Journal of High School Science

Enhancing the COVID-19 diagnostic for patients using machine learning and voting classifier approaches

Brian T.Y[1], Estrada A.M[2], Hsu N[3], Raaief R[4], Xu H[5], Tseng T.L[6], Emon S.H[7]

Submitted: August 23, 2024, Revised: version 1, November 1, 2024
Accepted: November 14, 2024

Abstract

The COVID-19 pandemic highlighted the need for prompt diagnostic techniques in order to control the disease's spread. This study, which focuses on the innovative use of machine learning (ML) models, intends to improve the efficiency and accuracy of diagnosis. Five supervised machine learning models were employed using a dataset acquired from the University of Texas Medical Branch at Galveston (UTMB) consisting of 1987 patient records with 30 vital measurements. The study utilized the ML models, including Support Vector Machine (SVM), Classification and Regression Tree (CART), Random Forest (RF), Artificial Neural Network (ANN), and Voting Classifier (VC). Using the PCR label and the remaining 29 vital measurements as the features, the models were trained to determine whether the feature combination was associated with COVID-19 infection. Data preprocessing included outlier detection, oversampling for class balance, and multicollinearity analysis through Variance Inflation Factor (VIF) analysis, confirming the data quality. By eliminating outliers and correcting the bias through the data preprocessing steps, a refined dataset was prepared for model training. The voting mechanism combined with a Support Vector Machine (SVM), Random Forest (RF), and Artificial Neural Network (ANN) yielded an accuracy of 0.804 and an Area Under Curve (AUC) of 0.85. Dimensionality reduction and feature contribution analysis were performed to obtain a comprehensive understanding of the model building process. These findings point to the potential application of machine learning models in diagnosing COVID-19.

Keywords

COVID-19, Diagnostics, Classification, Artificial Intelligence, Machine Learning, Voting Classifier, Accuracy of diagnosis, Dimensionality reduction, Feature engineering, Hyperparameters

[1]Brian Tseng, Franklin High School, 900 Resler Dr, El Paso, TX 79912, USA. btseng1030@gmail.com
[2]Alejandro M. Estrada, Franklin High School, 900 Resler Dr, El Paso, TX 79912, USA. aestra9116@episd.org
[3]Nathan Hsu, [4]Raaief Raaief, Coronado High School, 100 Champions Pl, El Paso, TX 79912, USA. nhsu009640@episd.org mraaie0744@episd.org
[5]Honglun Xu, Department of Industrial, Manufacturing, and Systems Engineering (IMSE), University of Texas at El Paso, 500 W. University Ave., El Paso, TX 79968, USA. hxu4@utep.edu
[6]Corresponding author: Tzu-Liang Tseng, Professor and Chair of the Department of IMSE, Director, Research Institute for Manufacturing & Engineering Systems (RIMES), University of Texas at El Paso, 500 W. University Ave., El Paso, TX 79968, USA. btseng@utep.edu
[7]Solayman H. Emon, Mathematical Science, University of Texas at El Paso, 500 W. University Ave., El Paso, TX 79968, USA. semon@miners.utep.edu

1 Introduction

The first case of the SARS-CoV-2 virus, which causes COVID-19, was reported in Wuhan, China, in December 2019. The virus (1) rapidly spread worldwide, leading the World Health Organization (WHO) to declare it a global pandemic by March 11, 2020. This unprecedented situation brought about significant challenges, profoundly affecting socio-economic conditions and reshaping daily life across the globe. In response, governments and health organizations implemented various measures to curb the spread of the virus, including lockdowns, travel bans, and social distancing. The urgency of the situation underscored the necessity of swift and effective measures to prevent further transmission of COVID-19.

As the pandemic unfolded, the limitations of traditional diagnostic methods became increasingly apparent. The pandemic catalyzed technological advancements in the medical field, particularly in diagnostics, treatment plans, and vaccine development. Manual diagnostics for COVID-19, primarily based on PCR testing and chest imaging, posed several challenges. These methods placed immense demands on healthcare resources, often resulting in delayed results and increasing the potential for human error. The strain on medical infrastructure highlighted the need for more efficient and accurate diagnostic approaches. These limitations of manual diagnosis can potentially be addressed by automation techniques such as machine learning. Machine learning algorithms can potentially offer promising solutions with high predictive accuracy utilizing the clinical datasets. By leveraging the power of automation, machine learning has the capability to revolutionize the diagnostic landscape, making it more adaptable to emerging health crises.

The primary goal of this study was to apply supervised machine-learning models to improve the precision and effectiveness of COVID-19 diagnostics. Four different machine learning models—Support Vector Machine (SVM), Classification and Regression Tree (CART), Random Forest (RF), Artificial Neural Network (ANN), and a Voting Classifier (VC) combined with other multiple classifiers were employed to predict COVID-19 infection in patients. These models utilized supervised learning, where labeled data was used to train the algorithms, allowing them to make accurate predictions on new, unseen (test) data.

2 Data description

The research utilized the data obtained from the University of Texas Medical Branch at Galveston (UTMB) containing COVID-19 screening diagnostics—the dataset from UTMB comprised records of 1987 patients, each including 30 vital measurements. The Polymerase Chain Reaction (PCR) measurement was the sole measurement for identifying COVID-19 infections in patients. All measurements were taken within a close timeframe of approximately two weeks, ensuring the relevance of all data collection. Moreover, the dataset included ICD codes (International Classification of Diseases codes). The ICDs are standardized alphanumeric codes used by healthcare providers to classify all diagnoses, symptoms, and procedures. These codes are essential for systematically recording, analyzing, and interpreting health information. Table 1 depicts a detailed description of the UTMB dataset, outlining each feature's definition and its relevance to the patient's information.

