International Journal of Innovative Computing, Information and Control
Volume **18**, Number **1**, February **2022**

ICIC International ©2022 ISSN 1349-4198
pp. **57–72**

AN INTELLIGENT COMPUTING FOR DIAGNOSING COVID-19 USING AVAILABLE BLOOD TESTS

Ahmed Yacoub Yousif[1], Sanar Mazin Younis[2], Samer Alaa Hussein[1] and Nadia Mohammed Ghanim Al-Saidi[3]

[1]Information Technology Center
[3]Department of Applied Sciences
University of Technology
Al Sina'a Street, Baghdad 10066, Iraq
{ ahmed.y.yousif; Samer.A.Hussein; nadia.m.ghanim }@uotechnology.edu.iq

[2]General Directorate of Education Al-Rusafa 2
Ministry of Education
Palestine Street, Baghdad 10064, Iraq
sanarrazoky@gmail.com

Received July 2021; revised November 2021

Abstract. *Since the end of 2019, the Coronavirus (known as COVID-19) has spread rapidly and caused considerable losses in terms of human life and the economy. There are many difficulties with diagnosing COVID-19, including leaks in the material and equipment used in laboratories, issues with the media used to transport the virus, and the worldwide shortage in supplies. These difficulties primarily affect countries with low standards of living. Hence, scientists have been motivated to use new, low-cost, highly efficient technology for such diagnoses. In this study, artificial intelligence (AI) algorithms, precisely the logistic regression (LR), support vector machine (SVM), and extreme gradient boosting machine (XGBoost) algorithms, were used as classifiers and achieved comprehensive performances. As a result, a new model for COVID-19 diagnoses based on standard blood tests, which are cheap and available, was designed. We found that many of these classical blood tests are significantly correlated with a COVID-19 diagnosis by implementing the proposed model. The results show that the best classification accuracy obtained was 0.87, associated with an F1-Score of 0.91. This overall accuracy is considered good despite the limited number of blood test samples. Hence, machine learning algorithms can be used in conjunction with blood tests in countries with insufficient resources to combat this pandemic.*

Keywords: Logistic regression, Support vector machine, XGBoost, COVID-19, Blood test

1. **Introduction.** After appearing and spreading worldwide, COVID-19 was declared a pandemic by the World Health Organization (WHO) [1]. It can cause the failure of many organs in the human body, and symptoms differ from one patient to another. However, diagnosing this disease is a challenge for researchers and health systems. Recently, artificial intelligence (AI) utilization in this endeavor has received significant attention [2].

AI is a field of computer science [3]. Intelligent behavior requires learning. Therefore, within this field, machine learning (ML) [4,5], which is based on mathematical and statistical methods, enables computers (the machines) to learn functions using algorithms. ML can be used to synthesize complex information from datasets in a reasonable time. Currently, this technology is used intensively for medical and biological applications, especially when it comes to improving diagnoses of diseases. It provides powerful tools for

DOI: 10.24507/ijicic.18.01.57

analyzing data, particularly medical data. Many extensive studies have investigated diagnosing diseases via ML [6-13].

ML methods can be adopted for the early diagnosis of COVID-19. Several efforts to combat it and limit its spread via early diagnosis have been based on these methods. Such measures include CT scan images [14], chest X-Ray images [15], and PCR tests. Unfortunately, these diagnostic tools are costly and must utilize queues, especially when there is a shortage of testing kits. However, countries with low incomes need to look for inexpensive tools, such as blood tests, which are cheap, quick, and possible to conduct in small laboratories. Some examples of such studies can be found in [16-20]. They used the same tools but with different dataset features and different machine learning algorithms.

In [16], the authors collected 5644 samples with 559 infected cases in Brazil. They utilized the extracted blood test features and applied the ensemble learning to enhancing the performance with an overall accuracy of 99.38%. In [17], the authors used a routine blood test of seven features for 207 patients with COVID-19 symptoms. Statistical methods were used to analyze the dataset, and experimental thresholds of two from blood test features allowed distinguish 70% of COVID-19 or negative patients based on routine blood test results. In [18], different machine learning algorithms are implemented based on routine blood test samples with 24 features. They highly outperform classification with 95.159% accuracy. The authors based on the same dataset samples in [16], which are available online. In [19], the SVM algorithm is applied to detecting the COVID-19. The authors used 32 features extracted from routine blood test samples. The SVM algorithm learned with 28 features only and with accuracy reached up to 81.4%. In [20], an early prediction system of COVID-19 patients was proposed. They used 287 dataset samples with twenty different features collected from the King Fahad University Hospital, Saudi. Three different machine learning algorithms (random forest (RF), logistic regression (LR), and extreme gradient boosting machine (XGBoost)) were used for virus prediction, and their results showed that RF performs better with an accuracy of 95% than the other two classifiers.

The objective of this study was to find a diagnostic method for the COVID-19 virus in Iraq through machine learning algorithms (LR, SVM, XGBoost) based on blood tests features of Iraqi patients trying to enhance the classification accuracy by selecting the appropriate one for the early prediction of COVID-19. In addition, the impact and effectiveness of these low-cost types of diagnosing are investigated. This type of study is considered helpful due to the lack of the material and equipment used in laboratories and the high cost of PCR equipment. Three hundred blood test samples were gathered from people with typical coronavirus symptoms. A new diagnosis model was then designed using machine and ensemble learning-based methods to distinguish between the infected and non-infected individuals. The classification system for COVID-19 was implemented using the Python 3.8 programming language, which has many mathematical libraries. The machine has the specification of Intel® Core$^{\mathrm{TM}}$ i5-3317U CPU @1.7 GHz and 8 GB RAM working on Windows 8.1 Pro. Operating system.

The rest of this paper is organized as follows. Section 2 presents some necessary mathematical background. Then Section 3 describes the dataset, along with its pre-processing and analysis. Section 4 presents the results and discussion. Finally, Section 5 concludes.

2. Mathematical Background.

Various optimization methods designed to minimize cost and computation time are used as classifiers in ML. Their binary outputs make them ideal for distinguishing between infected and non-infected cases. The mathematics behind these methods follows.

2.1. **Logistic regression (LR).** Logistic regression (LR) is used to estimate probabilities of occurrence using a logistic formula. Logistic regression outputs values of a binary dependent variable. Thus, it is widely used in classification methods [21,22]. This single binary variable is used to decide whether a sample belongs to a class or not and follows the Bernoulli probability density function. This probability, which varies over the observations as an inverse logistic function of a vector, includes a constant and k explanatory variables. Its workflow is illustrated in Figure 1.

