], [46].

The Bayesian optimization algorithm is a global optimization method that is specially designed to deal with such expensive-to-evaluate objective function, which is the population and genetic operator (mutation, cross-over, and selection) free algorithm. Bayesian optimization utilizes a Gaussian process to compute an acquisition function that evaluates the objective function. Besides, Bayesian optimization memorizes its previous evolution and utilize these statistics towards good solutions. It has been recently used in COVID-19 detection using x-ray images [22]. Considering the above rationale, Bayesian optimization has been applied in this study.

To justify further, the proposed Bayesian optimization is compared with the recently proposed Harris Hawk Optimisation algorithm [25]. This popular swarm-based and gradient-free optimization algorithm is based on the cooperative behaviour and chasing styles of Harris' hawks in nature called "surprise pounce" [25]. We have chosen this algorithm for comparison as it is very recent and outperformed by many popular meta-heuristic algorithms such as GWO, Multi-Verse Optimizer, Moth-Flame Optimization, Whale Optimization Algorithm, Bat Algorithm, Cuckoo Search, Firefly Algorithm.

### E. BAYESIAN OPTIMIZATION

Bayesian optimization (BO) is superior to grid search, random search, and manual tuning and therefore used in this study [47]. This algorithm keeps track of the past evaluation results and uses them to form a probabilistic Gaussian model of BO of the objective function and use it to find out the most optimal hyper-parameters; as an exemplar, in the case of RBF-SVM, the hyper-parameters are  $C$  and  $\gamma$ . The BO algorithm selects  $C$  and  $\gamma$  for which objective function  $J(RBFSVM; (C, \gamma))$  provides the minimum value. Note that, the classification error is used as an objective function. The BO algorithm is given below:

**Step 1:** Build a Gaussian probability model of the objective function. In this study, classification error is the objective function.

**Step 2:** Find the controlling parameters or hyper-parameters that perform best on the Gaussian process.

**Step 3:** Apply these hyper-parameters to the true objective function.

**Step 4:** Update the Gaussian model incorporating the new results.

**Step 5:** Repeat Step 2-4 until maximum iteration is reached.

The Mathematics behind the Bayesian Optimization for  $X = (x_1, x_2, x_3, \dots, x_n)$  independent features and  $y$  target variable

is given below:

$$P(y|X) = \frac{P(X|y)P(y)}{P(X)} \quad (8a)$$

$$= \frac{P(x_1|y)P(x_2|y) \dots P(x_n|y)P(y)}{P(x_1)P(x_2) \dots P(x_n)} \quad (8b)$$

$$= \frac{P(y) \prod_{i=1}^n P(x_i|y)}{P(x_1)P(x_2) \dots P(x_n)} \quad (8c)$$

$$= \left[ \frac{1}{P(x_1)P(x_2) \dots P(x_n)} \right] \times P(y) \prod_{i=1}^n P(x_i|y), \quad (8d)$$

Since all the variables except the target variable are independent,  $P(x_1)P(x_2) \dots P(x_n) = \text{Constant}$ , Then Eq. (8d) can be simplified as:

$$P(y|x_1, x_2, \dots, x_n) \propto P(y) \prod_{i=1}^n P(x_i|y), \quad (9)$$

Now, from Eq. (9), we find the probability of a given set of inputs for all possible values of the target variable  $y$  and pick up the output with maximum probability:

$$y = \underset{y}{\operatorname{argmax}} P(y) \prod_{i=1}^n P(x_i|y), \quad (10)$$

### F. STATISTICAL EVALUATION OF CLASSIFICATION METRICS

We have used several performance evaluation metrics to evaluate the performance of the proposed framework. The accuracy (ACC), error, false-positive rate (FPR), sensitivity (SE), specificity (SP), positive predictive value (PPV), Matthew's correlation coefficient (MCC), F1\_score, and Kappa index can be calculated from confusion matrix [48], [49]. A lower value of error and FPR, and a higher value of ACC, SE, SP, PPV, MCC, F1\_score, and Kappa index indicate a better model. Besides, 10-fold cross-validation has been used [52] on the overall dataset. The most significant point should be mentioned here that the box-plot and Analysis of Variance (ANOVA) test are typically executed, relying on the 10-fold cross-validation result. The statistical significance is determined by the p-value derived from the ANOVA test [50], [51]. Furthermore, the receiver operating characteristic (ROC) curve and the area under the ROC curve (AUC) has also been used to evaluate the performance of the classifier. The recall rate vs the decision boundary curve has been used to examine the performance. In this study, we have used the value of 0.5 as the decision boundary threshold to provide the same importance to COVID-yes and COVID-no classes.

### G. FEATURE IMPORTANCE USING SHAP VALUES

The SHapely Adaptive exPlanations (shortly known as SHAP), proposed in recent papers by Lundberg and Lee [53], are calculated for any tree-based model. The SHAP values from Game Theory to attribute  $\phi_i$  value to each feature can be mathematically ascertained using the following



formula [54]:

$$\phi_i = \sum_{S \in N \setminus i} \frac{|S|!(M - |S| - 1)!}{M!} [f_x(S \cup \{i\}) - f_x(S)], \quad (11)$$

where  $M$  is the total input features,  $N$  is the set of all input features, and  $S$  is a subset of the input features.

- In this plot, all the variables are ranked in descending order.
- The horizontal line (x-axis) quantifies how much the value is associated with a higher or lower prediction. All the left-sided points represent the observations shifting the predicted value in a negative direction. In contrast, the points on the right contribute to shifting the prediction in a positive direction. All the features are on the left y-axis.
- The color shows whether that variable is high (in red) or low (in blue) for that observation.

### III. EXPERIMENTAL RESULTS

In this paper, the Bayesian optimization has been used along with and without the ADASYN algorithm. In the case of ADASYN, sufficient adaptive synthetic data has been created to eliminate the imbalanced nature among the majority and the minority classes. Firstly, the effect of ADASYN has been evaluated along with ROC, shown in section III.A. The balanced model has also been tested on the original test data in section III.B. Box-plot and ANOVA are presented in section III.C using cross-validation accuracy to evaluate the statistical significance. The Recall rate vs. decision boundary curve and Bootstrap ROC with ADASYN are discussed in sections III.D and III.E, respectively. Then, the evaluation of feature importance using SHAP and the analysis of SHAP values have been presented in sections III.F and III.G, respectively. Finally, the performance of Bayesian optimization has been compared with the Grid search and random search in section III.H.

#### A. BAYESIAN OPTIMIZATION WITH AND WITHOUT ADASYN

The newly obtained balanced dataset has been utilized; 67% of the total dataset is used for training and validation, and 33% is used for testing. After that, multiple classifiers are used, and various statistical measurements are presented. The effect of ADASYN has been experimented and validated in this subsection.

To begin, in the upper portion of Table 2, the performance analysis for the COVID Dataset with the utilization of the ADASYN algorithm has been demonstrated. It can be seen that; RF provides the highest classification performance. However, the performance of XGB and GBC is very close to RF. LDA and QLD show the worst classification performance among various classifiers presented in Table 2. The same AUC value of 99.70% is observed among these three classifiers, as shown in Figure 4. To demonstrate the effect of the ADASYN algorithm, the original unbalanced dataset is used. The dataset is also divided in the same manner,

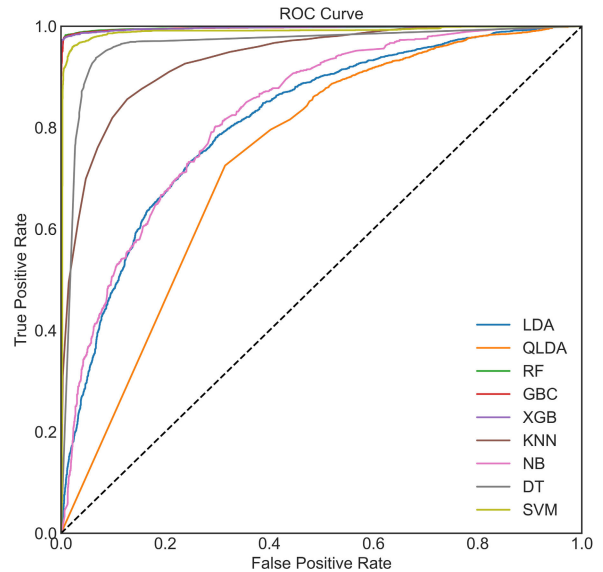


FIGURE 4. ROC Curve with ADASYN.

i.e., 67% of the total dataset is used for training and validation, and 33% is used for testing. We rerun the optimized code on this dataset, and the results on the test dataset without ADASYN is presented in the lower portion of Table 2. It can be observed that the highest accuracy of 97.17% is obtained by RF, which is close to the classification accuracy using RF with ADASYN. This could happen in the imbalance dataset. Therefore, accuracy is not a good performance indicator. The Kappa index, MCC, and AUC are more robust and reliable indicators in this case.