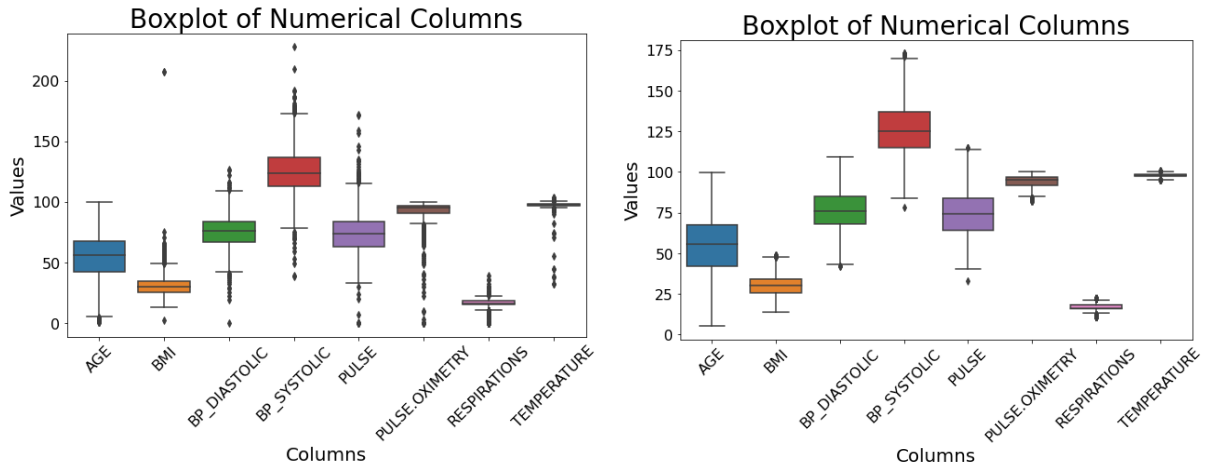
Table 1. Overview of the UTMB dataset

Feature	Description
SEX (binary)	Gender of the patient
ETHNICITY (binary)	Self-reported ethnicity of the patient
AGE	Age of the patient in years
BMI	Body Mass Index, a measure of body fat based on height and weight
BP_DIASTOLIC	Diastolic blood pressure, measuring pressure in the arteries when the heart rests between beats
BP_SYSTOLIC	Systolic blood pressure, measuring pressure in the arteries when the heart beats
PULSE	Heart rate in beats per minute
PULSE.OXIMETRY	Oxygen saturation in the blood, as a percentage
RESPIRATIONS	Respiratory rate in breaths per minute
TEMPERATURE	A patient's temperature
Z20.828(binary)	Exposure to viral communicable eases
R05(binary)	Cough
R50.9(binary)	Unspecified fever
I10(binary)	Essential (primary) hypertension
R07.9(binary)	Unspecified chest pain
J18.9(binary)	Unspecified pneumonia
R06.9(binary)	unspecified abnormalities of breathing
R06.00(binary)	Dyspnea, unspecified
R09.02(binary)	Hypoxemia
Z11.59(binary)	Screening for other viral diseases
B34.9(binary)	Unspecified viral infection
N17.9(binary)	Acute kidney failure that is unspecified
J02.9(binary)	Acute pharyngitis, unspecified
R07.89(binary)	Other chest pain
R17.9(binary)	Hyperbilirubinemia without jaundice
R53.1(binary)	Weakness, diminished or absent energy and strength, or a lack of concentration
R52(binary)	Pain that is unspecified
E11.9(binary)	Type 2 diabetes mellitus
I50.9(binary)	Unspecified heart failure
R06.02(binary)	Shortness of breath
PCR (binary)	PCR test result for COVID-19, indicating virus detection status

3 Methods

3.1 Data preprocessing

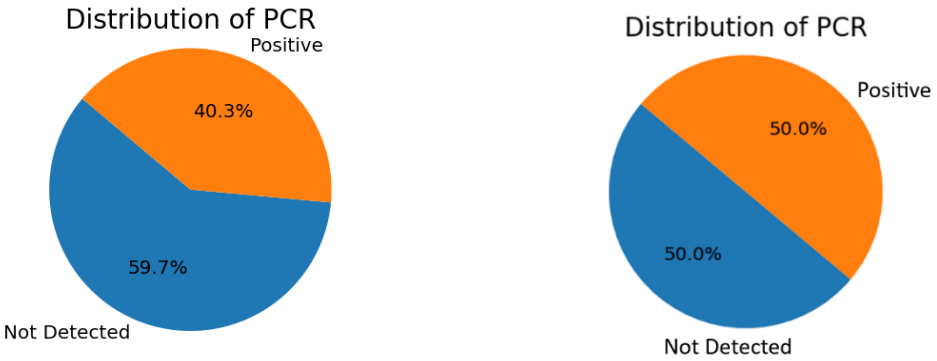
Datasets of all kinds are prone to human error. This may influence the quality of the data, which also might have an impact on the performance of ML models. Data processing is an important procedure that ensures quality data, which can perform better compared to models using unprocessed data. Although the UTMB dataset did not contain any missing values, it had a considerable number of outliers. Box plots or box and whisker plots help to identify outliers that can shift data quality. Outliers were identified and addressed using box plots, as shown in Figure 1. This visualization highlights the distribution of data before and after cleaning, where outliers outside the range indicated by the dashed lines were removed.



(a) Before removing outliers (b) After removing outliers

Figure 1. The box plot of UTMB (continuous features). The left panel (a) depicts before cleaning outliers and the right panel (b) is after the outliers cleaning.

Additionally, data standardization was applied to continuous features to reduce computational time and ensure consistent scaling. Originally, the covid dataset contained 1987 observations. In Figure 1(a), we performed outlier analysis, then removed those outlier observations with Inter Quartile Ratio (IQR) bounds. We were left with 1860 observations for model building. In this classification task, the variable 'PCR' was designated as the response variable. To eliminate bias in the model building process due to class imbalance, we used the oversampling (2) technique to balance the minority class, resulting in an equal ratio of 0s and 1s (no-Covid and Covid respectively), as shown in Figure 2. The minority class increased to match the majority class at a 50/50 balance. This approach ensured that the model was less prone to overfitting to the majority class. The dataset was divided into an 75/25 split, with 75% of the data used for training and 25% reserved for testing. This split ensured that the models were trained on a substantial portion of the data while being evaluated on unseen data to assess their generalization capability.



(a) Imbalanced class distribution (b) Balanced class distribution (Oversampling)

Figure 2. Class distribution before and after data balancing.

The correlation analysis shown in Figure 3 visually represents the relationship between various clinical variables. Each cell of the correlation matrix corresponds to the correlation coefficient between variables. The color intensity and size of the circles represent the correlation magnitude. The correlation coefficients range from -1 to 1 indicating the strength and direction of their linear relationship.

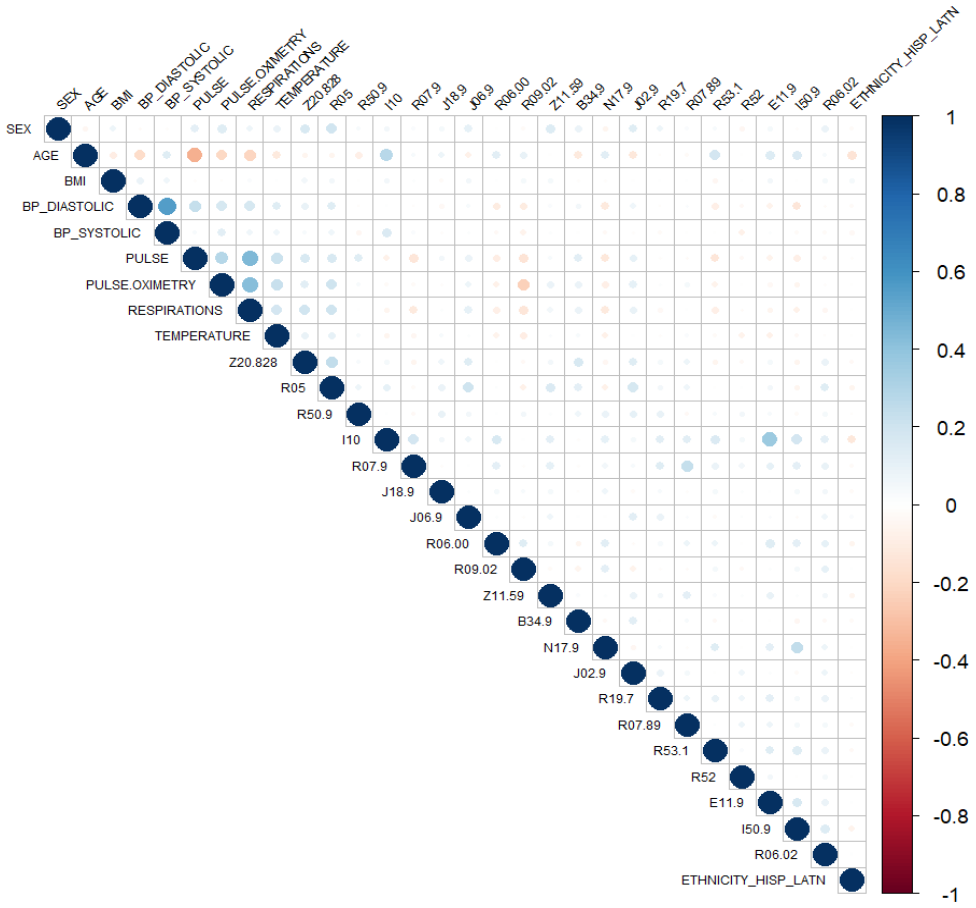


Figure 3. Correlation analysis among features.

Since the proposed work was related to clinical research, the accurate modeling of relationships between variables was crucial to the decision making process. The presence of multicollinearity among variables could affect the relative magnitude of the coefficients of the explanatory variables. To assess the multicollinearity, we performed Variance Inflation Factor (VIF) analysis as shown in Figure 4.

Since our data was high dimensional with ~30 features we further performed principal component analysis (Figure 5).