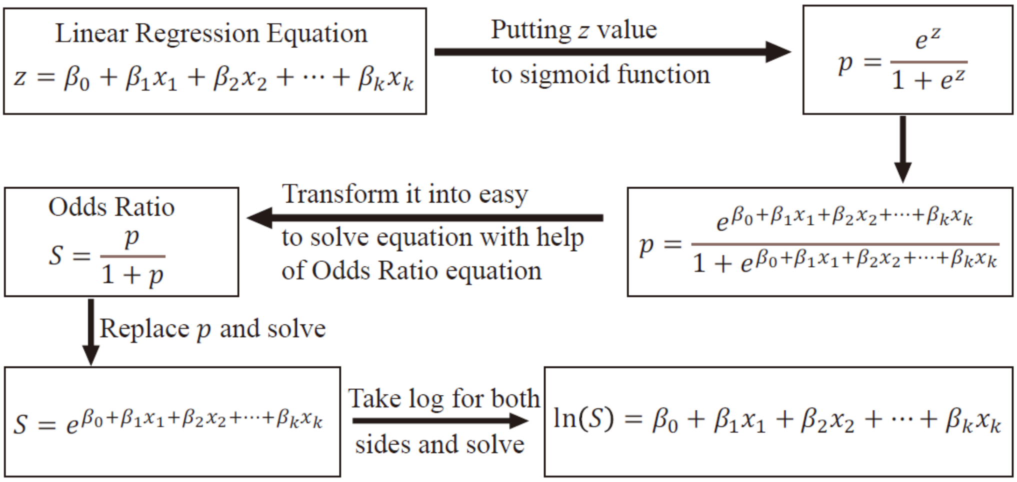

Linear Regression Equation
$z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k$
Putting z value to sigmoid function
$p = \frac{e^z}{1+e^z}$
$p = \frac{e^{\beta_0+\beta_1 x_1+\beta_2 x_2+\cdots+\beta_k x_k}}{1+e^{\beta_0+\beta_1 x_1+\beta_2 x_2+\cdots+\beta_k x_k}}$
Transform it into easy to solve equation with help of Odds Ratio equation
Odds Ratio
$S = \frac{p}{1+p}$
Replace p and solve
$S = e^{\beta_0+\beta_1 x_1+\beta_2 x_2+\cdots+\beta_k x_k}$
Take log for both sides and solve
$\ln(S) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k$

Figure 1. Logistic regression workflow

The probability p of belonging is designed by

$$p = \frac{e^{\beta_0+\beta_1 x}}{1+e^{\beta_0+\beta_1 x}} \tag{1}$$

The variable x is the input variable for the binary response variable $y = 1$, and β is the coefficient of x.

In this work, the multiple input variables $(x_1, x_2, \ldots, x_k)$, can be accommodated as superior for the response function given in Equation (1) as follows:

$$p = \frac{e^{\beta_0+\beta_1 x_1+\beta_2 x_2+\cdots+\beta_k x_k}}{1+e^{\beta_0+\beta_1 x_1+\beta_2 x_2+\cdots+\beta_k x_k}} \tag{2}$$

The parameters β_i, $i = 1, \ldots, k$ are estimated from the given data when the response variable is $y = 1$, which is performed by logistic regression. For each patient, the value of parameters β_i, $i = 1, \ldots, k$ is estimated in Algorithm 1. This nonlinear function is transformed to linearity, which is the algebraically equivalent way of representation, such that

$$S = \frac{p}{1+p} = e^{\beta_0+\beta_1 x_1+\beta_2 x_2+\cdots+\beta_k x_k} \tag{3}$$

This equation is called odds of belonging, and the linearity is obtained by taking the log function to both sides. It is called logit transformation, as shown in Equation (4).

$$\ln(S) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k \tag{4}$$

Here, S represents the sigmoid function.

2.2. **Support vector machine (SVM).** SVM is used for binary and multinomial classifications. It was introduced by Cristianini and Ricci [23]. The SVM objective is to find the optimal hyperplane that separates the dataset into two classes according to the features. Its workflow is illustrated in Figure 2.

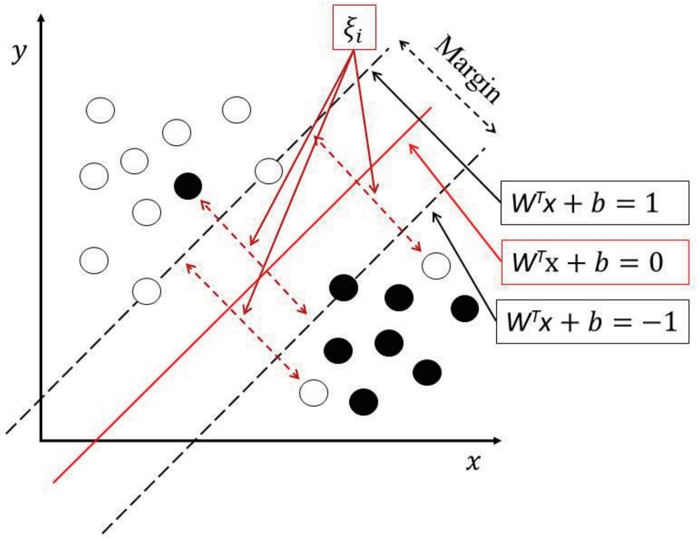


Figure 2. Support vector machine workflow

The closest data is recognized as a feature vector $\{x_i\colon x_i \in R^d\}$ from the classes $\{y_i\colon y_i \in (-1, 1)\}$ of a training compound i in a given dataset, such that

$$(x_1, y_1), \ldots, (x_n, y_n), \quad x_i \in R^d$$

These labeled points are called support vectors. The linear classifier is given by the decision function $f(x) = W^Tx + b$ which is formulated by solving the optimization problem over w_i, such that

$$\min \ \frac{1}{p}W^TW + C\sum_{i=1}^{p}\xi_i \tag{5}$$
$$\text{Subject to } t^{(i)}\left(W^Tx + b\right) \geq 1 - \xi^{(i)}$$

W^TW is the Manhattan norm that should be minimized to increase the margins, C is the trade-off between two conflicting objectives, and ξ is a cost function used as a slack variable to measure the allowance of the instance i to violate the margin.

$W^T = (w_1, w_2, \ldots, w_n)$, n is the number of features. The decision function is a linear combination of the training data and the weight function, such that

$$f(x) = \sum_{j=1}^{n} w_jx_j + b \tag{6}$$

The predicted output $\hat{y}$ is calculated by Equation (7).

$$\hat{y} = \begin{cases} 0 & \text{if } W^Tx + b < 0 \\ 1 & \text{if } W^Tx + b \geq 0 \end{cases} \tag{7}$$

2.3. **Extreme gradient boosting machine (XGBoost).** Extreme gradient boosting machine (XGBoost) is a form of tree boosting proposed by Chen and Guestrin [24] in 2016. It is a type of supervised training process that aims to find the best parameters for fitting the training data. The fitting is achieved by designing an objective function. In this work, multiple features training data $(x_1, x_2, \ldots, x_k)$ is used to predict the value of the predictor y_i. The workflow of this method is illustrated in Figure 3.