It can be seen that the highest Kappa, MCC, and AUC values of 8.96%, 9.36% using NB, and 75.80% using XGB (Figure 5), respectively, are obtained. Compared to the upper portion of Table 2, i.e., results with ADASYN, the Kappa,

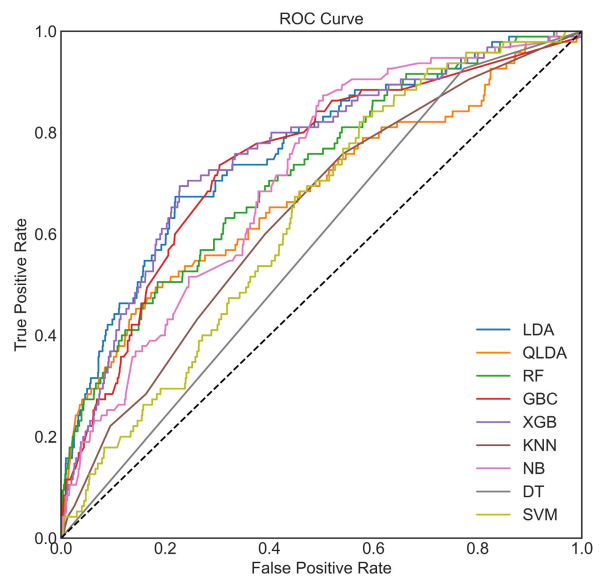


FIGURE 5. ROC curve without ADASYN. Note that the optimized model has not been created by using a balanced dataset.

**TABLE 2.** Classification performance (in %) on the COVID dataset with and without ADASYN.

With ADASYN										
Classifiers	ACC	Error	F1_score	FPR	Kappa	MCC	PPV	SE	SP	AUC
LDA	73.69	26.31	75.11	31.55	47.35	47.60	71.68	78.88	68.45	81.30
QLDA	53.89	46.11	16.83	0.93	8.29	18.90	90.96	9.27	99.07	73.90
KNN	80.01	19.99	76.52	4.51	60.10	63.20	93.56	64.72	95.49	93.50
NB	73.74	26.26	75.60	33.48	47.43	47.91	70.98	80.87	66.52	82.70
DT	93.53	6.47	93.55	6.25	87.06	87.06	93.80	93.31	93.75	96.20
RF	98.59	1.41	98.61	2.08	97.19	97.20	97.97	99.26	97.92	99.70
XGB	98.50	1.50	98.52	2.30	97.00	97.02	97.76	99.29	97.70	99.70
GBC	98.50	1.50	98.52	2.02	97.00	97.01	98.02	99.02	97.98	99.70
SVC	96.60	3.40	96.63	3.51	93.20	93.20	96.54	96.71	96.49	98.90
Without ADASYN										
LDA	96.57	3.43	98.25	84.21	19.11	20.14	97.58	98.93	15.79	76.20
QLDA	17.96	82.04	27.22	7.37	0.56	3.86	98.66	15.79	92.63	68
KNN	97.14	2.86	98.55	100.00	0.06	0.30	97.16	99.97	0.00	64
NB	96.06	3.94	97.99	91.58	8.96	9.36	97.36	98.62	8.42	71
DT	97.14	2.86	98.55	100.00	0.06	0.30	97.16	99.97	0.00	57.60
RF	97.17	2.83	98.56	98.95	1.95	6.94	97.19	99.97	1.05	71.90
XGB	97.14	2.86	98.55	100.00	0.06	0.30	97.16	99.97	0.00	75.80
GBC	97.14	2.86	98.55	100.00	0.06	0.30	97.16	99.97	0.00	74.20
SVC	97.14	2.86	98.55	100.00	0.06	0.30	97.16	99.97	0.00	63.40

**TABLE 3.** Classification performance (in %) on the original test data of COVID.

Classifier	ACC	Error	F1_score	FPR	Kappa	MCC	PPV	SE	SP	AUC
LDA	78.42	21.58	87.66	36.84	9.69	16.75	98.66	78.87	63.16	77.20
QLDA	11.82	88.18	16.97	1.05	0.51	4.76	99.67	9.28	98.95	67.20
KNN	65.62	34.38	78.53	3.16	8.98	21.16	99.86	64.71	96.84	94.40
NB	79.68	20.32	88.55	61.05	5.10	8.27	97.84	80.87	38.95	68.60
DT	93.38	6.62	96.50	27.37	35.65	40.97	99.16	93.98	72.63	88.80
RF	98.54	1.46	99.25	20.00	74.87	74.99	99.41	99.08	80.00	97.10
XGB	98.63	1.37	99.29	24.21	75.08	75.08	99.29	99.29	75.79	96.40
GBC	91.76	8.24	95.61	25.26	30.95	37.68	99.21	92.26	74.74	86.50
SVC	96.75	3.25	98.30	2.11	61.57	66.28	99.94	96.71	97.89	98.90

MCC, and AUC values are 88.23%, 87.84%, and 23.90% times lower ADASYN algorithm is not applied, respectively. This can be happened due to an imbalanced model. This significant improvement using ADASYN concludes that classification performance can significantly be improved through directly applying the ADASYN algorithm.

### B. RESULTS USING ORIGINAL TEST DATA ONLY

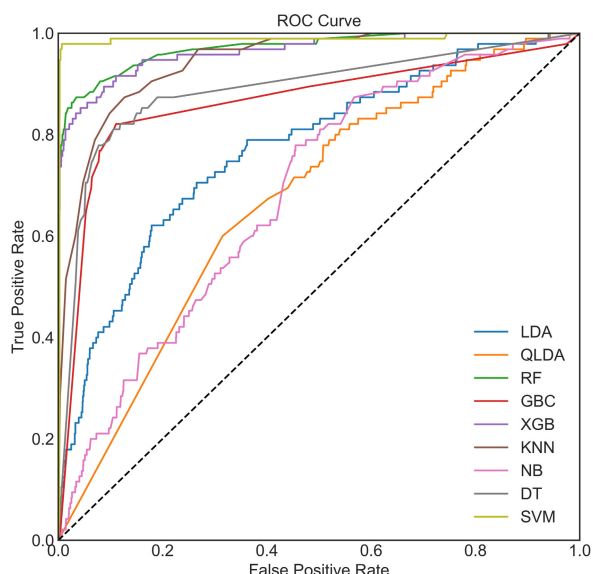
So far, we have seen the effect of ADASYN on classification performance. The ADASYN is an oversampling method, and the synthetic data is mixed with original test data during data balancing. Therefore, it could be argued that what are the

results of the balanced model on the original test data only where synthetic data is not mixed?

To answer this question, balanced and Bayesian-optimized models have been applied to the original test data. Different performance measures, such as accuracy, sensitivity, specificity, and ROC, are presented in Table 3 and Figure 6. It can be seen that XGB provides the highest accuracy, error, F1\_score, FPR, Kappa, MCC and sensitivity of 98.63%, 1.37%, 99.29%, 24.21%, 75.08%, 75.08%, and 99.29%, respectively. In contrast, SVC provides the highest PPV, specificity, and AUC of 99.94%, 97.89% and, 98.90%, respectively. It can also be seen that XGB performs the best in most of the classification metrics presented in Table 3.

**TABLE 4.** The accuracy score (in %) of the different optimized classifiers using 10-fold cross-validation.

	LDA	QLDA	KNN	NB	DT	RF	XGB	GBC	SVC
Fold-1	76.03	20.39	68.16	82.11	89.98	95.44	95.98	89.09	94.10
Fold-2	75.45	20.16	69.62	83.96	89.61	96.33	96.24	92.03	95.25
Fold-3	78.87	20.05	67.68	80.75	91.23	96.87	96.69	92.66	95.17
Fold-4	78.23	20.88	67.47	80.91	91.40	96.95	96.77	92.12	95.61
Fold-5	76.28	23.37	69.29	82.27	90.15	96.96	96.96	91.67	95.70
Fold-6	76.81	19.88	66.97	82.27	91.76	96.42	96.60	91.14	94.36
Fold-7	78.27	20.48	66.19	79.79	89.18	96.60	97.23	89.54	93.92
Fold-8	77.15	22.76	68.19	81.81	91.31	96.42	97.22	90.23	95.43
Fold-9	77.06	21.77	67.38	81.45	91.22	96.06	96.42	90.68	95.70
Fold-10	78.09	24.51	67.62	80.59	90.16	96.60	96.87	89.27	94.72
Average	77.22	21.43	67.86	81.59	90.60	96.46	96.70	90.84	95.00



**FIGURE 6.** ROC Curve for COVID on original test data only using each model. The optimized model has been created by using a balanced dataset and then applied to the original test dataset.

Furthermore, these results are mostly inclined with ADASYN results (upper portion of Table 2), and results are significantly better than without ADASYN in all classification measures. The ROC curve shown in Figure 6 is also visually very close to Figure 4. Note that the same test dataset has been used without ADASYN (i.e., in the lower portion of Table 2) and in Table 3 for a fair comparison. Finally, it can be concluded that a balanced model can significantly improve the performance of the COVID dataset and XGB shows the best classifiers. The confusion matrix of the best performing balanced model with ADASYN and with original test data have been presented in Figure 7 to show how much COVID and Non-COVID patients are correctly classified.