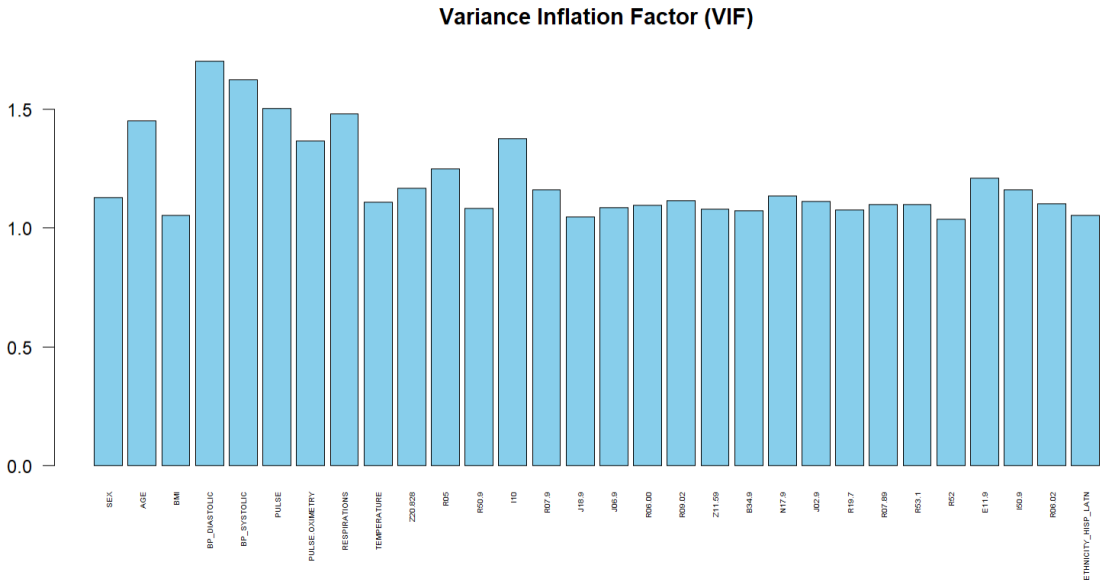


Figure 4. Assessment of multicollinearity among features.

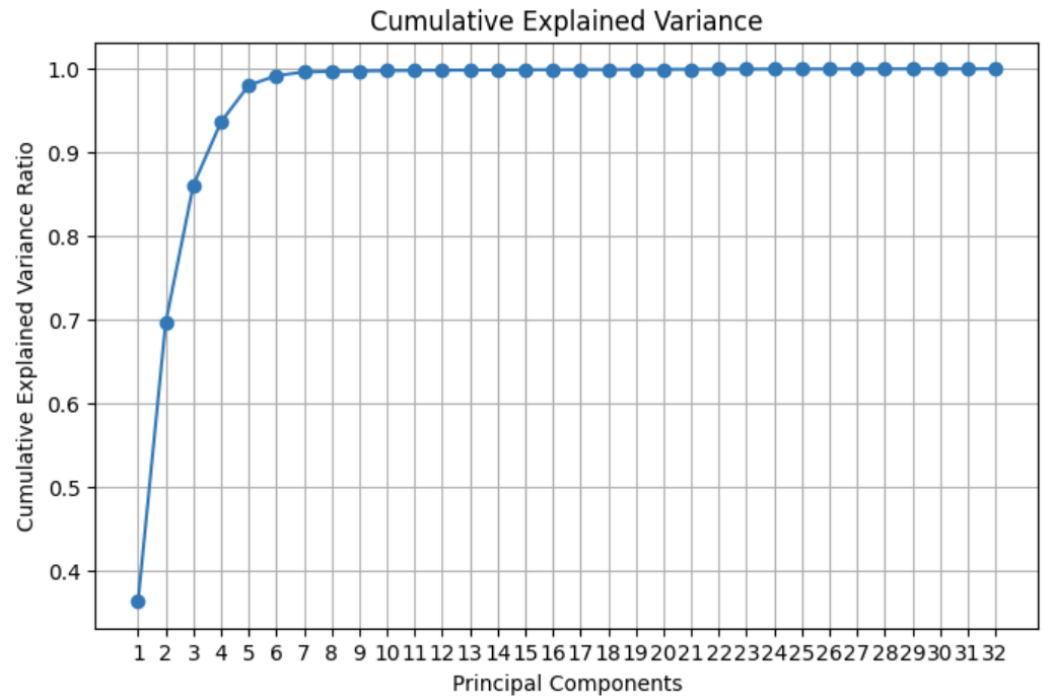


Figure 5. Analysis of principal components (PCs).

Five different machine learning models were selected initially, including Support Vector Machine (SVM), Classification and Regression Tree (CART), Random Forest (RF), Artificial Neural Network (ANN). Additionally, we employed an ensemble method (Voting Classifier) which integrated the predictions of SVM, RF and ANN models. By leveraging the strengths of these diverse ML models, the Voting Classifier (VC) aimed to achieve higher accuracy and robustness.

3.1 Support Vector Machine

Support vector machine (SVM) is a machine learning supervised model used for both classification and regression tasks (3). An SVM utilizes an optimized hyperplane to achieve an optimal distance between different data point

classes. Even with linearly inseparable data, SVM can convert that data into a multidimensional space and can, therefore, find a linear margin to classify the data correctly. However, this process can take a lot of time to complete, hence an SVM can implement a strategy called the kernel trick, where a function calculates the relationship between each point as if it were in that higher dimension without it transforming into that dimension.

The most widely used and flexible kernel function is the Radial Basis Function (RBF). In this context, we applied Radial Basis Function (RBF) kernel function mathematically expressed by Equation 1. The convergence in SVM is based on the optimization problem through finding maximum boundary/appropriate support vectors rather than iterating over epochs. When the solver algorithm converges on an optimal solution, the training process is completed. To identify the most effective parameter combination for the SVM classifier in the context of the UTMB dataset, we utilized a systematic exploration approach called grid search cross-validation (GridSearchCV) (4). The GridSearchCV was applied to perform an extensive search over the specified parameter grid. In this setup, we used a 5-fold cross-validation, which balanced computational efficiency with the reliability of the evaluation metrics. The parameters within the search space included different values for the penalty parameter C (optimal value = 6), the type of kernel function (rbf), and the kernel coefficient gamma (γ). The final classifier, fine-tuned with these optimal hyperparameters, demonstrated improved performance on the training dataset, ensuring robust generalization to unseen data. For the training dataset, the final classifier, was configured with the optimal hyperparameters. The kernel function is given by, $k(x_i, x_j) = exp\left(-\gamma \left|\left|x_i - x_j\right|\right|^2\right)$ (eq1), where $k(x_i, x_j$ represents the kernel function, x_i and x_j representing points in the plane, the expression, $\left|\left|x_i - x_j\right|\right|^2$ calculates the squared distance between the points x_i and x_j and the gamma parameter, γ , determines the influence or importance assigned to each individual training sample. Low gamma causes a smoother decision boundary and high gamma causes a more complex decision boundary. In simple terms, the kernal coefficient, γ controls the overfitting of the model.

3.2 Classification and Regression Tree

A classification tree (5) is a standard method in machine learning known for its decision-making process. It falls under a type of decision tree that predicts classifications. The main parts of a decision tree are the root node, internal nodes, branches, and leaf nodes. Figure 6 depicts the structure of a decision tree. The root node is the first node. This refers to the entire set of datasets. The root node divides into branches that define different feature values. Internal nodes represent features and conditions for splitting the dataset, and branches represent possible values of the feature at that node. Finally, leaf nodes do not split further, and each one of them corresponds to a class label or a predicted value. Each split is performed by calculating the best feature to split using a criterion which can be the Gini, entropy, or log loss criterion. We utilized the Gini Impurity (presented in equation 2) criterion for the modeling purpose, as $Gini(D) = \sum_{k=1}^{K} p_k\,(1 - p_k) = 1 - \sum_{k=1}^{K} {p_k}^2$ (eq2), where $Gini(D)$ is the Gini impurity of the dataset D, k is the number of classes in the target variable and p_k is the proportion of class k within the subset of the dataset.