The essential mathematical elements to perform this task are given in the following equations. Finally, the prediction value is calculated by the linear combination of the

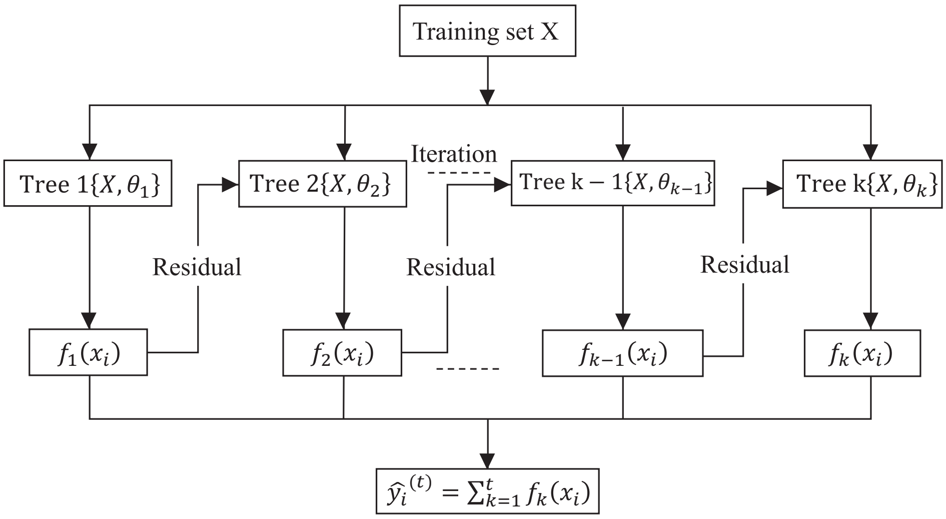


Figure 3. The XGBoost workflow

weighted input features:

$$\hat{y}_i = \sum_j \theta * x_{ij} \tag{8}$$

The objective function used for the training process is defined by

$$Obj(\theta) = L(\theta) + \Omega(\theta) \tag{9}$$

In Equation (9), θ represents the trained parameter, $L(\theta)$ represents the trained loss function (Residual), which is used to measure the fitting of the model on training data and the perversion between the prediction of the model and the actual y_i value. $\Omega(\theta)$ is the regularization term, which is added to control the complexity of the model to avoid overfitting. To optimize the loss function, we need to use the residual to correct the predictor. It was commonly defined by

$$L(\theta) = \sum_i (y_i - \hat{y}_i)^2 \tag{10}$$

If the prediction variable is denoted by $f(x_i)$ then the output $\hat{y}_i$ is averaged by collecting f of k trees in the set of all regression trees, as shown in Equation (11).

$$\hat{y}_i^{(t)} = \sum_{k=1}^{t} f_k(x_i), \quad f_k \in F \tag{11}$$

The optimal tree is determined by minimizing the objective function using the XGBoost. The tree is formulated by calculating the loss function $L(\theta)$ (the residuals). For each node, the similarity score can be found by the following formula:

$$Similarity\ score\ (SR) = \frac{\sum (residual_i)^2}{|residuals| + \lambda} \tag{12}$$

where λ is the regularization parameter.

The gain function is used to calculate the ability of the classification split over the root node, such that

$$Gain = Similarity_{left} + Similarity_{right} - Similarity_{root} \tag{13}$$

By comparing the gain value with the complexity parameter γ, tree pruning prevents overfitting. A branch containing the terminal node is prune when the gain is less than γ for an arbitrary choice of γ.

Therefore,

$$Gain - \gamma = \begin{cases} +ve & \text{not prune} \\ -ve & \text{prune} \end{cases} \tag{14}$$

After finishing the training on the residuals, the output is found by applying Equation (15) such that

$$Output = \frac{\sum residuals}{|residuals| + \lambda} \tag{15}$$

3. **Proposed Diagnosis Model.** The proposed model consists of four stages: collecting the dataset, the pre-processing of the data, feature extraction for the dataset using the LR, SVM, and XGBoost models as classifiers, and, finally, using different metrics to evaluate the results. The proposed diagnosis system based on commonly used blood tests is illustrated in Figure 4.

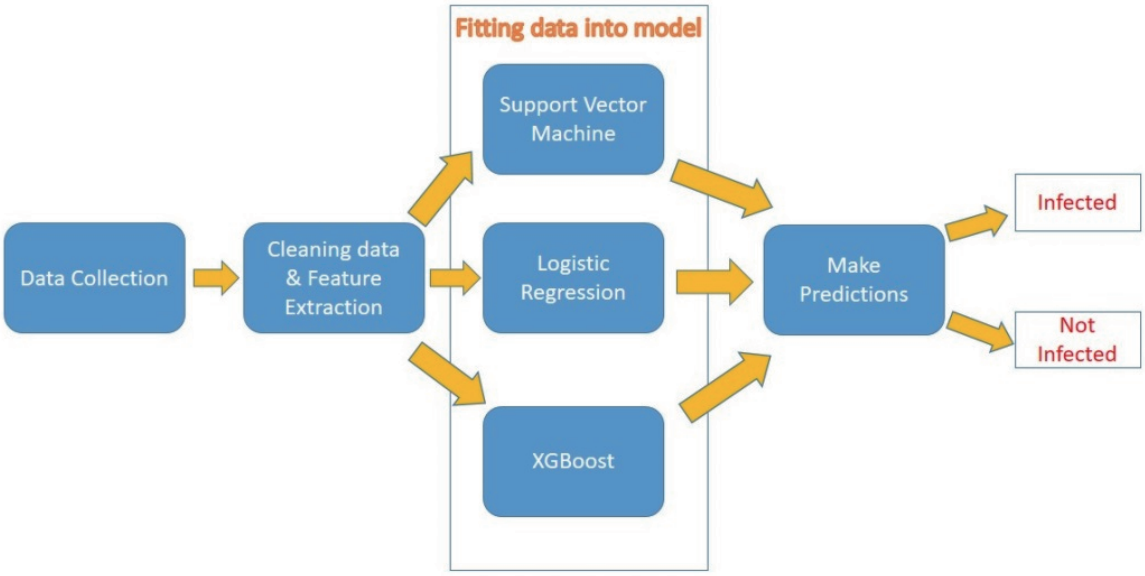


FIGURE 4. Steps of the proposed model

3.1. **Dataset.** The dataset in this study consists of 300 samples collected from many private laboratories in Iraq/Baghdad (213 not infected, 87 infected individuals). The dataset contains many attributes, but we used the results of low-cost and available blood tests. A sample from the collected dataset is shown in Table 1.

The relationship between each blood test and COVID-19 diagnosis is explained in Table 2, and the correlation coefficient of each attribute is presented in Figure 5.

TABLE 1. Collected dataset samples

O_2 content	Ferritin ng/ml	CRP	CRP Titer mg/l	WBC $10^9/L$	LYM $10^9/L$	GRA $10^9/L$	RBC $10^{12}/L$	Infected/ non-infected
0.85	87	0	0	7.8	1.81	5.26	5.89	1
0.9	97	1	5	3.83	1.85	1.75	6.22	1
0.93	87	0	0	6.57	1.86	4.27	4.47	1
0.95	10	0	0	4.94	1.13	3.75	4.27	0
0.95	79	0	0	5.85	1.68	4	5.89	0
0.92	38	0	0	13.1	3.08	9.44	5.6	0

Table 2. The relationship between the used blood test and COVID-19

Blood test	Description
O_2 content	Reflect the amount of oxygen gas, which is dissolved in the blood. - Normal range 85-100 mm HG - In COVID-19 infection, due to lung infection target $O_2 > 94$ mm HG, $O_2 < 94$ mm HG need O_2, $O_2 < 85$ mm HG considers severe COVID-19 and needs face mask or intubation.
Ferritin	Serum ferritin is considered a marker of SARS-Cov infection. Therefore, it is a helpful and straightforward laboratory test to identify and monitor the inflammatory process in COVID-19 patients. - Normal range in female 10-200 ng/ml, male 30-300 ng/ml - Value > 500 ng/ml is considered severe COVID-19 infection.
CRP	C reactive protein is a considerable marker of bacterial infection. Positive > 8.0, Negative < 8.0 At the early stage of COVID-19, CRP levels were positively correlated with lung lesions.
CRP Titer	Used as a key indicator for disease monitoring. - Normal range < 8.0/mg/l - Mild and moderate infection > 8.0/mg/l - Severe COVID-19 infection > 1.0/mg/l.
WBC	White blood cells leukocytes normal value 4.500-11.00/micro L. In COVID-19 infection case WBC < 4000.
LYM	Lymphocyte count is a prognostic marker in COVID-19. In severe infection, LYM count < 800/micro L. Normal range for age >= 21 years 1800-7700/micro L.
GRA	Granulocytes are highly abundant; phagocytic WBC could help predict patient outcomes in COVID-19.
RBC	Red blood cells are impaired in COVID-19 patients; this could increase the risk for thromboembolic events and affect microvascular blood flow.