**C. K-FOLD CROSS-VALIDATION**

In the standard train-test-split method, generally, a small portion of the data is taken as the test set, and the total dataset is

not tested. To overcome this issue, the k-fold cross-validation (CV) is one of the helpful techniques exploited to test the effectiveness of machine learning models. It is also a re-sampling procedure to evaluate, and  $k = 10$  is used in this study. The first fold is used for testing, and the remaining folds are used for training and repeated ten times to test the total dataset fold-by-fold basis. The 10-fold cross-validation result is presented in Table 4, where the classification result of each fold is shown. The final row provides the average classification accuracy of the 10-fold results. From the Table 4, it is observed that the least score has been obtained using QLDA, whereas the XGB touched the mountain point, grabbing a score of 96.70% and RF has attained an average accuracy of 96.46%. On the other side, the classification performance using Decision Tree, SVC, and GBC was less than XGB and RF but above 90%. Note that, the data processed by ADASYN is used only to train the classifier, but the original test is used during testing and performance comparison.

Figure 8(a) showed the accuracy of different classifiers using the COVID original dataset using a box-plot. Here one-way ANOVA provided a  $p$ -value of  $3.32 \times 10^{-107}$  for the original COVID test dataset, which is statistically significant ( $p < 0.005$ ). It also provided an interactive plot of multiple comparisons of means in Figure 8(b) that showed the highest mean accuracy from XGB that is statistically significant from seven classifiers (GBC, DT, SVC, NB, KNN, QLDA, and LDA). In contrast, it is statistically not significant from RF, because the mean of RF is almost identical. Note that, Figure 8(b) is an interactive plot where the significance of different classifiers can be visualized by clicking on the specific classifier level. For instance, RF is blurred (shown in grey) defining its insignificance as XGB is selected. Similarly, GBC and DT will also exhibit statistical insignificance if one of them is selected.

**D. RECALL RATE VS. DECISION BOUNDARY CURVE**

The recall rate, in general, depends on the decision boundary using a certain threshold. To exemplify, the recall rate vs. decision boundary curve displayed in Figure 9(a), where

**Confusion Matrix Using RF**

	Covid	Non-Covid	
Covid	3150 48.7%	24 0.4%	99.2% 0.8%
Non-covid	67 1.0%	3233 49.9%	98.0% 2.0%
Overall	97.9% 2.1%	99.3% 0.7%	98.6% 1.4%
	Covid	Non-Covid	Overall
	<b>Target Class</b>		

(a)

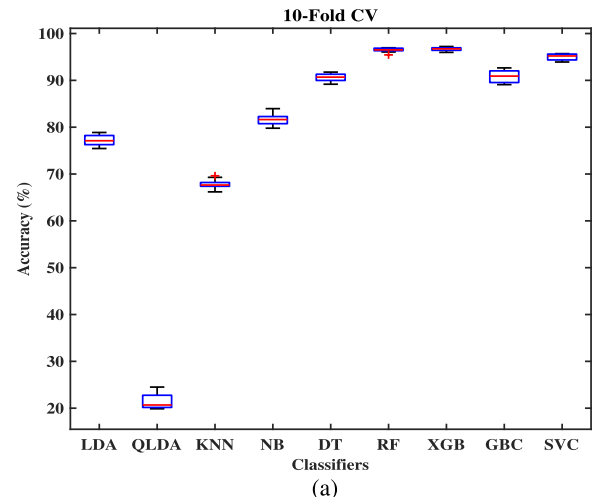
**Confusion Matrix Using XGB**

	Covid	Non-Covid	
Covid	72 2.1%	23 0.7%	75.8% 24.2%
Non-covid	23 0.7%	3233 96.5%	99.3% 0.7%
Overall	75.8% 24.2%	99.3% 0.7%	98.6% 1.4%
	Covid	Non-Covid	Overall
	<b>Target Class</b>		

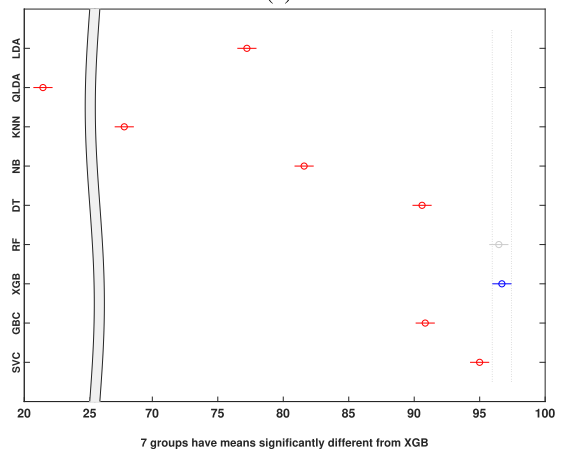
(b)

**FIGURE 7.** Confusion matrix of the balanced model applied in (a) COVID test Dataset with ADASYN, (b) original COVID test Dataset only. Figure 7(a) depicts the percentage of the correct classification in with the first two diagonal cells generated by the trained network. The numbers of patients who are correctly classified as a COVID and non-COVID were 3150 and 3233, corresponding to 48.7% and 49.9% in each group’s patients, respectively. Likewise, the numbers of patients who are incorrectly classified as a COVID and non-COVID were 24 and 67, with 0.4% and 1.0% correspondingly among all patients in each group. Similarly, the overall 99.2% were correctly, and 0.8% were incorrectly classified COVID, and non-COVID were overall, 98.0% and 2.0% correctly and incorrectly classified accordingly. In the case of prediction, the correct overall predictions for COVID and non-COVID were 97.9% and 99.3%, respectively. On the other hand, the incorrect results for COVID and non-COVID were 2.1% and 0.7%. Similarly, we can also interpret Figure 7(b).

0.5 decision boundary threshold ( $T$ ) has been used for the “COVID-19-yes” class. The recall rate of QLDA is about 0.98 at default threshold  $T = 0.5$ , meaning that about 98% times this optimized classifier can truly classify the



(a)



(b)

**FIGURE 8.** Box-plot for (a) COVID Dataset and (b) multi-comparison test. Note that (b) is a graphical user interface tool by which one can test the statistical significance of any classifiers. Here we only show the effect of XGB. The effect of other classifiers can also be interpreted in the same way.

“COVID-19-yes”. The XGB and RF provided a moderate performance of around 0.75 at default threshold  $T = 0.5$  defining the “COVID-19-yes” class. The SVC shows the third highest performance of around 0.90. In contrast, the recall rate of NB at this threshold is 0.25, meaning that only 25% times NB can truly classify the “COVID-19-yes” class. A similar scenario is observed for the LDA classifier.

On the other hand, looking at Figure 9(b), the recall rate of QLDA is drastically falling to a value of 0.1 at  $T = 0.5$ , revealing that only 10% times QLDA can classify the “COVID19-no” class. The recall rate of XGB, GBC, and RF is about 0.99 at this threshold whereas the recall rate of SVC is 0.90. Finally, considering both “COVID19-yes” and “COVID-19-no” classification using recall rate vs. decision threshold measure, it can be concluded that SVC, XGB, and, RF provide the satisfactory recall rate among different optimized classifiers predicting both classes.



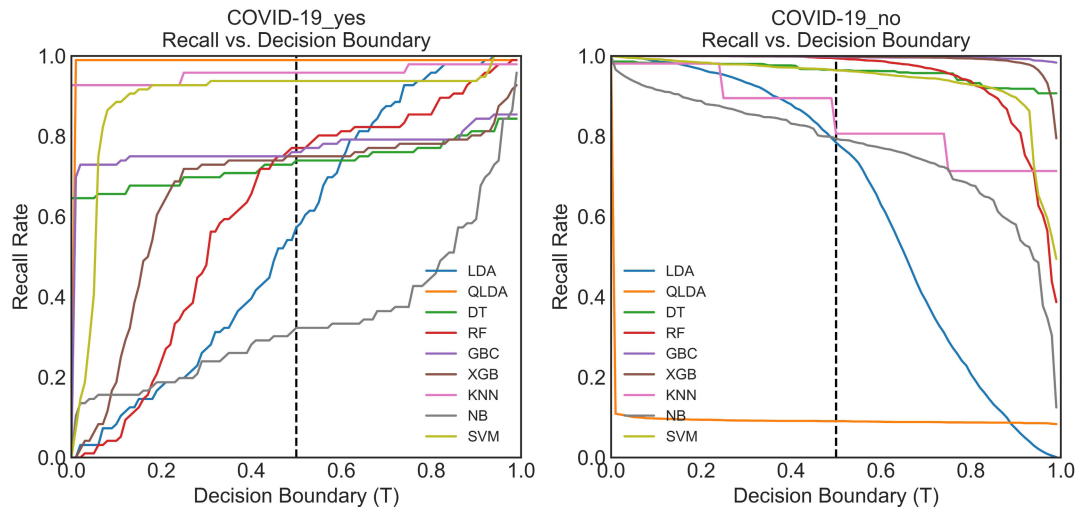


FIGURE 9. Recall rate vs. decision boundary curve for (a) COVID positive and (b) COVID negative.