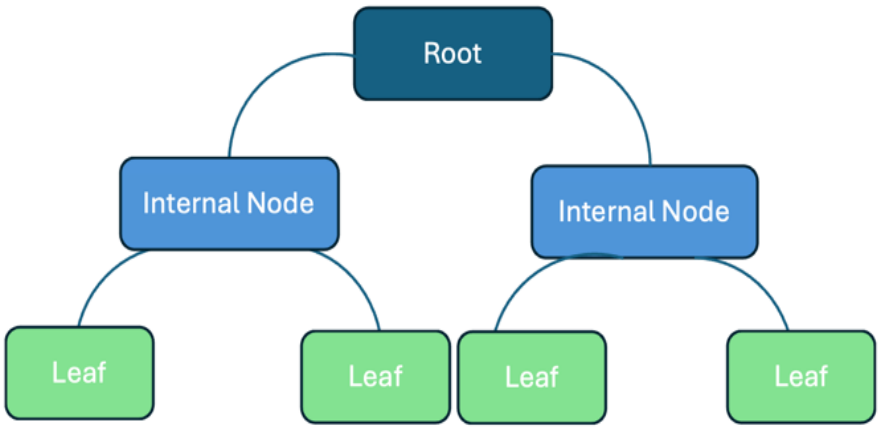


Figure 6. Overview of a decision tree.

The Classification and Regression Tree (CART) model was employed with specific hyperparameters to optimize its performance. The model converges within the criterion of gini impurity, since Gini impurity is typically more robust than classification error. From grid search cross validation, the final model parameters of maximum depth, minimum samples split and minimum samples leaf were returned as 16, 2 and 90 respectively.

The Gini impurity criterion was utilized as the splitting measure. It quantifies the impurity of the dataset by assessing the probability of incorrectly classifying a randomly chosen element. This helps to determine how well the split separates different classes. The maximum depth of the tree was set to 16, allowing the model to capture intricate patterns in the data without overfitting. To further refine the model's structure, the minimum number of samples required to split an internal node was set to 2, enabling the model to make splits even with the smallest number of data points. Additionally, the minimum number of samples required to be at a leaf node was constrained to 90, ensuring that each leaf node represents a sufficiently large subset of the data. This combination of parameters aimed to balance the model's complexity and its ability to generalize to unseen data, thus providing reliable predictions.

3.3 Randomized Forest

Random Forests (RF) are popular due to their simplicity, flexibility, and high predictive performance across a variety of datasets. During the training stage of the algorithm, several decision trees were constructed. Each tree in the forest is made using a subset of the training data and a random selection of features. This randomness helps to differentiate the trees and minimize overfitting, leading to a more generalized model. The predictive capabilities are derived from the majority rule of the outputs. For classification tasks, the prediction is what most of the trees labeled the output to be; this works similarly to a 'majority rule' system. In highly imbalanced datasets where one class is significantly more prevalent than others, the majority class may dominate the decision-making process within individual trees. As a result, the algorithm may struggle to accurately represent and predict the minority class, leading to biased predictions. However, we previously balanced the class distribution in the data pre-processing stage to mitigate this issue. In our study, we utilized a Random Forest (6) classifier with parameters optimized using GridSearchCV technique. The best model

selected returned 300 estimators (trees) and 10 as the maximum depth, determined through cross-validation on the training data.

3.4 Artificial Neural Network

The Artificial Neural Network (ANN) model is a popular choice in machine learning that can perform both classification and regression tasks. The ANN (7) excels at handling and interpreting complex, non-linear relationships in datasets. An ANN has a structure like that of a biological neural network. To start, the architecture of the ANN consists of three different layers: input layer, hidden layers, and the output layer (Shown in Figure 7). In an ANN model, each node is attached to a weight value (w) and a bias value (b). The output is decided by using an activation function which has a different threshold used to classify the output. In this classification task, we utilized sigmoid/logistic activation function (8). This activation function assigns the input values to a range between 0 and 1, which can be interpreted as False (0) and True (1). It can be defined by the mathematical formula $f(x) = \frac{1}{1+e^{-x}}$ eq(3), where, x is the function's input and e is the natural logarithm's base. The key characteristic of the sigmoid function is its S-shaped curve, which allows it to smoothly transition between 0 and 1.

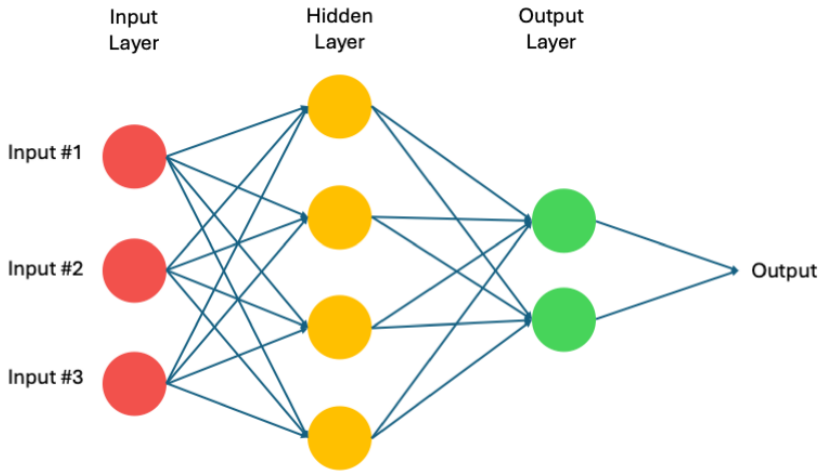


Figure 7. Example of a shallow artificial neural network (ANN).

Training an ANN involves hyperparameter tuning and changing weight and bias values. This is done by comparing the model's outputs to the actual targets and constantly updating the weights using techniques such as backpropagation. The model converges with maximum number of iterations by utilizing the binary cross-entropy loss. A comprehensive hyperparameter tuning approach using GridSearchCV was employed to optimize the ANN classifier for the task. The grid search systematically explored various configurations, including different hidden layer sizes, activation functions, solvers, and regularization strengths. After evaluating each configuration using 5-fold cross-validation, the model with the highest accuracy on the validation set was selected. The optimal hyperparameters were 100 for the Hidden layer size, 0.001 for the learning rate, and 200 for the maximum number of iterations. The activation function, optimizer and loss functions used were ReLU, Adam and Binary cross-entropy respectively.

3.5 Voting Classifier

In machine learning, a voting classifier combines several models to enhance predictive performance. This ensemble technique (9) leverages the diversity of different models, operated by aggregating the predictions of each individual model and choosing the outcome based on a majority vote (10) or by averaging the predicted probabilities. There are two ways to cast a vote: hard and soft voting. Hard voting can be described as the simplest case of majority voting. In contrast, soft voting obtains the averages of the models. Figure 8 illustrates the process of using an ensemble of classification models to make a final prediction through a voting mechanism.

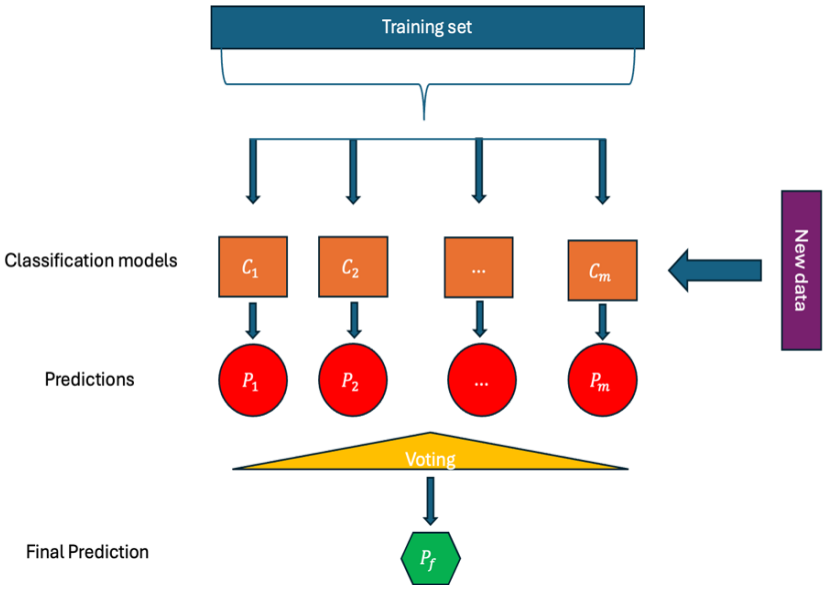


Figure 8. A voting classifier combining the predictions of multiple classification models.