The independent variable with the values 0 for non-infected and 1 for infected was used as a classifier for diagnosis, as shown in Figure 6. The other variables are used as features for classification, such as the p-value given in Table 3. The p-value is used to judge whether the outcome is significant or not. The value of .05 or less is substantial for the tests conducted in this paper; otherwise, the result will be neglected because it is not essential.

The descriptive statistics for the dataset including the min., max., mean, and standard deviation for each feature of the given dataset is shown in Table 4.

The relationships within the data are visualized with a pair plot in Figure 7. Note that they are both continuous and categorical variables. In this figure, the variations in each variable could be observed. It is set up in matrix format, where the row refers to the x-axis, the column refers to the y-axis, and the main diagonal refers to the distributions of the features.

3.2. **Dataset pre-processing.** The data is passed through some pre-processing steps: it is separated into train, validation, and test splits to prevent overfitting. In our model, 70% of the data is used for training, while 30% is used for testing. This essential stage leads to a precise assessment of the model.

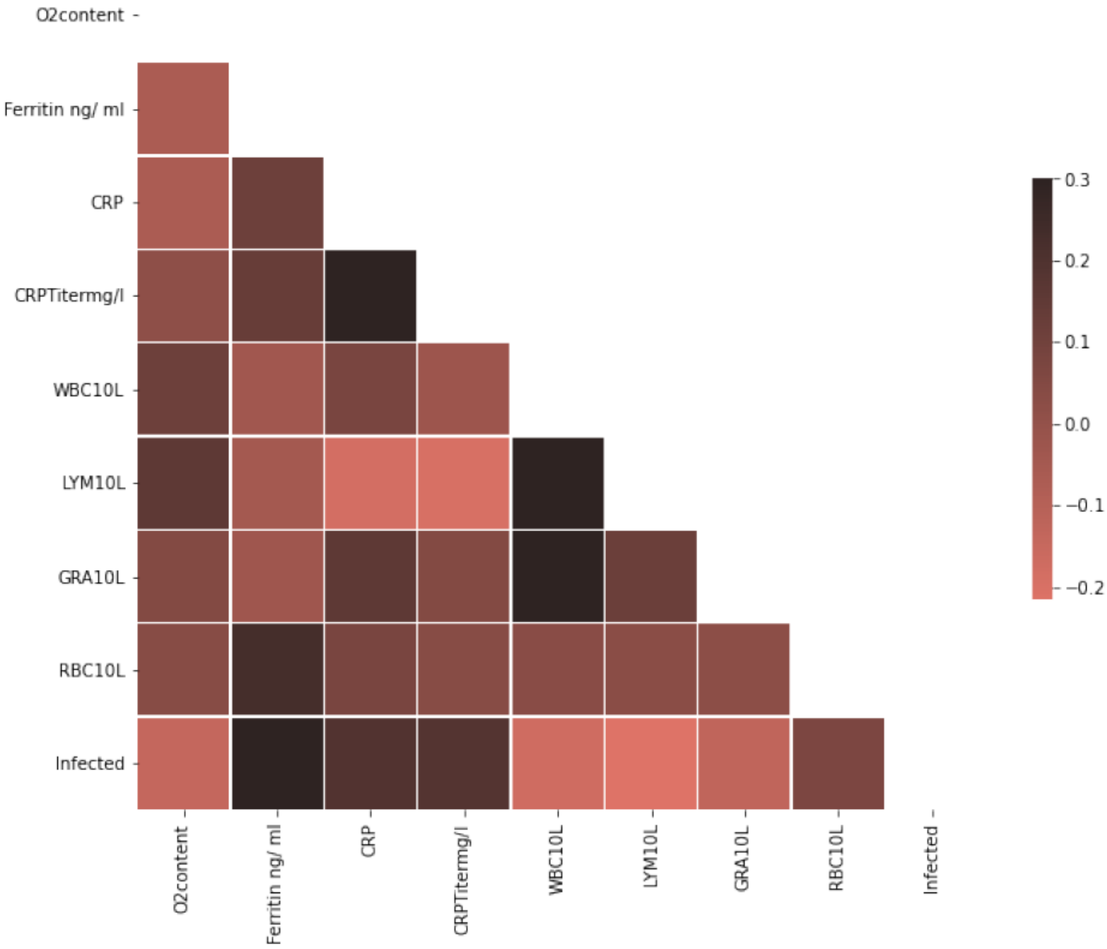


FIGURE 5. (color online) The correlations between all attributes and the diagnosis of COVID-19 (the units in (WBC, LYM, GRA) 10 L = 10^9/L, and in RBC 10 L = 10^{12}/L)

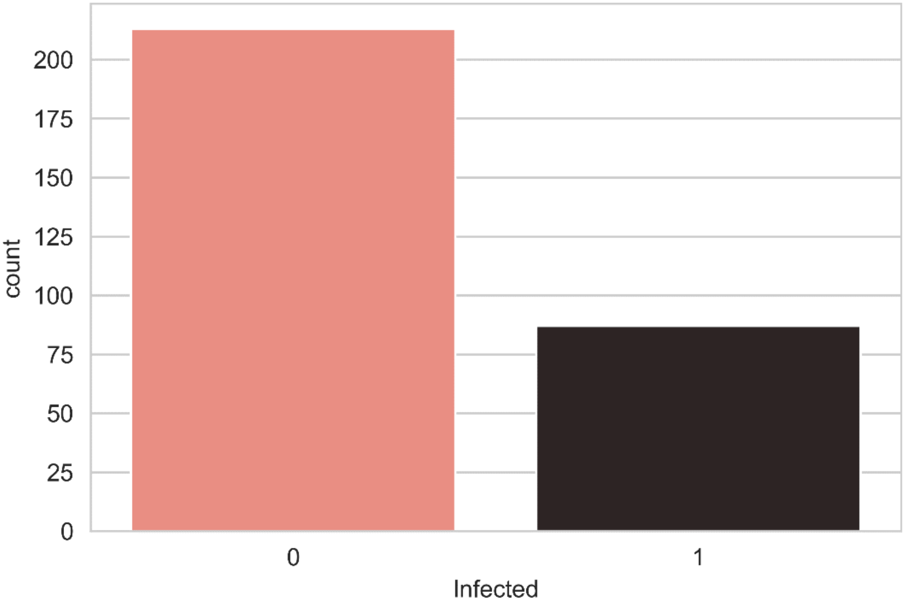


FIGURE 6. Infected and non-infected samples in the dataset

TABLE 3. The p-values of the eight features (the blood tests)