**E. BOOTSTRAP ROC WITH ADASYN**

To determine whether the optimized model is highly sensitive to training data or not, bootstrapping is performed on the XGB model as it is the best performing model. This gives  $N_{boot}$  XGB having slightly different discriminative abilities. To show the error, three ROC curves are plotted in Figure 10; the middle one represents the average ROC where upper and lower curves represent the 95% confidence interval (CI). To obtain this bootstrap ROC,  $N_{boot} = 100$  XGB models are trained and mean AUC of 0.98 with an upper and lower confidence interval of 0.97 and 0.99, respectively, are obtained. This indicates that training is not highly sensitive to the training dataset.

**F. FEATURE IMPORTANCE USING SHAP**

In a variable importance plot, the most significant variables are sorted in descending order. The top variables contribute more to the model than the bottom ones and thus have high predictive power. By way of example, “fever”, “cough”, “high\_risk\_exposure\_occupation”, “high\_risk\_interactions”, “wheezes” are the most important features, where “fever” touched the mountain point in this case [shown in Figure 11]. Simultaneously, “pulse” and “sore\_throat” received the least importance in classifying the COVID-19 contaminated patients.

**G. SHAP VALUE ANALYSIS**

From the pictorial example of SHAP analysis [Figure 12] for training data, it can be summarized that the three features, “fever”, “cough”, “high\_risk\_exposure\_occupation” and “loss\_of\_smell” have a massive positive impact is on the target variable. The “high” comes from the red colour, and the “positive” impact is shown on the X-axis. Whereas, we conclude by mentioning that the features “ctab” and “wheeze” are highly negatively correlated with the target variable. In this way, all the variables can be efficiently explained. It should be mentioned that the behaviour of the

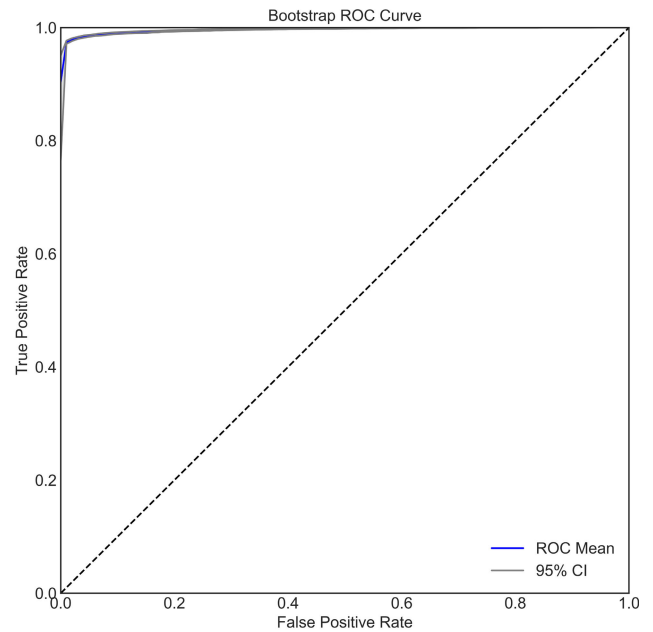


FIGURE 10. Bootstrap ROC curve of the COVID dataset using XGB with 95% CI.

XGB model is defined by the SHAP and are not necessarily causal in the real world. In other word, SHAP values do not provide the causality; it only describes the model behaviour and the behaviour of the data used to build the model [55]. As the model does not predict all the COVID patients accurately, it is plausible to get some false positives and false negatives. However, the SHAP value can able to explain such results, and the summary plot will be helpful to explain those results.

**H. PERFORMANCE ON THE GRID SEARCH, RANDOM SEARCH, BAYESIAN OPTIMIZATION AND HARRIS HAWKS OPTIMIZATION**

We propose to use Bayesian optimization techniques in our framework, and therefore, it is logical to compare

TABLE 5. Comparative search techniques.

Optimization techniques	Parameters evaluated	The overall time is taken (in Sec)*	Cross-validation score (%)	Test score (%)
Grid Search	6561	10473.740	97.39	98.00
Random Search	30	162.794	97.97	98.10
Bayesian optimization	30	675.389	98.00	98.20
Harris Hawks optimization	200	6204.80	98.39	98.00

\*In case of grid search: time is taken to iterate overall parameter combination.

\*In case of random search and Bayesian optimization: time is taken to go over a predefined number of iterations (30)

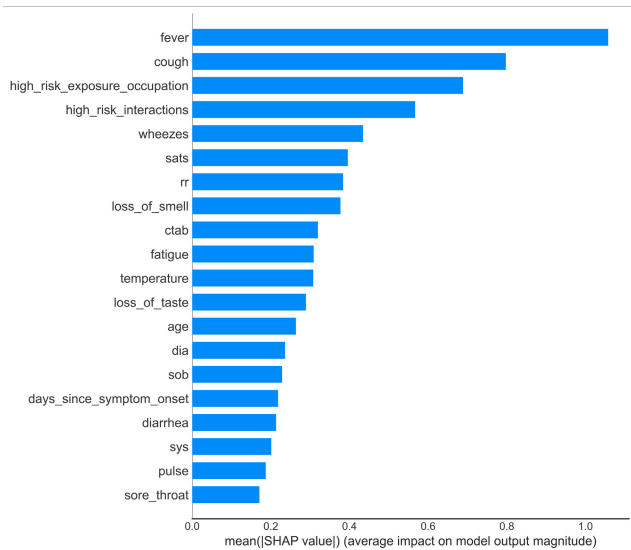


FIGURE 11. Feature importance plot using SHAP for XGB.

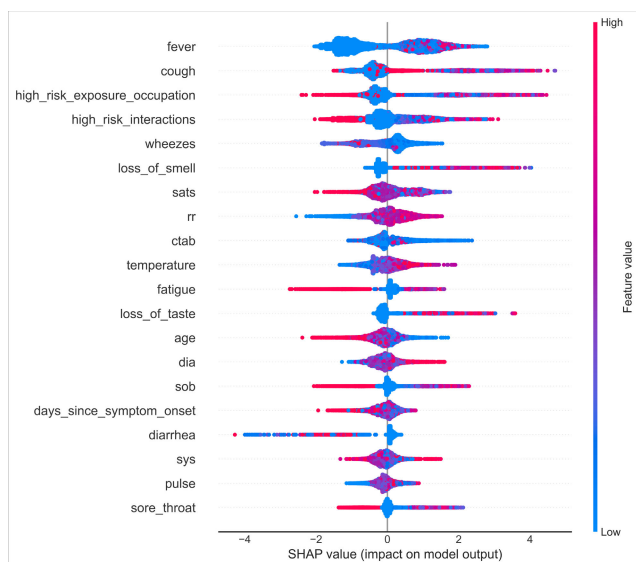


FIGURE 12. The SHAP variable importance plot of training data using XGB.

the Bayesian optimization algorithm with commonly used parameter search algorithms. Two popular and widely used algorithms, namely, grid search and random search, compare

with our proposed techniques. Table 5 presents the comparison of different search algorithms in terms of several parameters evaluated; the overall time is taken (in sec.) to complete the program, cross-validation accuracy score, test score. All the simulations were run on Intel core i9 computer having 64GB RAM and used the XGB model. It can be seen that it takes 10473.740 Sec. to complete the simulation using grid search, whereas random search and proposed Bayesian optimization take only 162.794 Sec. and 675.389 Sec, respectively. Furthermore, the random search and Bayesian algorithm take 30 parameters each, while the grid search requires more parameters, which is 218 times than that of others. The test score using Bayesian optimization is 98.20%, which is better than grid search, random search.

The pictorial depiction of the comparative search methods has also been given in Figure 13, from where it can be added that at the initial stage, the accuracy of Random Search was nearly 97.50%, which was almost stable up to 12 iterations. Then, with a single iteration, it takes a sharp change in its accuracy, touching closely the score of 98%, which was followed by an unchanged condition until 30 iterations. In contrast, the score of our proposed Bayesian Optimization technique commenced before 97%, which was almost steep up to 2 iterations, touching the accuracy

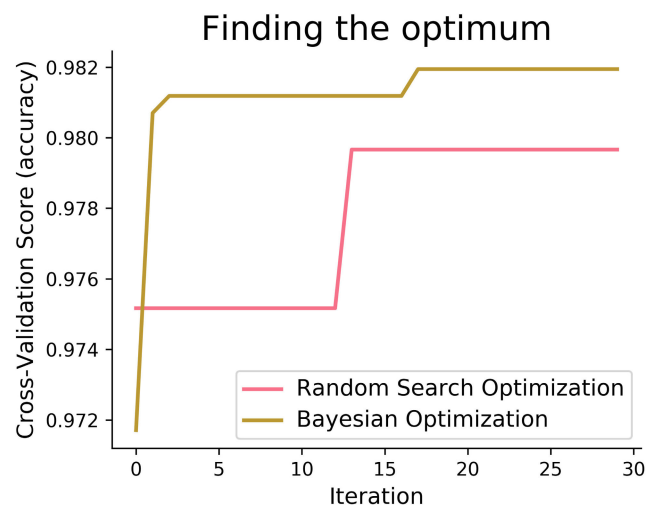
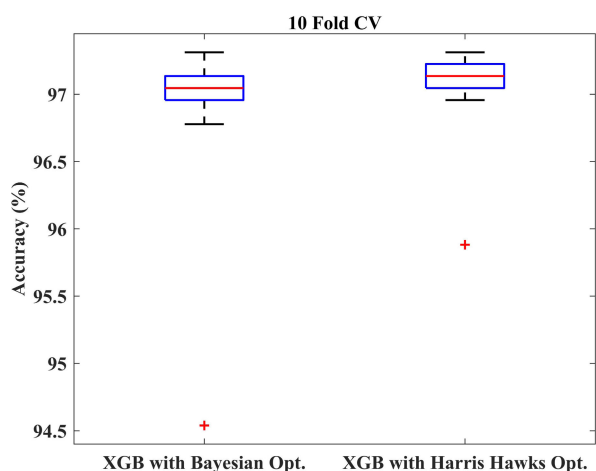


FIGURE 13. Comparative optimization techniques applied to the XGB model.

above 98%. The most exciting information should be mentioned here that the score of our proposed method remains unchanged, except for a slight change after 15 iterations. Before finishing 30 iterations, its accuracy touched the mountain point.