Initially, a training set is used to train multiple classification models , C_2 ,….. C_m . Each of these models is trained using the same training data but may represent different algorithms or approaches to classification. Once trained, these models are used to make individual predictions (P_1 , P_2 ,….. P_m) based on new data. These predictions are then combined by a voting mechanism, which can be either hard voting (where the majority prediction is chosen) or soft voting (where predictions are weighted and averaged). The outcome of this voting process is the final prediction P_f which benefits from the combined strengths of all the models, thereby enhancing the overall accuracy of the prediction. In the context of this study, we utilized the voting mechanism with the combination of Support Vector Machine (SVM), Random Forest (RF), and the Artificial Neural Network (ANN). For the combination purpose, we used Hard Voting mechanism from each classifier C_m represented by = $mode$ $\{P_1(x), P_2(x), \ldots \ldots, P_m(x)\}$ (eq4). Assuming a combination of the three classifiers that classified the sample as follows $Classifier1(SVM) \rightarrow Class0$, $Classifier2(RF) \rightarrow Class1$, $Classifier3(ANN) \rightarrow Class1$. The voting classifier output would be $= mode\{0,1,1\} = 1$. In this majority voting system, the training sample was classified as class 1.

3.6 Performance evaluation

To comprehensively evaluate the performance of the machine learning models, several metrics (11) were utilized such as accuracy, precision, recall, F1-score, Receiver Operating Characteristics (ROC) curve, and confusion matrix. A confusion matrix (12) was used to rigorously analyze the model performance and gain a deeper understanding of its performance. The matrix consisted of a two-by-two array containing real output values on one axis and predicted output values on the other axis. Four values were contained within the matrix: TP (true positive), FP (false positive), FN (false negative), and TN (true negative), as shown in Figure 9.

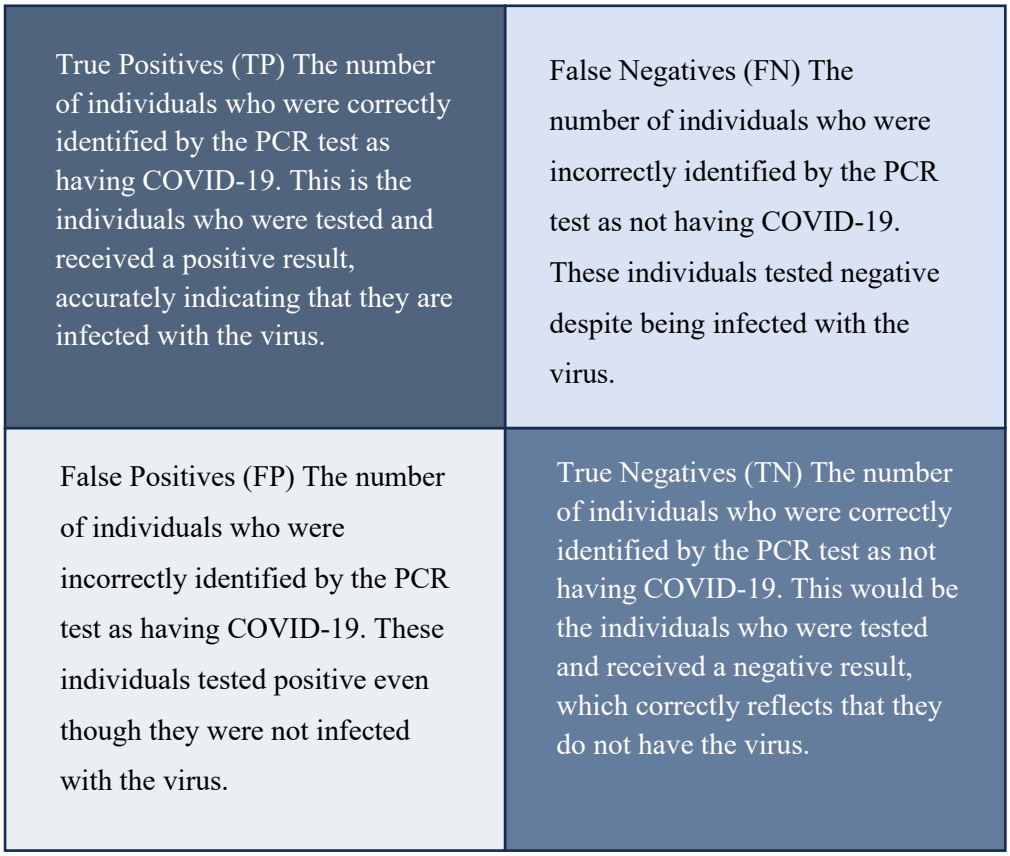


Figure 9. Example of confusion matrix.

Classification accuracy is the ratio of correct predictions to the sum of all predictions made. It can be defined from the confusion matrix as Accuracy = $\frac{TP+TN}{TP+FN+TN+FP}$ (eq5). Precision is the proportion of true positive predictions relative to the total number of positive predictions made for a specific class. Precision informs how likely it is that positive predictions returned by the model are truly positive. The metric recall shows how well the model is at predicting positive instances. It can be mathematically represented as Precision = $\frac{TP}{TP+FP}$ (eq6). Recall is the ratio of correct positives predicted to the sum of all samples that should have been positive in one class . Recall = $\frac{TP}{TP+FN}$ (eq7). The F1 score is the harmonic mean of the calculated precision and recall.

$$F1Score = 2 \times \frac{(Precision \times Recall)}{(Precision + Recall)}$$ (eq8).

The greater the F1 score, the greater the precision and recall values.

The Receiver Operating Characteristics (ROC) curve is a graph of two parameters: actual positive rate (TPR) and false positive rate (FPR). The two-dimensional area under the ROC curve is referred to as Area Under the Curve (AUC). It is a value between 0 and 1, with a value of 0 indicating that the model predictions are equivalent to random predictions, and 1 indicating that the predictions are 100% correct. AUC can be interpreted as the probability that the classification model ranks a random positive sample more highly than a random negative sample in the dataset.

4 Results

In this section, we present and analyze the performance of various machine learning models applied to the COVID-19 diagnostic dataset. We discuss the results of individual models, including Support Vector Machine (SVM), Classification and Regression Tree (CART), Random Forest (RF), and Artificial Neural Network (ANN). Furthermore, we examine the performance of the ensemble Voting Classifier, which combines the strengths of the top-performing models.

Figure 1 shows boxplots of numerical columns before and after removing obvious outliers from an IQR. The final dataset excluded the rows that contained outliers from any of these eight categories. Figure 2a shows the class imbalance between the Covid positive (40.3%) and Covid negative (59.7%) cases as determined by PCR analysis. The class imbalance was adjusted to a 50:50 ratio (Figure 2b) after Random oversampling (ROS), as described in the Methods section. Two other methods – Random Under Sampling (RUS) and Synthetic Minority Over-sampling Technique (SMOTE) - for correcting class imbalance were also used. The ROS method yielded the best performance as shown in Table 5. From an examination of Figure 3, examples of significant positive correlations can be seen between pulse-oximetry and pulse with respirations, Essential hypertension (I10) and unspecified heart failure (150.9), and systolic and diastolic blood pressure. Similarly, examples of significant negative correlations are evident between pulse and respirations with age and screening for other viral diseases (Z11.59) and Pulse oximetry. The correlogram in Figure 3 thus served as a visual aid to estimate multicollinearity. Figure 4 shows the results of quantification of feature collinearity using the VIF analysis. Since the VIF for all the features was < 2, there was determined to be no multicollinearity among the studied variables. Even though the VIF returned no evidence of multicollinearity, only 8 of the 30 features studied were continuous; the others being binary. This implied that there was a possibility to explain much of the variance in the data using a considerably lesser number of features. A PCA analysis returned a graph (Figure 5) wherein 7 principal components could explain $> 98\%$ variance in the data.