Test	p-value	Description
O_2 content	0.007	The concentration of oxygen in arterial blood (CaO_2) per L of blood.
Ferritin ng/ml	0.000	Measures of iron in the blood are given in nanograms per milliliter (ng/ml).
CRP	0.059	C-reactive protein level.
CRP Titer mg/l	0.043	The C-reactive protein amount in the blood.
WBC 10^9/L	0.05	White Blood Cell Count.
LYM 10^9/L	0.046	**Lymphocytosis** is an increase in the number or proportion of **lymphocytes** in the **blood**.
GRA 10^9/L	0.034	Granulocytes, one of the results, which WBC test is broken down into, to rule out a diagnosis.
RBC 10^{12}/L	0.559	No. red blood cells (erythrocytes) per cubic millimeter (mm^3) of blood.

TABLE 4. Features (blood tests) statistical analysis

Non-infected or infected		N	Minimum	Maximum	Mean	Std. deviation
Non-infected	O_2 content	213	85.00%	98.00%	92.6432%	2.43295%
	Ferritin ng/ml	213	6.0	110.0	67.423	24.4751
	CRP	213	0	92	2.52	10.445
	CRP Titer mg/l	213	0	92	2.52	10.445
	WBC	213	3.06	13.10	7.4492	1.82892
	LYM	213	.84	6.63	2.3830	.73833
	GRA	213	1.34	10.00	4.7169	1.58643
	RBC	213	3.65	7.28	5.0112	.59873
	Valid N (listwise)	213				
Infected	O_2 content	87	85.00%	97.00%	91.9080%	2.37052%
	Ferritin ng/ml	87	6.4	475.0	111.095	95.2935
	CRP	87	0	120	9.00	23.411
	CRP Titer mg/l	87	0	120	9.00	23.411
	WBC	87	2.56	14.01	6.6616	2.52196
	LYM	87	.14	5.45	2.0040	.87668
	GRA	87	1.26	11.54	4.2138	2.16655
	RBC	87	2.98	8.25	5.1137	.78711
	Valid N (listwise)	87				

3.3. **Classification model.** In this paper, three classification algorithms are utilized, which are LR, SVM and XGBoost.

3.3.1. *Logistic regression (LR) [21,22].* LR is used to distinguish between infected and non-infected cases. Figure 1 illustrates the workflow of the logistic regression method.

The logistic regression is used to categorize the output variables as binary values 0 or 1 (infected or not) as a relationship between the predictor variables. Finally, Algorithm 1 presents the steps of LR on the blood test dataset.

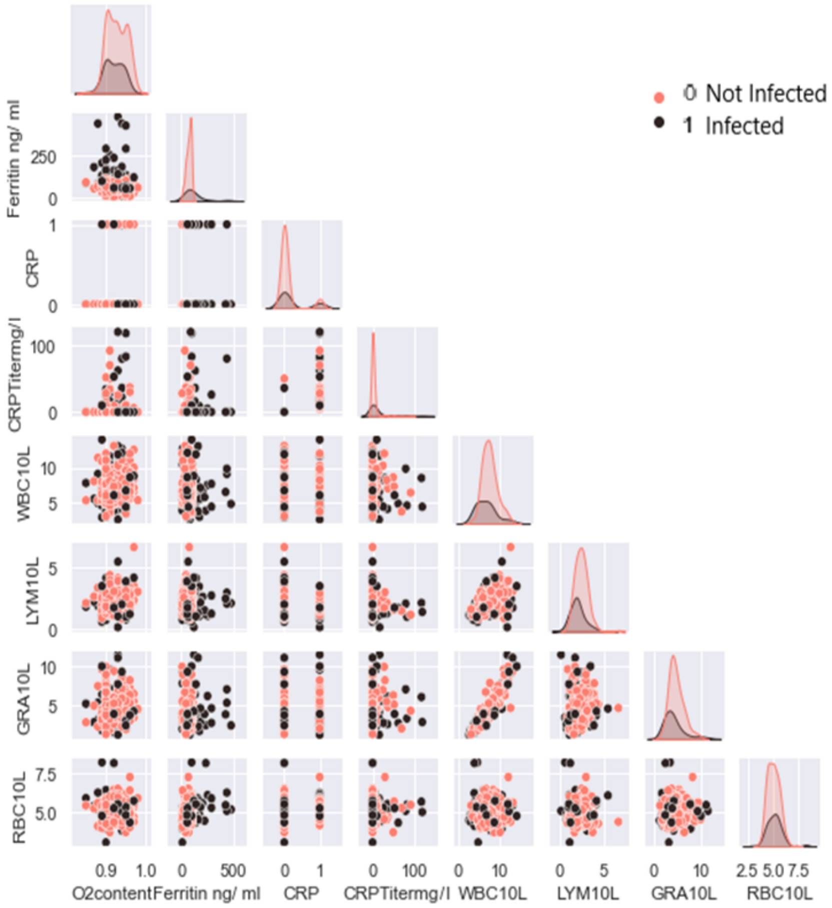


Figure 7. Pairwise relationships in the dataset (the units in (WBC, LYM, GRA) 10 L = 10^9/L, and in RBC 10 L = 10^{12}/L)

```
Algorithm 1: The logistic regression algorithm
Input: The CSV file of the blood tests instances (features). //70% for training set, 30% for testing set
Output: The predicted value (infected, uninfected) applied on the testing set
    1. Repeat for each instance in the training dataset $k = 1$ to 300.
    2. Find the best $\beta_i$ values when the class $y = 1$, and as follows:
        a. Set $\beta_i = 0$, for $i = 1, \ldots, 8$, for the instance $k_1 = (x_1, x_2, \ldots, x_8)$
        b. Apply Equation (2) to calculating the $p$ (prediction).
        c. Calculate the values of the new parameters using the update parameters equation given in (3), such that, new $\beta_i = \beta_i + \alpha(y - p) * p * (1 - p) * x_i$, where $\alpha$ is the learning rate in the range [0.1-0.3].
        d. Repeat until the error approaches zero.
    3. Apply Equation (3) to finding the $S$ value.
    4. The response variable of the patient $k$ is calculated from the logit transformation given in Equation (4), which represents the prediction output for $k$, if it is infected or not.
End
```

3.3.2. *The support vector machine [23].* SVM is also used to distinguish between infected and non-infected cases by creating a line or a hyperplane to separate the given data into two classes. Figure 2 illustrates its workflow.