The proposed Bayesian Optimisation Framework has also been applied to the most recent Harris Hawks optimization algorithm calculated over 200 evolutions with 20 populations on the same train-test settings. It provides 98.39% cross-validation accuracy, whereas the testing accuracy is 98%. The result is very similar to the Bayesian Optimisation framework. However, it takes 6204.80 Sec. which is 9.4 times slower than our proposed framework as it requires more evaluations and optimization calculations; see Table 5.

To further justify, a statistical significance test between Bayesian optimization and Harris Hawks optimization algorithm is performed on 10-fold cross-validation using t-test. After that, the p-value is calculated, and the box-plot is plotted. A p-value of 0.47 is found, which suggests that there is no statistically significant difference between these two optimizations. The box-plot illustrated in Figure 14 also justifies the same statements.



**FIGURE 14.** Box-plot of Bayesian optimization and Harris Hawks optimization.

#### IV. DISCUSSION AND COMPARISON

In this research, a Bayesian optimization-based machine learning framework with a class balancing strategy using the ADASYN algorithm is proposed to identify COVID patients from their inpatient facility data. Nine state-of-the-art classifiers such as LDA, QLDA, NB, KNN, DT, RF, XGB, GB, and SVC are utilized in this proposed framework to identify COVID patients. Different classification measures such as accuracy, sensitivity, specificity, Kappa index, Matthews correlation coefficient are used to show the efficacy of different classifiers. This study also performed 10-fold cross-validation accuracy to achieve statistical significance using ANOVA, recall rate vs. decision boundary threshold analysis, ROC, and bootstrap ROC. Finally, SHAP

analysis is performed to interpret the feature importance and interpret the model. These different classification indicators describe model performance from another point of view. The primary intention to use these indicators is to describe the classification performance from a different perspective. It can be seen from Table 2 that RF yielded the highest classification performance in terms of accuracy, kappa index, and MCC, etc. However, the classification performance of XGB and GBC is very close to RF. The ANOVA and multi-comparison tests show that the average accuracy of RF, XGB, and GBC are very close and are not statistically significant. However, the 10-fold cross-validation accuracy of XGB provides the highest value (see Table 4). Moreover, the balanced XGB model offers the highest classification performance when applied to the original test data (see Table 3). Also, the recall rate vs. decision threshold boundary indicates the superior performance of XGB and SVC (see Figure 9). This concludes that the balanced and optimized XGB model would be the best choice for detecting COVID patients using their inpatient facility data. Therefore, further analyses such as bootstrap ROC and SHAP analysis and features importance analysis are done on a balanced and optimized XGB model.

Regarding the ADASYN algorithm, it should be mentioned that ADASYN adaptively generates synthetic data samples for the COVID-yes class since it is a minority class to reduce bias introduced by imbalanced data distribution. ADASYN moves the classifier's decision boundary towards harder-to-learn examples, improving the learning performance [28]. Therefore, applying the ADASYN algorithm enhances the learning process and eventually improves our COVID classification performance; see Table 2 to understand the effect of ADASYN in detail. Regarding Bayesian optimization (BO), unlike grid search and random search, it can be mentioned that BO takes the previous objective function evaluation into account, and the function goes to the optimal solution. Therefore, the hyperparameter using BO provides fine-tuning parameters, which ultimately builds an optimized model and consequently increases the classification performance. SHAP is used to determine feature importance and model interpretation; it can be mentioned that SHAP uses a game-theoretic approach, which has an excellent mathematical background and current state-of-the-art approach.

Due to the salient features mentioned above, it can be noted that the proposed framework can not only be applied to COVID-19 detection but also applied to other classification problems such as diabetic prediction, asthma prediction, etc. While describing the significance and strength of this study, it is also logical to explain the weaknesses of this study. The database used in this study is a moderately large dataset. It will be useful to apply the proposed framework on a larger dataset and validate the proposed approach on a completely independent dataset before clinical use. Clinical blood sample data and integration to X-ray and CT-scan will enhance the detection rate and validity. This is beyond the scope of this study.

TABLE 6. Description of the clinically operable decision tree algorithm.

Node No	Input: $x_1$ = cough; $x_2$ = loss of smell; $x_3$ = high risk exposure occupation; $x_4$ = sats;
Output: The decision of Covid Yes or Covid_no.	
Node 1:	if $x_2 < 0.5$ then node 2 elseif $x_2 \geq 0.5$ then node 3 else Covid_no
Node 2:	if $x_1 < 0.5$ then node 4 elseif $x_1 \geq 0.5$ then node 5 else Covid_no
Node 3:	class = Covid_Yes
Node 4:	if $x_3 < 0.5$ then node 6 elseif $x_3 \geq 0.5$ then node 7 else Covid_no
Node 5:	class = Covid_Yes
Node 6:	class = Covid_no
Node 7:	if $x_4 < 2.5$ then node 8 elseif $x_4 \geq 2.5$ then node 9 else Covid_no
Node 8:	class = Covid_Yes
Node 9:	class = Covid_no

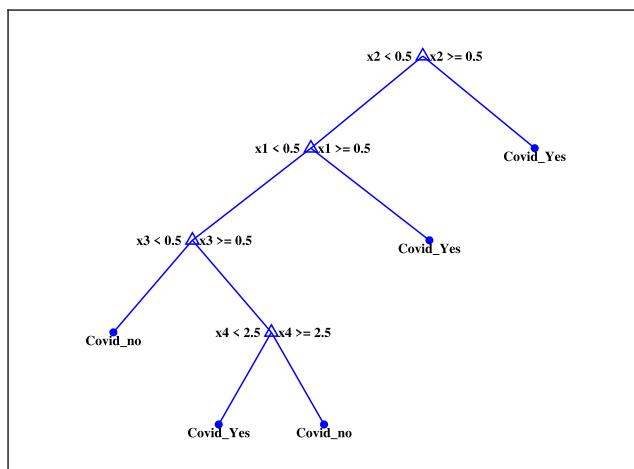


FIGURE 15. A decision rule using four key features and their thresholds in absolute value.

A. DEVELOPMENT OF A CLINICALLY OPERABLE DECISION TREE

A clinically operable decision tree would benefit clinical staff as it is straightforward to understand the underlying process. As DT are simple classifiers consisting of sequences of binary decisions organized hierarchically [56], we have built a simple tree by using four important features,  $x_1$  = cough;  $x_2$  = loss of smell;  $x_3$  = high-risk exposure occupation;  $x_4$  = sats; Note that, the continuous value of oxygen saturation feature, i.e.,  $x_4$  feature is discretized into three different levels of 1, 2 and 3 to denote severe, moderate and normal level, respectively.  $x_4$  feature value lies between 75 and 90 mm-Hg is treated as severe, 91 and 95 mm-Hg as moderate, and 96 and 100 mm-Hg as a normal level. Figure 15 represents the corresponding DT, and the description of the tree algorithm is given in Table 6.

B. DEVELOPMENT OF A DECISION SUPPORT SYSTEM (DSS)

A DSS could be beneficial to support clinical staff for screening COVID-19 patients from their inpatient facility data. A DSS is usually a graphical representation of decision,

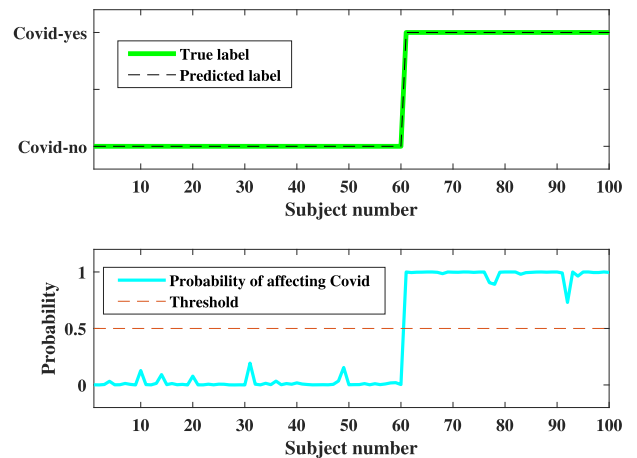


FIGURE 16. Probabilistic output for the DSS. In the upper figure, the 0 has represented a subject with COVID negative, whereas 1 represented a subject with COVID positive. The lower figure represents a probabilistic outcome of the subject affected by COVID, where the red dotted line defines the threshold level. When the patient data level exceeds this threshold level, then the subject will be considered as COVID positive. Whereas the subject with the probability of less than 0.5, i.e., the threshold value, will be regarded as COVID negative. In either way, we can say that this the chance that a person is affected by COVID.