The evaluation metrics, including accuracy, precision, recall, F1-score, and Area Under the Curve (AUC), provided a comprehensive assessment of each model's effectiveness in predicting COVID-19 cases.

Table 2. Performance comparison between ML models.

Method	Accuracy	Precision	Recall	F1-score
SVM	0.723	0.722	0.737	0.730
CART	0.690	0.707	0.665	0.686
RF	0.763	0.756	0.788	0.772
ANN	0.740	0.724	0.787	0.755

Table 2 summarizes the performance metrics of the individual models, including accuracy, precision, recall, F1-score. The metrics were calculated based on the confusion matrixes of the individual models. The Support Vector Machine (SVM) model achieved an accuracy of 72.3%, precision of 72%, recall of 73.7%, F1-score of 73%. This indicated a reasonable ability to discriminate between COVID-19 positive and negative cases. The Classification and Regression Tree (CART) model returned the lower performance among the models, with an accuracy of 69%, precision of 70.7%, recall of 66.5%, F1-score of 68.6%. It reflects moderate discrimination ability for COVID-19 cases. The Random Forest (RF) model outperformed others ML classifiers with an accuracy of 76.3%, precision of 75.6%, recall of 78.8%, and F1-score of 77.2%. By averaging multiple decision trees, the Random Forest model reduced overfitting and improved generalization, making it one of the top performers. The ANN model demonstrated competitive performance, with an accuracy of 74%, a precision of 72.4%, a recall of 78.7%, and an F1-score of 75.5%.

Table 3. Performance metrics of different models across cross-validation folds.

Model	Fold	Accuracy	Precision	Recall	F1
SVM	fold1	0.715	0.731	0.710	0.723
	fold2	**0.723**	**0.722**	**0.737**	**0.730**
	fold3	0.713	0.715	0.713	0.713
	fold4	0.710	0.711	0.710	0.709
	fold5	0.713	0.715	0.713	0.713
CART	fold1	0.678	0.660	0.662	0.673
	fold2	0.680	0.691	0.679	0.701
	fold3	0.712	0.671	0.663	0.682
	fold4	**0.690**	**0.707**	**0.665**	**0.686**
	fold5	0.658	0.680	0.667	0.688
RF	fold1	0.742	0.742	0.742	0.742
	fold2	**0.763**	**0.756**	**0.788**	**0.772**
	fold3	0.749	0.752	0.749	0.748
	fold4	0.720	0.721	0.720	0.720
	fold5	0.699	0.700	0.699	0.698
ANN	fold1	0.721	0.730	0.731	0.711
	fold2	0.753	0.755	0.753	0.752
	fold3	**0.740**	**0.724**	**0.787**	**0.755**
	fold4	0.717	0.709	0.717	0.721
	fold5	0.731	0.711	0.721	0.731

Table 3 shows the metrics results across cross-validation folds. By analyzing results from each fold, we can observe the variance in metrics like Accuracy, Precision, Recall, and F1-score, which highlights the model's stability and robustness. This complete presentation ensured that any anomalous variability was detected, facilitating a more accurate assessment of each

model's reliability. Additionally, Figure 10 shows the ROC curve and confusion matrices of the respective classifiers, providing a visual representation of the results presented in Table 2.

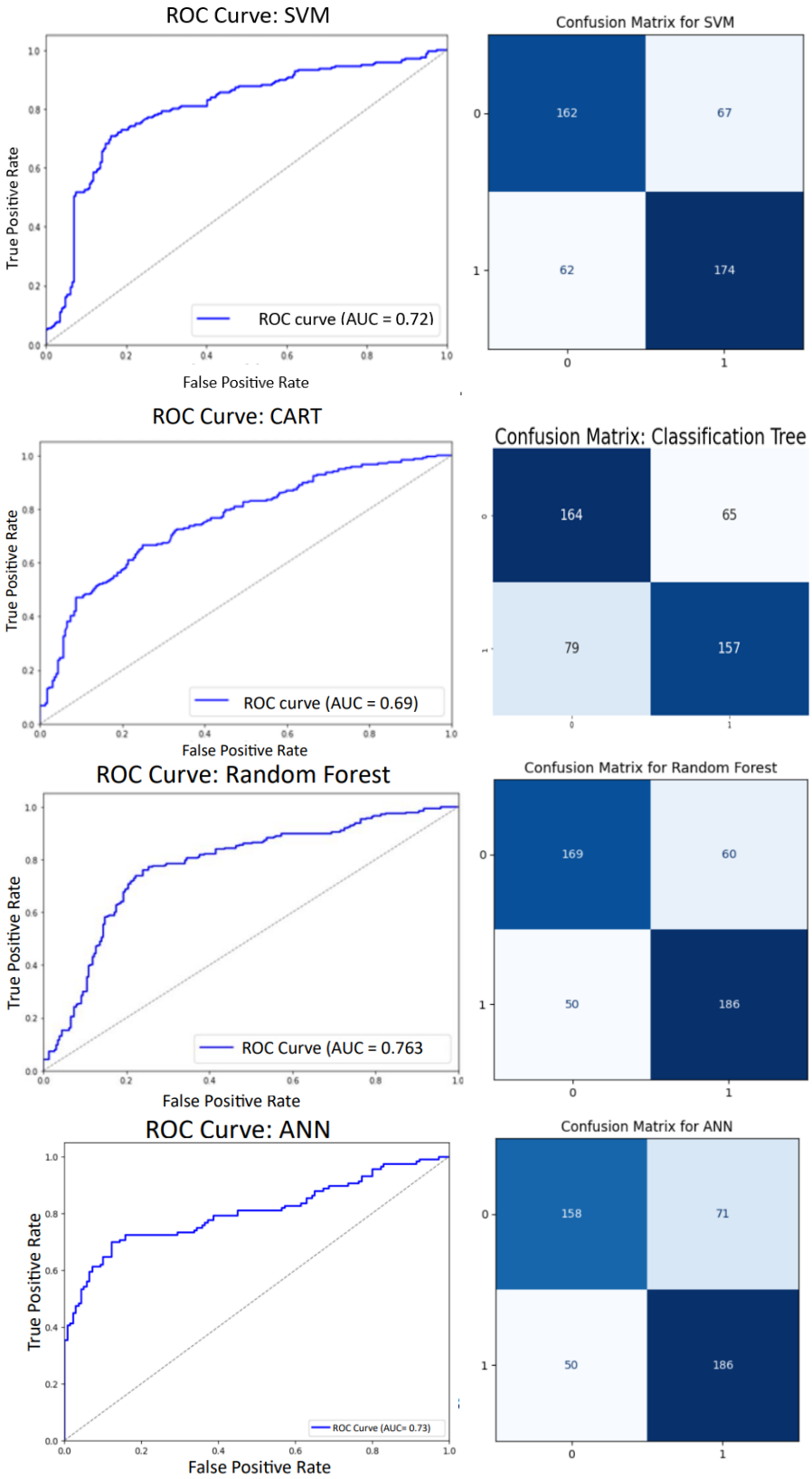


Figure 10. ROC curve and confusion matrix of different classifiers.