The following algorithm (Algorithm 2) aims to find w_i and b that helps in separating the data and optimizing the margins.

```
Algorithm 2: The SVM pseudo code
Input: The CSV file of the blood tests instances (features), a set of (input, output) training pair samples. The input sample features $(x_1, x_2, \ldots, x_8)$, $y$ represents the output.
Output: Set of weights $w_i$ for each $x_i$. The predicted value (infected, uninfected) is a linear combination of $w_i$ calculated by Equation (7).
For each instance in the blood test dataset:
    1- Find the hyperplane that is linearly separating the data into two classes using $W^T x + b = 0$.
    2- Solve the optimization problem given in Equation (5).
    3- Apply Equation (6) to finding the decision function for $n = 8$.
    4- Find the output from Equation (7).
End
```

3.3.3. *Extreme gradient boosting machine [24].* The XGBoost is the combination of decision trees used to give the best predictor results. Its advantages are preventing the occurrence of overfitting, managing missing values efficiently, and providing an efficient computational time because their works are based on parallelism. In addition, it can gather a robust classifier from a set of weak classifiers. Algorithm 3 presents the steps of XGBoost for the blood test dataset.

3.4. **Evaluation metrics.** The following metrics are used for the evaluation of the diagnostic model

$$\text{Accuracy} = \frac{TRUE\ POSITIVE(TP) + TRUE\ NEGATIVE(TN)}{TRUE\ POSITIVE(TP) + FALSE\ POSITIVE(FP) + FALSE\ NEGATIVE(FN) + TRUE\ NEGATIVE(TN)}$$

$$\text{Sensitivity} = \frac{TRUE\ POSITIVE(TP)}{TRUE\ POSITIVE(TP) + FALSE\ POSITIVE(FP)}$$

$$\text{Specificity} = \frac{TRUE\ NEGATIVE(TN)}{FALSE\ POSITIVE(FP) + TRUE\ NEGATIVE(TN)}$$

$$\text{F1-Score} = \frac{2 * TRUE\ POSITIVE(TP)}{2 * TRUE\ POSITIVE(TP) + FALSE\ POSITIVE(FP) + FALSE\ NEGATIVE(FN)}$$

The results of these evaluators are presented in Section 4.

4. Results and Discussion.

4.1. **Results.** Eight blood test results are used as features and the blood test results for oxygen content. Most of these results are significant (with p-values less than 0.05). Table 3 lists the p-values of the eight features. After training the diagnostic models with 70% of these test results, the confusion matrices were used to test the remaining data.

Table 5 explains the results of the classification metrics. Again, many performance measurements have been applied to assessing each classifier's performance: accuracy, sensitivity, specificity, F1-Score, and AUC. Again, note that the XGBoost classifier's metrics are the best, with its highest performance measurement values reaching 87%.

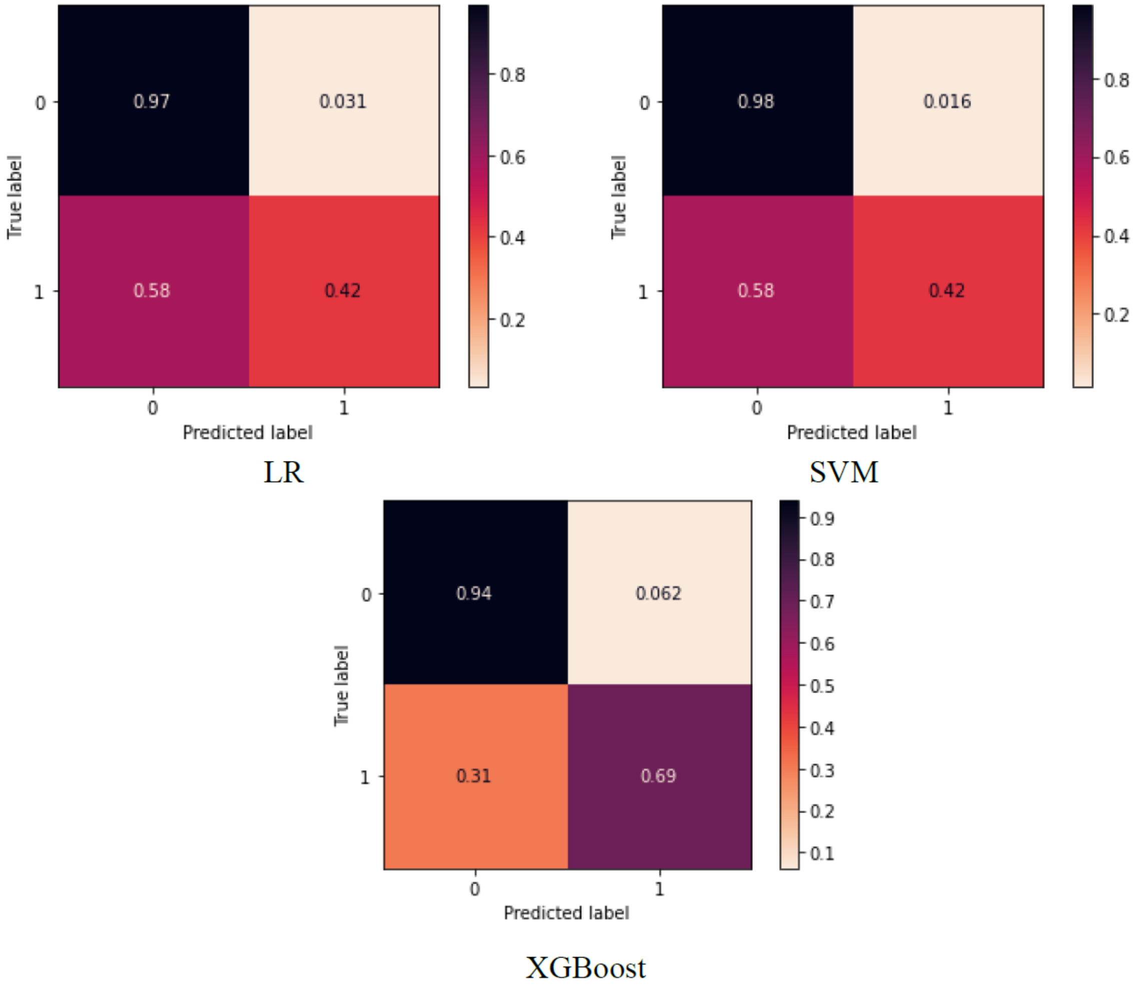
Algorithm 3: XGBoost pseudo code

```
Input: The CSV file of the blood tests instances (features) $\{x_1, x_2, \ldots, x_8\}$
Divided into a training set (70%) and testing set (30%)
Output: The predicted value $y$ (infected, uninfected)
Required parameters
DT = 50, $\lambda = 1$, Sl = 2, IP = 0.5, $eta = 0.3$ {DT represents the number of decision
tree to be created, $\lambda$ represents regularization factor, Sl represents the depth of each
tree, IP represents the initial probability (prediction or the residual), $eta$ represents
the default learning rate, $y$ is the actual prediction value}
For $j = 1$ to DT
{
  //Use the input training set values $(x_1, x_2, \ldots, x_8)$ to fit the tree[$j$] model.
For $i = 1$ to Sl
    {
        • Find the Residual IP = $y$ − IP   //for each instance in the training set.
        • Set the calculated IP as the root of tree$_i$
        • Calculate the similarity score SR for each IP from Equation (12)
        • Specify splitting criteria (threshold) for each feature $x_i$ to split the tree into
          two branches according to the threshold value.
        • Find the similarity score of the values in each branch, left and right ($S_L$, $S_R$).
        • Calculate the $Gain = S_L + S_R - S_{Root}$ Equation (13)
        • Find Pruning Tree $PN = Gain - \lambda > 0$ not prune; otherwise, prune Equation
          (14).
    }
    • Calculate the output $y$ for DT$j$ from Equation (15)
    • For the tree $j$ Make new probability by
                → Start with IP //start with the same intial IP
                → Convert probability to log(odds) value IP/1 − IP
                → Take log for both sides log(IP/1 − IP) = log(odds)
                → output = log(odds) = IP + (output ∗ $eta$)
    • Convert log(odds) to a probability by applying logistic function IP = $e^{\text{output}}/(1$
      + $e^{\text{output}})$ //Calculate the new prediction IP by logistic function and use it as the
      new probability for building the second, third until $j$th tree.
}
Output: The output model IP = $\hat{y}_i$ is obtained by collecting $f$ of $j$ trees as shown in
Equation (15)
End
```