COVID-19, in this case, to visualize the probable state of the patient. A possible outcome of COVID suspected patient’s inhouse facility data is presented in Figure 16, in terms of the posterior probability. A probabilistic result is more intuitive to the clinical staff and, therefore, used in this DSS. Note that 100 patients are used from the test database for illustration purposes. The patient is sorted in ascending order so that patients with “COVID-no” labelled appears first, and patients with “COVID-yes” appear.

C. COMPARISONS WITH OTHER METHODS/STUDIES

To delineate the superiority of our proposed research, an illustrative comparison of our work has been accomplished to other COVID studies. From the tabular illustration [Table 7], it can be mentioned that both Jim et al. [11] and Ozturk et al. [57] used CNN to obtain the accuracy respectively 92.50% and 98.08%. Furthermore, multiples research works have been carried out by [7], [63], [64] with the direct



TABLE 7. Comparison of performance with other methods.

References	Classifiers	Dataset	ACC	SE	SP	AUC
[11]	Deep Convolutional Neural Network	Clinical Image Data	92.5%	94.2%	95.6%	
[58]	RF	Clinical, Demographics	95.95%		96.95%	
[60]	SVM	Clinical, Demographics	77.5%		78.4%	98%
[57]	CNN Darknet	Clinical, Mammographic	98.08%			
[7]	XGB	Clinical, Blood samples of 75 Features	90%			
[61]	Deep learning using LSTM	Demographic	92.67%			
[62]	Logistic Regression and Multinomial NB	Clinical Data	96.2%	96%		
[59]	DT, RF	Hematochemical Values from Blood Exams	86%	95%		
[63]	XGB, RF, DT, SVM	Demographic and Symptom	85%	90%		
[64]	XGB	Clinical Data		92.5%	97.9%	>90%
Proposed	XGB	Inpatient Facility Data	98.50%	99.02%	97.98%	99.4%

implementation of XGB using mostly clinical data, where the average of the accuracy obtained from [7], [63] was less than 90%. On top of that, Wu *et al.* [58] used RF to get a classification accuracy of approximately 96%, which outperformed Brinati *et al.* [59], who utilized both DT and RF. In addition, the lowest performance was obtained by Sun *et al.* [60], who used the SVM classifier for clinical and Demographic data. Most importantly, although the accuracy of Wu *et al.* [58] is slightly higher than that of our proposed method, the AUC and Specificity of our work far outweigh the other methodologies mentioned here.

## V. CONCLUSION

This paper presents the optimal use of different machine learning techniques, including state-of-the-art classifiers, to predict COVID. The proposed approach is aimed to handle the real-time in-home dataset in detecting the COVID effectively. Thus, the proposed technique provides a user-friendly and low-cost tool for COVID detection. In designing the method, the COVID dataset, collected from CH-BH, has been used to assess the performance using different classification metrics such as accuracy, sensitivity, specificity, kappa index, etc. The hyper-parameters of different classifiers have been optimized using Bayesian optimization, and the ADASYN has been used to balance the dataset. Compared to the studies presented in this study, it is evidenced that both the classification accuracy and AUC for our proposed framework has attained the highest values of 98.50% and 99.40% using XGB, respectively. As the proposed approach has been applied to a moderately large dataset, it should be used on a big dataset before clinical trials. However, our primary intention is to test the feasibility of such settings. A similar approach can be applied to design other classification problems. Finally, two potential applications of our proposed technique, namely clinically operable decision tree and decision support system, would be beneficial for clinical staff and building an efficient recommender system. It could easily be integrated into mobile devices which would be very useful for the end-users.

## DATA AVAILABILITY

The raw dataset can be accessed through Github: <https://github.com/mdcollab/covidclinicaldata>. The processed data can be obtained from the first author (Md Abdul Awal; m.awal@ece.ku.ac.bd) of this paper.

## REFERENCES

- [1] *Mental Health and Psychosocial Considerations During the COVID-19 Outbreak*, World Health Org., Geneva, Switzerland, 2020. Accessed: Mar. 18, 2020.
- [2] *Coronavirus Disease 2019 (COVID-19): Situation Report 88*, World Health Org., Geneva, Switzerland, 2020.
- [3] N. M. T. Jebri, "World Health Organization declared a pandemic public health menace: A systematic review of the coronavirus disease 2019 'COVID-19,'" *Int. J. Psychosocial Rehabil.*, vol. 24, no. 9, pp. 2784–2795, May 2020.
- [4] WHO. *Coronavirus Disease (COVID-19) Dashboard*. Accessed: Aug. 4, 2020. [Online]. Available: <https://covid19.who.int/>
- [5] N. Van Doremalen, T. Bushmaker, D. H. Morris, M. G. Holbrook, A. Gamble, B. N. Williamson, A. Tamin, J. L. Harcourt, N. J. Thornburg, S. I. Gerber, and J. O. Lloyd-Smith, "Aerosol and surface stability of SARS-CoV-2 as compared with SARS-CoV-1," *New England J. Med.*, vol. 382, no. 16, pp. 1564–1567, 2020.
- [6] J. Farooq and M. A. Bazaz, "A novel adaptive deep learning model of Covid-19 with focus on mortality reduction strategies," *Chaos, Solitons Fractals*, vol. 138, Sep. 2020, Art. no. 110148.
- [7] X. He, S. Wang, S. Shi, X. Chu, J. Tang, X. Liu, C. Yan, J. Zhang, and G. Ding, "Benchmarking deep learning models and automated model design for COVID-19 detection with chest CT scans," *medRxiv*, pp. 1–13, Jan. 2020. [Online]. Available: <https://www.medrxiv.org/content/medrxiv/early/2020/06/09/2020.06.08.20125963.full.pdf>
- [8] A. Makris, I. Kontopoulos, and K. Tserpes, "COVID-19 detection from chest X-ray images using deep learning and convolutional neural networks," in *Proc. 11th Hellenic Conf. Artif. Intell.*, Sep. 2020, pp. 60–66.
- [9] H. X. Bai, B. Hsieh, Z. Xiong, K. Halsey, J. W. Choi, T. M. L. Tran, I. Pan, L.-B. Shi, D.-C. Wang, J. Mei, X.-L. Jiang, Q.-H. Zeng, T. K. Egglin, P.-F. Hu, S. Agarwal, F.-F. Xie, S. Li, T. Healey, M. K. Atalay, and W.-H. Liao, "Performance of radiologists in differentiating COVID-19 from Non-COVID-19 viral pneumonia at chest CT," *Radiology*, vol. 296, no. 2, pp. E46–E54, Aug. 2020.
- [10] S. Rajaraman and S. Antani, "Training deep learning algorithms with weakly labeled pneumonia chest X-ray data for COVID-19 detection," *medRxiv*, pp. 1–13, May 2020. [Online]. Available: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7239073/>
- [11] A. A. J. Jim, I. Rafi, M. S. Chowdhury, N. Sikder, M. P. Mahmud, S. Rubaie, M. Masud, A. K. Bairagi, K. Bhakta, and A.-A. Nahid, "An automatic computer-based method for fast and accurate Covid-19 diagnosis," *medRxiv*, pp. 1–11, Jan. 2020. [Online]. Available: <https://www.medrxiv.org/content/medrxiv/early/2020/07/05/2020.07.02.20136721.full.pdf>