We performed a feature importance analysis for four different machine learning models—Support Vector Machine (SVM), Classification and Regression Trees (CART), Random Forest (RF), and Artificial Neural Network (ANN)—using the Local Interpretable Model-agnostic Explanations (LIME) technique. Each subplot of Figure 11 displays the contribution of individual features to the model's predictions, indicating their influence in distinguishing between the "Not Detected" and "Positive" classes. The overall figure provides insight into the distinct patterns of feature influence across different models, showcasing LIME's interpretability in identifying critical predictors within each model's decision-making process.

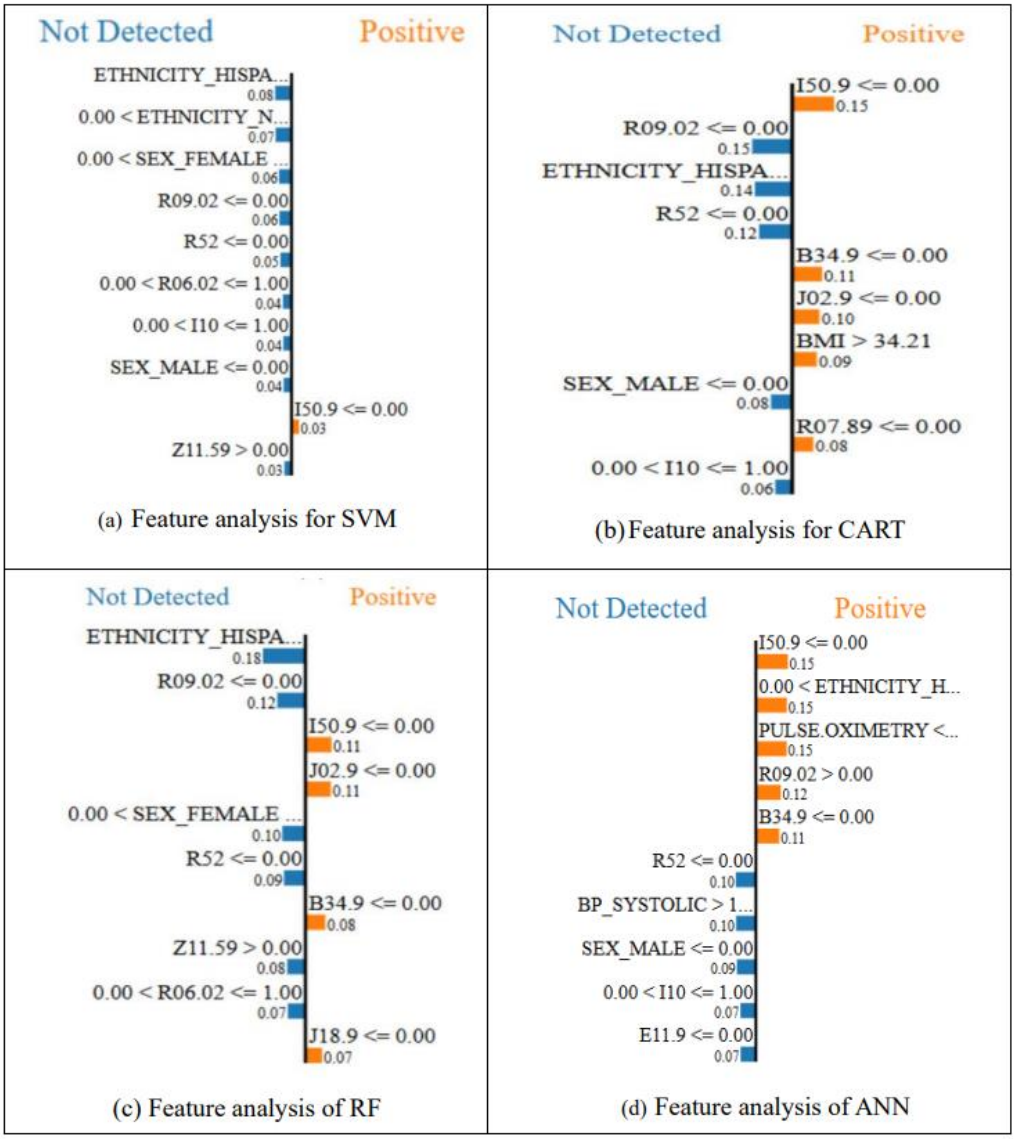


Figure 11. Feature contribution analysis for different machine learning models using LIME.

The hard voting combined with classifiers SVM, RF and ANN yielded the greatest accuracy and AUC of all models. Table 4 presents the performance the Voting Classifier that utilized a hard voting mechanism. The values in the table demonstrate that the Voting Classifier performed well across all metrics, with scores ~ 0.80. The Voting Classifiers, tended to be more resistant to overfitting compared to the individual models. This

occurred because diverse models average out the biases and reduce variance, leading to a more balanced and generalized model. Figure 12 depicts the AUC (Area Under the Curve) and confusion matrix of the Voting Classifier. The AUC value was 0.85, which indicated that the classifier possessed a strong ability to discriminate between classes, with a higher score representing better performance.

Table 4. Performance metrics of voting classifier with hard voting mechanism.

Method	Mechanism	Accuracy	Precision	Recall	F1-score
Voting Classifier	Hard Voting	0.804	0.809	0.805	0.807

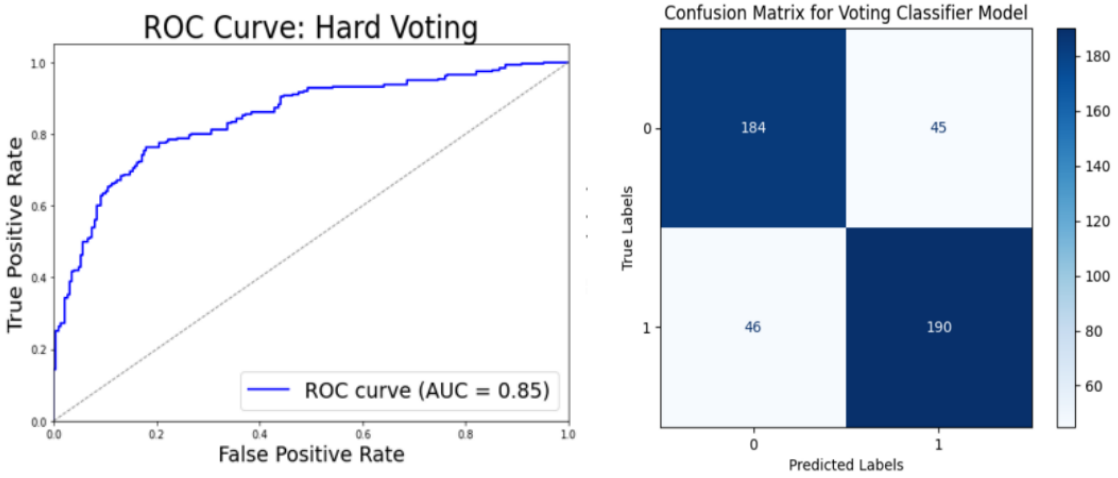


Figure 12. ROC Curve and Confusion Matrix of Voting Classifier.

Table 5. Performance metrics of voting classifier (VC) with different data balancing methods.

Method	Data Balance Method	Accuracy	Precision	Recall	F1-score
Voting Classifier	RUS	0.762	0.770	0.803	0.784
	ROS	0.804	0.809	0.805	0.807
	SMOTE	0.781	0.756	0.720	0.749

Table 5 summarizes the results obtained using three different data balancing methods such as Random Under-Sampling (RUS), Random Over-Sampling (ROS), and Synthetic Minority Over-sampling Technique (SMOTE). The proposed ROS method yielded better performance than the others.

Table 6. Comparison of voting classifier performance metrics with and without PCA.