Table 5. Performance metrics for each classifier

The metrics	Logistic regression	SVM classifier	XGBoost classifier
Accuracy	0.81	0.82	0.87
Sensitivity	0.96	0.98	0.94
Specificity	0.42	0.42	0.69
F1-Score	0.88	0.89	0.91

The confusion matrices are calculated to evaluate the classification methods, as shown in Figure 8. Productivity and efficiency are measured using the traditional metrics of accuracy, precision, and recall. Precision refers to the model's correct predictions overall predictions. The graphs of the receiver operating characteristic (ROC) curve are shown in Figure 9. The measurements were taken over 20% of the dataset.

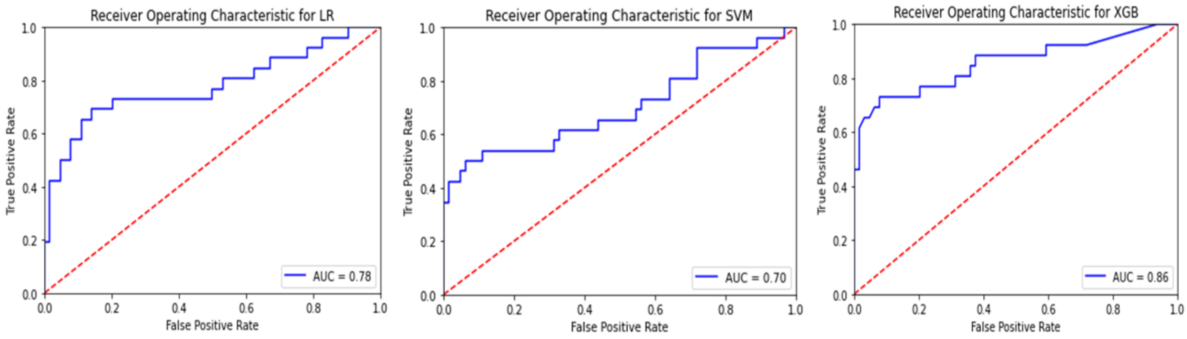


LR

SVM

XGBoost

Figure 8. (color online) Confusion matrices for the LR, SVM, and XGBoost models



LR

SVM

XGBoost

Figure 9. ROC curves and AUCs for LR, SVM, and XGBoost

4.2. **Discussion.** This study aimed to find the relationships between the diagnosis of COVID-19 and the results of some commonly used blood tests, as presented in Table 2. Most associations were significant (p-values less than 0.05), so we used these relationships to build a classification model that helps diagnose COVID-19 rapidly without the need for expensive tests, which are not always available in some countries. The achieved overall accuracy was 0.87, which is relatively good considering the diagnostic problems surrounding COVID-19. Furthermore, depending on these models, it is easier to diagnose COVID-19 depending on just eight blood tests that are easy to make and available in every medical laboratory. This, in turn, will help people in countries that suffer from the lack of unique laboratory materials, which are used newly to diagnose COVID-19 that are hard to find or expensive in many countries under the pressure of a rapidly increasing number of infected people.

Our study was limited to 300 samples (213 not-infected, 87 infected). The accuracy would improve with more samples and could be through using other machine learning algorithms. The accuracies obtained in some studies were better. This is because they employed more tests and more features extracted from the blood test samples (making their procedures more expensive) or used chest X-Ray images. In this study, we utilized eight simple blood tests that could be done in any laboratory. As a result, we obtained a reasonable precision relative to other studies.

A comparison between our proposed model with other related works is presented in Table 6. This table shows the effect of using machine learning algorithms on classification accuracy relative to the number of samples. It is evident that most studies with a limited number of samples lead to low accuracy, for example, [16,17,19]. However, we notice that the proposed model performs better than most of them [17]. This is due to two reasons: first, it depends on the machine learning algorithm, not on the statistical method, which is trained automatically to test the new samples; second, we have more samples.

Table 6. Comparison between some of the related works and our proposed model

Study	**Dataset**	**Algorithm(s)**	**Accuracy %**
Ref. [16]	Blood test with 5644 samples and 18 features	Ensemble learning	99.38
Ref. [17]	Blood test with 207 samples and 7 features	Statistical analysis	70
Ref. [18]	Blood test with 5644 samples and 24 features	MLP, RF, NB, SVM	95.06, 97.3, 97.4, 96.4
Ref. [19]	Blood test with 137 samples and 32 features	SVM	81.4
Ref. [20]	Blood test with 287 samples and 20 features	LR, RF, XGBoost	75, **95**, 92
Our proposed model	**Blood test with 300 samples and 8 features**	**LR, SVM XGBoost**	**81, 82, 87**

5. **Conclusions.** Reducing the spread of COVID-19 is a challenge that has motivated researchers to look for new strategies for the early detection of this dangerous virus. AI is considered a necessity for finding a robust diagnosis in a short time. Due to their efficiency and availability, routine blood tests can be used for an initial diagnosis. A dataset consisting of 300 samples from several laboratories in Iraq was utilized to assess the proposed diagnostic system based on three ML algorithms (LR, SVM, and XGBoost). They were used as classifiers for the early diagnosis of the Coronavirus.

Reasonable results were obtained via the classifier (XGBoost), with the accuracy reaching 87%. However, this percentage could be improved with a more significant number of samples. As a future direction, these results can be enhanced using deep learning algorithms with a reduced number of samples. Furthermore, adopting optimization metaheuristic methods may help omit low effect features and focus only on the optimal used one that affects the diagnosis results. This leads to reducing the test cost while maintaining the same level of accuracy.

REFERENCES

[1] World Health Organization, *Coronavirus Disease 2019 (COVID-19): Situation Report, 67*, 2020.

[2] J. S. Kahn and K. McIntosh, History and recent advances in coronavirus discovery, *Pediatr. Infect. Dis. J.*, vol.24, pp.S223-S227, https://doi.org/10.1097/01.inf.0000188166.17324.60, 2005.

[3] P. A. Catherwood, J. Rafferty and J. McLaughlin, Artificial intelligence for long-term respiratory disease management, *Proc. of the 32nd International BCS Human-Computer Interaction Conference*, pp.1-5, 2018.

[4] E. Alpaydin, *Machine Learning*, MIT Essential Knowledge Series, Cambridge, MA, 2016.

[5] N. C. Caballé et al., Machine learning applied to diagnosis of human diseases: A systematic review, *Applied Sciences*, vol.10, no.15, 2020.