- [12] A. Abbas, M. M. Abdelsamea, and M. Gaber, "4S-DT: Self supervised super sample decomposition for transfer learning with application to COVID-19 detection," 2020, *arXiv:2007.11450*. [Online]. Available: <http://arxiv.org/abs/2007.11450>
- [13] M. Shorfuzzaman and M. Masud, "On the detection of covid-19 from chest X-ray images using CNN-based transfer learning," *Comput., Mater. Continua*, vol. 64, no. 3, p. 1359–1381, 2020.
- [14] M. Escobar, G. Jeanneret, L. Bravo-Sánchez, A. Castillo, C. Gómez, D. Valderrama, M. F. Roa, J. Martínez, J. Madrid-Wolff, M. Cepeda, and M. Guevara-Suarez, "Smart pooling: AI-powered COVID-19 testing," *medRxiv*, pp. 1–21, Jan. 2020. [Online]. Available: <https://www.medrxiv.org/content/medrxiv/early/2020/08/05/2020.07.13.20152983.full.pdf>
- [15] A. K. Jaiswal, P. Tiwari, V. K. Rathi, J. Qian, H. M. Pandey, and V. H. C. Albuquerque, "COVIDPEN: A novel COVID-19 detection model using chest X-rays and CT scans," *medRxiv*, pp. 1–8, Jan. 2020. [Online]. Available: <https://www.medrxiv.org/content/medrxiv/early/2020/07/10/2020.07.08.20149161.full.pdf>
- [16] T. Majeed, R. Rashid, D. Ali, and A. Asaad, "Problems of deploying CNN transfer learning to detect COVID-19 from chest X-rays," *medRxiv*, pp. 1–11, Jan. 2020. [Online]. Available: <https://www.medrxiv.org/content/medrxiv/early/2020/06/07/2020.05.12.20098954.full.pdf>
- [17] A. K. Bairagi, M. Masud, D. H. Kim, M. S. Munir, A.-A. Nahid, S. F. Abedin, K. M. Alam, S. Biswas, S. S. Alshamrani, Z. Han, and C. S. Hong, "Controlling the outbreak of COVID-19: A noncooperative game perspective," *IEEE Access*, vol. 8, pp. 215570–215581, 2020.
- [18] S. N. Nan, Y. Ya, T. L. Ling, G. H. Nv, P. H. Ying, and J. Bin, "A prediction model based on machine learning for diagnosing the early COVID-19 patients," *medRxiv*, pp. 1–12, Jan. 2020. [Online]. Available: <https://www.medrxiv.org/content/medrxiv/early/2020/06/04/2020.06.03.20120881.full.pdf>
- [19] R. N. Phursule, P. N. Mahalle, P. K. Ukhalkar, and S. R. Todmal, "Machine learning-based IoT-enabled perspective model for prediction of COVID-19 test in early stage," *Mach. Learn.*, vol. 29, no. 12s, pp. 2599–2604, 2020.
- [20] S. Rajaraman, S. Sornapudi, P. O. Alderson, L. R. Folio, and S. K. Antani, "Interpreting deep ensemble learning through radiologist annotations for COVID-19 detection in chest radiographs," *medRxiv*, pp. 1–12, Jan. 2020. [Online]. Available: <https://www.medrxiv.org/content/medrxiv/early/2020/07/16/2020.07.15.20154385.full.pdf>
- [21] M. Roser, H. Ritchie, E. Ortiz-Ospina, and J. Hasell. (2020). *Coronavirus Pandemic (COVID-19)*. Our World in Data. [Online]. Available: <https://github.com/owid/covid-19-data/tree/master/public/data/>
- [22] M. Nour, Z. Cömert, and K. Polat, "A novel medical diagnosis model for COVID-19 infection detection based on deep features and Bayesian optimization," *Appl. Soft Comput.*, vol. 97, Dec. 2020, Art. no. 106580.
- [23] I. Ahmadianfar, O. Bozorg-Haddad, and X. Chu, "Gradient-based optimizer: A new Metaheuristic optimization algorithm," *Inf. Sci.*, vol. 540, pp. 131–159, Nov. 2020.
- [24] S. Li, H. Chen, M. Wang, A. A. Heidari, and S. Mirjalili, "Slime mould algorithm: A new method for stochastic optimization," *Future Gener. Comput. Syst.*, vol. 111, pp. 300–323, Oct. 2020.
- [25] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: Algorithm and applications," *Future Gener. Comput. Syst.*, vol. 97, pp. 849–872, Aug. 2019.
- [26] C. C. Health and B. Health. *Coronavirus Disease 2019 (COVID-19) Clinical Data Repository*. Accessed: Aug. 10, 2020. [Online]. Available: <https://covidclinicaldata.org/>
- [27] S. V. Buuren and K. Groothuis-Oudshoorn, "Mice: Multivariate imputation by chained equations in R," *J. Stat. Softw.*, vol. 45, no. 3, pp. 1–68, 2011.
- [28] H. He, Y. Bai, E. A. Garcia, and S. Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in *Proc. IEEE Int. Joint Conf. Neural Netw., IEEE World Congr. Comput. Intell.*, Jun. 2008, pp. 1322–1328.
- [29] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Ann. Eugenics*, vol. 7, no. 2, pp. 179–188, Sep. 1936.
- [30] T. Sapatinas, "Discriminant analysis and statistical pattern recognition," *J. Roy. Stat. Soc. A, Statist. Soc.*, vol. 168, no. 3, pp. 635–636, 2005.
- [31] M. Maniruzzaman, M. J. Rahman, M. Al-MehediHasan, H. S. Suri, M. M. Abedin, A. El-Baz, and J. S. Suri, "Accurate diabetes risk stratification using machine learning: Role of missing value and outliers," *J. Med. Syst.*, vol. 42, no. 5, p. 92, May 2018.
- [32] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Stanford, CA, USA: Springer, 2009.
- [33] T. M. Cover, "Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition," *IEEE Trans. Electron. Comput.*, vol. EC-14, no. 3, pp. 34–326, Jun. 1965.
- [34] E. Fix and J. Hodges, "Discriminatory analysis, nonparametric discrimination," United States Air Force School Aerosp. Med., Universal City, TX, USA, Tech. Rep. 4, 1951.
- [35] U. R. Acharya, F. Molinari, S. V. Sree, S. Chattopadhyay, K.-H. Ng, and J. S. Suri, "Automated diagnosis of epileptic EEG using entropies," *Biomed. Signal Process. Control*, vol. 7, no. 4, pp. 401–408, Jul. 2012.
- [36] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
- [37] G. Biau, B. Cadre, and L. Rouvière, "Accelerated gradient boosting," *Mach. Learn.*, vol. 108, no. 6, pp. 971–992, Jun. 2019.
- [38] L. Zhang and C. Zhan, "Machine learning in rock facies classification: An application of XGBoost," in *Proc. Int. Geophys. Conf.*, Qingdao, China, May 2017, pp. 17–20.
- [39] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.
- [40] N. Barakat, A. P. Bradley, and M. N. H. Barakat, "Intelligible support vector machines for diagnosis of diabetes mellitus," *IEEE Trans. Inf. Technol. Biomed.*, vol. 14, no. 4, pp. 114–120, Jul. 2010.
- [41] M. A. Rahman, S. M. Shoaib, M. A. Amin, R. N. Toma, M. A. Moni, and M. A. Awal, "A Bayesian optimization framework for the prediction of diabetes mellitus," in *Proc. 5th Int. Conf. Adv. Electr. Eng. (ICAEE)*, Sep. 2019, pp. 357–362.
- [42] E. V. Altay and B. Alatas, "Intelligent optimization algorithms for the problem of mining numerical association rules," *Phys. A, Stat. Mech. Appl.*, vol. 540, Feb. 2020, Art. no. 123142.
- [43] E. V. Altay and B. Alatas, "A novel clinical decision support system for liver fibrosis using evolutionary multi-objective method based numerical association analysis," *Med. Hypotheses*, vol. 144, Nov. 2020, Art. no. 110028.
- [44] E. V. Altay and B. Alatas, "Association analysis of parkinson disease with vocal change characteristics using multi-objective metaheuristic optimization," *Med. Hypotheses*, vol. 141, Aug. 2020, Art. no. 109722.
- [45] E. C. Garrido-Merchán and D. Hernández-Lobato, "Dealing with categorical and integer-valued variables in Bayesian optimization with Gaussian processes," *Neurocomputing*, vol. 380, pp. 20–35, Mar. 2020.
- [46] I. Roman, J. Ceberio, A. Mendiburu, and J. A. Lozano, "Bayesian optimization for parameter tuning in evolutionary algorithms," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2016, pp. 4839–4845.
- [47] M. Pelikan, D. E. Goldberg, and E. Cantú-Paz, "BOA: The Bayesian optimization algorithm," in *Proc. Genetic Evol. Comput. Conf. (GECCO)*, vol. 1, 1999, pp. 525–532.
- [48] A. M. Hay, "The derivation of global estimates from a confusion matrix," *Int. J. Remote Sens.*, vol. 9, no. 8, pp. 1395–1398, Aug. 1988.
- [49] W. Yu, T. Liu, R. Valdez, M. Gwinn, and M. J. Khoury, "Application of support vector machine modeling for prediction of common diseases: The case of diabetes and pre-diabetes," *BMC Med. Informat. Decis. Making*, vol. 10, no. 1, p. 16, Dec. 2010.
- [50] M. Frigge, D. C. Hoaglin, and B. Iglewicz, "Some implementations of the boxplot," *Amer. Statistician*, vol. 43, no. 1, pp. 50–54, Feb. 1989.
- [51] D. J. Benjamin, J. O. Berger, M. Johannesson, B. A. Nosek, E. J. Wagenmakers, E. R. Berk, K. A. Bollen, B. Brembs, L. Brown, C. Camerer, and D. Cesarini, "Redefine statistical significance," *Nature Hum. Behav.*, vol. 2, no. 1, pp. 6–10, 2018.
- [52] T. G. Dietterich, "Approximate statistical tests for comparing supervised classification learning algorithms," *Neural Comput.*, vol. 10, no. 7, pp. 1895–1923, Oct. 1998.
- [53] S. M. Lundberg and S.-I. Lee, "Consistent feature attribution for tree ensembles," 2017, *arXiv:1706.06060*. [Online]. Available: <http://arxiv.org/abs/1706.06060>
- [54] S. M. Lundberg, G. G. Erion, and S.-I. Lee, "Consistent individualized feature attribution for tree ensembles," 2018, *arXiv:1802.03888*. [Online]. Available: <http://arxiv.org/abs/1802.03888>
- [55] D. Datamann. *Explain Your Model with the SHAP Values*. Accessed: Sep. 14, 2019. [Online]. Available: <https://towardsdatascience.com/explain-your-model-with-the-shap-values-bc36aac4de3d>
- [56] L. Yan, H. T. Zhang, J. Goncalves, Y. Xiao, M. Wang, Y. Guo, C. Sun, X. Tang, L. Jing, M. Zhang, and X. Huang, "An interpretable mortality prediction model for COVID-19 patients," *Nature Mach. Intell.*, vol. 2, pp. 283–288, May 2020.