Method	Dimension Reduction	Accuracy	Precision	Recall	F1-score
Voting Classifier	With PCA (n=7)	0.745	0.731	0.724	0.738
	Without PCA	0.804	0.809	0.805	0.807

Table 6 presents the performance metrics of a Voting Classifier model with and without applying Principal Component Analysis (PCA) as a dimensionality reduction technique. The results indicate that implementing PCA degraded the Voting Classifier's performance, yielding lesser scores in all metrics. Excluding some information could explain this decrease in performance. Whether or not the projected increase in computational speed and need for fewer feature acquisition compensates for a loss

in accuracy using PCA would need to be evaluated further.

Figure 13 shows the feature contributions for the proposed voting classifier (VC), illustrating the factors that influence the predictions for the "Not Detected" and "Positive" classes. The features are organized vertically, with blue bars on the left representing features associated with the "Not Detected" outcome and orange bars on the right indicating features contributing to the "Positive" outcome. The length of each bar reflects the strength of the feature's contribution to the classification. This LIME analysis figure provided insight into how different health metrics and demographic factors impact the model's predictions, which could be useful for the decision making process.

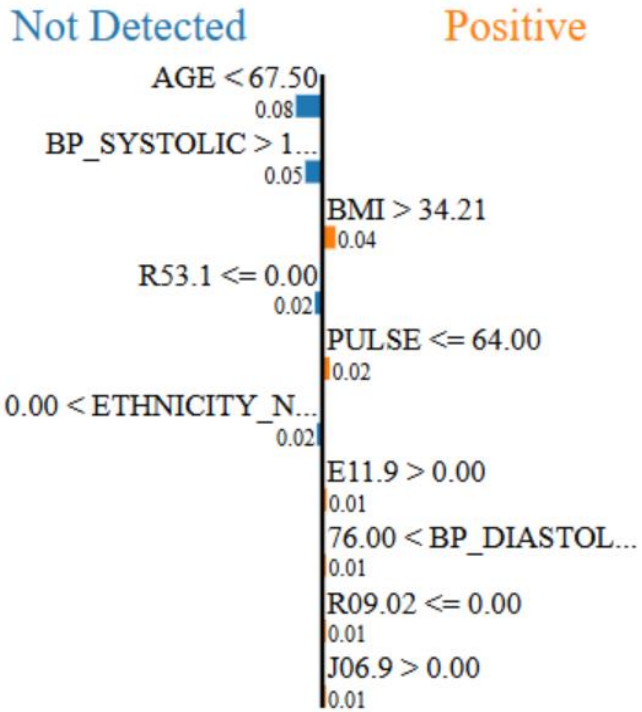


Figure 13. Feature contribution analysis for voting classifier (VC).

5 Discussion

The application of machine learning models in COVID-19 diagnostics shows considerable promise in enhancing both diagnostic accuracy and efficiency. This study employed five different supervised machine learning models: Support Vector Machine (SVM), Classification and Regression Tree (CART), Random Forest (RF), Artificial Neural Network (ANN), and a Voting Classifier. The results indicated that while individual models such as SVM and ANN showed acceptable performance, the Voting Classifier, which integrates the strengths of SVM, RF, and ANN, yielded the highest accuracy of 80.4% and an AUC of 85%. This ensemble approach capitalizes on the diverse strengths of multiple models, reducing the risk of overfitting and enhancing generalization. The Artificial Neural Network (ANN) achieved an accuracy of 74%, and an AUC of 0.73 The ANNs possibly could capture more complex patterns from the data if the variation of the features in the dataset were to be larger. Although we found reasonable performance from almost all the classification evaluation metrices, there is still concern for erroneous

predictions. This is because a model's prediction error is compounded, the larger the input sensitivity value. In our COVID-19 analysis, we assumed that the data collection for 'PCR' was 100% sensitive and 100% specific. However, this is not so in the real-life clinical setting (13, 14). Therefore, if we consider the input data presents with a sensitivity (∂) and machine learning model's sensitivity is (β), the actual compounded sensitivity would be $\partial \times \beta$, which would be different from whatever we compute in the conventional way. In future work, we will incorporate the input data sensitivity during the data collection into the model. Future work should focus on mitigating the compounded sensitivity challenge, expanding the dataset, and incorporating additional features to further enhance model performance. Additionally, exploring other ensemble techniques and advanced machine learning algorithms could provide even greater improvements in diagnostic accuracy and reliability. The continued evolution and refinement of these models holds promise for revolutionizing COVID-19 diagnostics and potentially other areas of medical diagnostics – including other viruses - ultimately contributing to better healthcare outcomes.

6 Conclusion

This study demonstrated the significant potential of machine learning models in enhancing the efficiency and accuracy of COVID-19 diagnosis. By employing machine learning models—Support Vector Machine (SVM), Classification and Regression Tree (CART), Random Forest (RF), Artificial Neural Network (ANN), and a Voting Classifier—the research found that these algorithms can return acceptable sensitivity and specificity. Among the models, the Voting Classifier, which amalgamates the strengths of SVM, RF, and ANN, achieved the highest performance metrics, including an accuracy of 80.4% and an AUC of 85%. These findings highlight the effectiveness of ensemble methods in mitigating individual model limitations and achieving robust diagnostic performance.

7 References

1. Jebril, N. (2020). World Health Organization declared a pandemic public health menace: a systematic review of the coronavirus disease 2019 "COVID-19". https://dx.doi.org/10.2139/ssrn.3566298

2. Viloria, A., Lezama, O. B. P., Mercado-Caruzo, N. (2020). Unbalanced data processing using oversampling: machine learning. Procedia Computer Science, 175, 108-113. https://doi.org/10.1016/j.procs.2020.07.018

3. Suthaharan, S., Suthaharan, S. (2016). Support vector machine. Machine learning models and algorithms for big data classification: thinking with examples for effective learning, 207-235. https://doi.org/10.1007/978-1-4899-7641-3_9

4. Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., et al. (2013). API design for machine learning software: experiences from the scikit-learn project. https://doi.org/10.48550/arXiv.1309.0238

5. Buntine, W. (2020). Learning classification trees. In Artificial Intelligence frontiers in statistics (pp. 182-201). Chapman and Hall/CRC. https://doi.org/10.1201/9781003059875

6. Rigatti, S. J. (2017). Random forest. Journal of Insurance Medicine, 47(1), 31-39. https://doi.org/10.17849/insm-47-01-31-39.1

7. Grossi, E., Buscema, M. (2007). Introduction to artificial neural networks. European journal of gastroenterology & hepatology, 19(12), 1046-1054. https://doi.org//10.1097/MEG.0b013e3282f198a0

8. Sharma, S., Sharma, S., Athaiya, A. (2017). Activation functions in neural networks. Towards Data Sci, 6(12), 310-316. https://www.ijeast.com/papers/310-16,Tesma412,IJEAST.pdf

9. Shiplu, A. I., Rahman, M. M., Watanobe, Y. (2023). A Robust Ensemble Machine Learning Model with Advanced Voting Techniques for Comment Classification. In International Conference on Big Data Analytics (pp. 141-159). Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-58502-9_10

10. Ruta, D., Gabrys, B. (2005). Classifier selection for majority voting. Information fusion, 6(1), 63-81. https://doi.org/10.1016/j.inffus.2004.04.008

11. Rainio, O., Teuho, J., Klén, R. (2024). Evaluation metrics and statistical tests for machine learning. Scientific Reports, 14(1), 6086. https://www.nature.com/articles/s41598-024-56706-x

12. Townsend, J. T. (1971). Theoretical analysis of an alphabetic confusion matrix. Perception & Psychophysics, 9, 40-50. https://doi.org/10.3758/BF03213026

13. Kortela, E., Kirjavainen, V., Ahava, M. J., Jokiranta, S. T., But, A., Lindahl, A., et al. (2021). Real-life clinical sensitivity of SARS-CoV-2 RT-PCR test in symptomatic patients. PloS one, 16(5), e0251661. https://doi.org/10.1371/journal.pone.0251661

14. https://www.cap.org/member-resources/articles/how-good-are-covid-19-sars-cov-2-diagnostic-pcr-tests#:~:text=The%20analytic%20performance%20of%20PCR,specificity%20is%20near%20100%25%20also