[6] A. Banerjee, S. Ray, B. Vorselaars, J. Kitson, M. Mamalakis, S. Weeks and L. S. Mackenzie, Use of machine learning and artificial intelligence to predict SARS-CoV-2 infection from full blood counts in a population, *Int. Immunopharm.*, vol.86, 2020.

[7] N. Das, M. Topalovic and W. Janssens, Artificial intelligence in diagnosis of obstructive lung disease: Current status and future potential, *Curr. Opin. Pulm. Med.*, vol.24, pp.117-123, 2018.

[8] M. A. Al-Masni, M. A. Al-Antari, J. M. Park, G. Gi, T. Y. Kim, P. Rivera, E. Valarezo, M. T. Choi, S. M. Han and T. S. Kim, Simultaneous detection and classification of breast masses in digital mammograms via a deep learning YOLO-based CAD system, *Comput. Methods Prog. Biomed.*, vol.157, pp.85-94, https://doi.org/10.1016/j.cmpb.2018.01.017, 2018.

[9] D. A. Anggoro and D. Novitaningrum, Comparison of accuracy level of support vector machine (SVM) and artificial neural network (ANN) algorithms in predicting diabetes mellitus disease, *ICIC Express Letters*, vol.15, no.1, pp.9-18, https://doi.org/10.24507/icicel.15.01.9, 2021.

[10] M. A. Al-Masni, D.-H. Kim and T.-S. Kim, Multiple skin lesions diagnostics via integrated deep convolutional networks for segmentation and classification, *Comput. Methods Prog. Biomed.*, vol.190, 2020.

[11] K. Kourou, T. P. Exarchos, K. P. Exarchos, M. V. Karamouzis and D. I. Fotiadis, Machine learning applications in cancer prognosis and prediction, *Comput. Struct. Biotechnol. J.*, vol.13, pp.8-17, 2015.

[12] S. H. Nallamala, P. Mishra and S. V. Koneru, Breast cancer detection using machine learning way, *Int. J. Recent Technol. Eng.*, vol.8, pp.1402-1405, 2019.

[13] M. Gurbina, M. Lascu and D. Lascu, Tumor detection and classification of MRI brain image using different wavelet transforms and support vector machines, *2019 42nd International Conference on Telecommunications and Signal Processing (TSP)*, Budapest, Hungary, https://doi.org/10.1109/TSP.2019.8769040, 2019.

[14] R. J. Al-Azawi et al., Efficient classification of COVID-19 CT scans by using q-transform model for feature extraction, *PeerJ Computer Science*, vol.7, 2021.

[15] R. Kumar et al., Accurate prediction of COVID-19 using chest X-ray images through deep feature learning model with SMOTE and machine learning classifiers, *MedRxiv*, DOI: 10.1101/2020.04.13.20063461, 2020.

[16] M. AlJame, I. Ahmad, A. Imtiaz and A. Mohammed, Ensemble learning model for diagnosing COVID-19 from routine blood tests, *Informatics in Medicine Unlocked*, vol.21, 2020.

[17] D. Ferrari, A. Motta, M. Strollo, G. Banfi and M. Locatelli, Routine blood tests as a potential diagnostic tool for COVID-19, *Clinical Chemistry and Laboratory Medicine (CCLM)*, https://doi.org/10.1515/cclm-2020-0398, 2020.

[18] V. A. de Freitas Barbosa, J. C. Gomes, M. A. de Santana, J. E. de Almeida Albuquerque, R. G. de Souza, R. E. de Souza and W. P. dos Santos, Heg.IA: An intelligent system to support diagnosis of COVID-19 based on blood tests, *Research on Biomedical Engineering*, https://doi.org/10.1101/2020.05.14.20102533, 2020.

[19] H. Yao et al., Severity detection for the coronavirus disease 2019 (COVID-19) patients using a machine learning model based on the blood and urine tests, *Frontiers in Cell and Developmental Biology*, vol.8, 2020.

[20] S. S. Aljameel et al., Machine learning-based model to predict the disease severity and outcome in COVID-19 patients, *Scientific Programming*, vol.10, 2021.

[21] C. L. Luo, Y. Rong, H. Chen, W. Zhang, L. Wu, D. Wei et al., A logistic regression model for non-invasive prediction of AFP-negative hepatocellular carcinoma, *Technol. Cancer Res. Treat.*, vol.18, no.2, DOI: 10.1177/1533033819846632, 2019.

[22] C.-Y. J. Peng, K. L. Lee and G. M. Ingersoll, An introduction to logistic regression analysis and reporting, *J. Educ. Res.*, vol.96, pp.3-14, https://doi.org/10.1080/00220670209598786, 2002.

[23] N. Cristianini and E. Ricci, Support vector machines, in *Encyclopedia of Algorithms*, M. Y. Kao (ed.), Boston, MA, Springer, https://doi.org/10.1007/978-0-387-30162-4_415, 2008.

[24] T. Chen and C. Guestrin, XGBoost: A scalable tree boosting system, *Proc. of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp.785-794, 2016.

Author Biography

Ahmed Yacoub Yousif received the B.Sc. degree in Mathematics and Computer Applications, from AL-Nahrain University, Baghdad, Iraq, 2008; the M.Sc. degree in Mathematics and Computer Applications, from AL-Nahrain University, Baghdad, Iraq, 2014.

Ahmed currently works an assistant lecturer at the Information Technology Center, University of Technology, Iraq, since 2009. His researches interested are reliability theory, fuzzy-logic, graph theory, and chaos theory.

Sanar Mazin Younis received the B.Sc. degree in Mathematics and Computer Applications, from AL-Nahrain University, Baghdad, Iraq, 2008; the M.Sc. degree in Mathematics and Computer Applications, from AL-Nahrain University, Baghdad, Iraq, 2015.

Sanar currently works as teacher at the General Directorate of Education Rusafa 2, Ministry of Education, Iraq, since 2010. His research interested are project networks scheduling, operations management, fuzzy system modeling and applications.

Samer Alaa Hussein received the B.Sc. degree in Computer Science – Artificial Intelligence, from the University of Technology, Baghdad, Iraq, 2009; the M.Sc. degree in Computer Science – Artificial Intelligence, from the University of Technology, Baghdad, Iraq, 2017.

Samer is currently an assistant lecturer at the Information Technology Center, University of Technology, Iraq. His research interests are in the artificial intelligence field.

Nadia Mohammed Ghanim Al-Saidi is a professor in the Department of Applied Sciences, University of Technology-Baghdad-Iraq since 2011. She completed her Bachelor of Science and Master of Science degrees in applied mathematics, from Department of Applied Sciences-University of Technology, Baghdad, Iraq, in 1989, and 1995, respectively. She received her Ph.D. degree in mathematics and computer applications sciences from Al-Nahrain University, Baghdad, Iraq in 2003. In 1989 she joined the Department of Applied Sciences at University of Technology as an academic staff member. She also joined the Institute for Mathematical Research (INSPEM), University Putra Malaysia (UPM) as a post doctorate researcher from 2008-2010 with the research project "Fractals in Cryptography". Prof. Dr. Nadia is the author of numerous technical papers since 1994, her research interests include cryptography, fractal geometry, chaos theory, and graph theory.