- [57] T. Ozturk, M. Talo, E. A. Yildirim, U. B. Baloglu, O. Yildirim, and U. R. Acharya, "Automated detection of COVID-19 cases using deep neural networks with X-ray images," *Comput. Biol. Med.*, vol. 121, Jun. 2020, Art. no. 103792.
- [58] J. Wu, P. Zhang, L. Zhang, W. Meng, J. Li, C. Tong, Y. Li, J. Cai, Z. Yang, J. Zhu, and M. Zhao, "Rapid and accurate identification of COVID-19 infection through machine learning based on clinical available blood test results," *medRxiv*, pp. 1–12, Jan. 2020. [Online]. Available: <https://www.medrxiv.org/content/medrxiv/early/2020/04/06/2020.04.02.20051136.full.pdf>
- [59] D. Brinati, A. Campagner, D. Ferrari, M. Locatelli, G. Banfi, and F. Cabitza, "Detection of COVID-19 infection from routine blood exams with machine learning: A feasibility study," *J. Med. Syst.*, vol. 44, no. 8, p. 135, Aug. 2020.
- [60] L. Sun, F. Song, N. Shi, F. Liu, S. Li, P. Li, W. Zhang, X. Jiang, Y. Zhang, L. Sun, X. Chen, and Y. Shi, "Combination of four clinical indicators predicts the severe/critical symptom of patients infected COVID-19," *J. Clin. Virol.*, vol. 128, Jul. 2020, Art. no. 104431.
- [61] V. K. R. Chimmula and L. Zhang, "Time series forecasting of COVID-19 transmission in Canada using LSTM networks," *Chaos, Solitons Fractals*, vol. 135, Jun. 2020, Art. no. 109864.
- [62] A. M. U. D. Khanday, S. T. Rabani, Q. R. Khan, N. Rouf, and M. M. U. Din, "Machine learning based approaches for detecting COVID-19 using clinical text data," *Int. J. Inf. Technol.*, vol. 12, no. 3, pp. 1–9, 2020.
- [63] M. M. Ahamad, S. Aktar, M. Rashed-Al-Mahfuz, S. Uddin, P. Liò, H. Xu, M. A. Summers, J. M. W. Quinn, and M. A. Moni, "A machine learning model to identify early stage symptoms of SARS-Cov-2 infected patients," *Expert Syst. Appl.*, vol. 160, Dec. 2020, Art. no. 113661.
- [64] W. T. Li, J. Ma, N. Shende, G. Castaneda, J. Chakladar, J. C. Tsai, L. Apostol, C. O. Honda, J. Xu, L. M. Wong, and T. Zhang, "Using machine learning of clinical data to diagnose COVID-19: A systematic review and meta-analysis," *BMC Med. Informat. Decis. Making*, vol. 20, no. 1, p. 247, Dec. 2020, doi: [10.1186/s12911-020-01266-z](https://doi.org/10.1186/s12911-020-01266-z).



**MD. SHAHADAT HOSSAIN** received the B.Sc. degree (Hons.) in mathematics and the M.Sc. degree in applied mathematics from the Mathematics Discipline, Khulna University, Khulna, Bangladesh, in 2014 and 2016, respectively. He is currently working as a Lecturer of Mathematics with the Department of Quantitative Sciences, International University of Business Agriculture and Technology (IUBAT), Dhaka, Bangladesh. He has five research articles published in international journals. He is currently working on several research articles related to machine learning and data science. His research interests include machine learning, deep learning, data science, optimization, image processing, signal processing, and applied mathematics.



**ABDULLAH AL-MAMUN BULBUL** received the B.Sc. and M.Sc. degrees in electronics and communication engineering (ECE) from the ECE Discipline, Khulna University, Khulna, Bangladesh, in 2013 and 2017, respectively, securing the First Place from the Department of Electronics and Communication Engineering. He is currently working as a Lecturer with the Department of Electronics and Communication Engineering (ECE), Khulna University. He has more than 32 publications in journals and conferences. Recently, he has presented five papers in IEEE conference. His current research interests include biomedical engineering using machine learning, bio-sensing, chemical sensing, optical and millimeter-wave communication, PCF model designing, IoT, and WSN.



He is currently investigating some projects as a Principal Investigator and a Co-Investigator and, supervising several undergraduate and post-graduate students. He has more than 35 papers published in internationally accredited journals and conferences. His research interests include signal processing, especially biomedical signal processing, big data analysis, image processing, time-frequency analysis, machine learning algorithms, deep learning, optimization, and computational intelligence biomedical engineering.

**MD. ABDUL AWAL** received the B.Sc. degree in electronics and communication engineering (ECE) from the ECE Discipline, Khulna University, Khulna, Bangladesh, in 2009, the M.Sc. degree in biomedical engineering from the Khulna University of Engineering and Technology, in 2011, and the Ph.D. degree in biomedical engineering from The University of Queensland, Australia, in 2018. He is currently working as an Associate Professor with the ECE Discipline, Khulna University.



distributed algorithms, data security, data interoperability, formal methods, and cloud and multimedia for healthcare. He is a member of ACM. He has served as a Technical Program Committee Member in different international conferences. He was a recipient of a number of awards, including the Research in Excellence Award from Taif University. He is on the Associate Editorial Board of IEEE ACCESS and *International Journal of Knowledge Society Research (IJKSR)*, and the Editorial Board Member of *Journal of Software*. He also served as a Guest Editor for *Computer Science and Information Systems Journal* and *Journal of Universal Computer Science (JUCS)*.

**MEHEDI MASUD** (Senior Member, IEEE) received the Ph.D. degree in computer science from the University of Ottawa, Canada. He is currently a Full Professor with the Department of Computer Science, Taif University, Taif, Saudi Arabia. He has authored and coauthored around 50 publications, including refereed IEEE/ACM/Springer/Elsevier journals, conference papers, books, and book chapters. His research interests include cloud computing, distributed algorithms, data security, data interoperability, formal methods, and cloud and multimedia for healthcare. He is a member of ACM. He has served as a Technical Program Committee Member in different international conferences. He was a recipient of a number of awards, including the Research in Excellence Award from Taif University. He is on the Associate Editorial Board of IEEE ACCESS and *International Journal of Knowledge Society Research (IJKSR)*, and the Editorial Board Member of *Journal of Software*. He also served as a Guest Editor for *Computer Science and Information Systems Journal* and *Journal of Universal Computer Science (JUCS)*.



and Ph.D. studies. Since 2013, he has been working as a Faculty Member with the Department of Software Engineering, Daffodil International University, Bangladesh. He has been published several conference and journal papers. His research interests include machine learning, deep learning, bioinformatics, drug discovery, and pattern recognition. He received the Best Paper Award from the IEEE conference of the ICCSNT 2016.

**S. M. HASAN MAHMUD** received the B.Sc. degree in software engineering from the Shenyang University of Chemical Technology, China, in 2011, and the M.Sc. degree in software engineering from Hohai University, China, in 2016. He is currently pursuing the Ph.D. degree in computer science and technology with the University of Electronic Science and Technology of China, China. He received full scholarships from the China Scholarship Council (CSC) for his Master and Ph.D. studies. Since 2013, he has been working as a Faculty Member with the Department of Software Engineering, Daffodil International University, Bangladesh. He has been published several conference and journal papers. His research interests include machine learning, deep learning, bioinformatics, drug discovery, and pattern recognition. He received the Best Paper Award from the IEEE conference of the ICCSNT 2016.



His research interests include wireless resource management in 5G and beyond, Healthcare, IIoT, cooperative communication, and game theory. He has served as a Technical Program Committee Member for different international conferences.

**ANUPAM KUMAR BAIRAGI** (Member, IEEE) received the B.Sc. and M.Sc. degrees in computer science and engineering from Khulna University (KU), Bangladesh, and the Ph.D. degree in computer engineering from Kyung Hee University, South Korea. He is currently an Associate Professor with the Computer Science and Engineering Discipline, KU. He has authored and coauthored around 40 publications, including refereed IEEE/ACM journals and conference papers. His research interests include wireless resource management in 5G and beyond, Healthcare, IIoT, cooperative communication, and game theory. He has served as a Technical Program Committee Member for different international conferences.