Comprehensive Evaluation of Machine Learning Techniques for Obstructive Sleep Apnea Detection

Alaa Sheta[1], Walaa H. Elashmawi[2], Adel Djellal[3], Malik Braik[4],
Salim Surani[5], Sultan Aljahdali[6], Shyam Subramanian[7], Parth S. Patel[8]
Department of Computer Science, Southern Connecticut State University, New Haven, CT, USA[1]
Department of Computer Science, Faculty of Computers and Informatics, Suez Canal University, Ismailia, Egypt[2]
EEA Department, National Higher School of Technology and Engineering, Annaba, Algeria[3]
Department of Computer Science, Al-Balqa Applied University, Salt, Jordan[4]
Department of Pharmacy and Medicine, College Station, Texas A&M University, Texas, USA[5]
Computer Science Department, Taif University, Taif, Saudi Arabia[6]
Chief, Pulmonary, Critical Care and Sleep Medicine, Sutter Health, Sacramento, California, USA[7]
Department of Internal Medicine, Mayo Clinic, Rochester, MN, USA[8]

***Abstract*—Obstructive Sleep Apnea (OSA) is a prevalent health issue affecting 10-25% of adults in the United States (US) and is associated with significant economic consequences. Machine learning methods have shown promise in improving the efficiency and accessibility of OSA diagnoses, thus reducing the need for expensive and challenging tests. A comparative analysis of Logistic Regression (LR), Support Vector Machine (SVM), Gradient Boosting (GB), Gaussian Naive Bayes (GNB), Random Forest (RF), and K-Nearest Neighbors (KNN) algorithms was conducted to predict Obstructive Sleep Apnea (OSA). To improve the predictive accuracy of these models, Random Oversampling was applied to address the imbalance in the dataset, ensuring a more equitable representation of the minority class. Patient demographics, including age, sex, height, weight, BMI, neck circumference, and gender, were employed as predictive features in the models. The RFC provided outstanding training and testing accuracies of 87% and 65%, respectively, and a Receiver Operating Characteristic (ROC) score of 87%. The GBC and SVM classifiers also demonstrated good performance on the test dataset. The results of this study show that machine learning techniques may be effectively used to diagnose OSA, with the Random Forest Classifier demonstrating the best results.**

***Keywords*—Machine learning; obstructive sleep apnea; random forest classifier; oversampling; classification**

I. Introduction

Obstructive sleep apnea (OSA) is a prevalent disorder affecting a substantial portion of the population. Characterized by recurrent obstructions of the upper airway during sleep, it results in intermittent cessation of airflow [1], [2]. A multitude of factors have been identified as risk determinants for OSA, including obesity, male gender, smoking, age, craniofacial anomalies, and menopause in women [3]. Symptoms that suggest OSA include chronic snoring, observed apneic episodes, gasping during sleep, frequent awakenings, non-restorative sleep, increased nighttime urination, and excessive daytime sleepiness. Timely diagnosis of OSA is crucial as untreated OSA can contribute to the development of cardiovascular diseases, metabolic disorders, and neurocognitive impairments [4].

The standard diagnostic method for OSA is overnight polysomnography (PSG), however, it is expensive and often limited in accessibility. Therefore, it is important to prioritize high-risk individuals for PSG, particularly those with moderate to severe OSA, to optimize the utilization of sleep laboratories [5]. The severity of OSA is commonly assessed using the Apnea-Hypopnea Index (AHI), with cutoff values of $6-15/hour$ indicating mild OSA, $16-30/hour$ indicating moderate to severe OSA, and values exceeding $30/hour$ indicating severe OSA [5].

In recent years, machine learning has drawn considerable interest as a potentially effective way to address complex problems in various sectors, most notably healthcare. Its strengths, such as robustness, self-organization, adaptive learning, and parallel processing, make it an attractive tool.

Machine learning algorithms, renowned for their ability to discern patterns within intricate datasets, have garnered significant attention. Prominent examples of such algorithms encompass Support Vector Machines (SVM) [6], [7], Gradient Boosting Classifiers (GBC) [8], Gaussian Naive Bayes (GNB) [9], Random Forest Classifiers (RFC) [10], [11], and K-Nearest Neighbors Classifiers (KNC) [12]. Consequently, machine learning models have seen increasing use in medical healthcare, including the prediction of OSA, and have shown promising outcomes.

This study conducts a comparative evaluation of traditional regression modeling and a suite of machine learning algorithms for predicting obstructive sleep apnea (OSA) based on physical parameters. The succeeding sections of this paper are structured as follows. Section II presents a comprehensive review of existing machine learning approaches applied to OSA prediction. Section III provides a detailed characterization of the dataset employed in this study. In Sections IV and V, the training and evaluation methodology for ML models, including particular classification algorithms, is described. Finally, in Section VI, we present the experimental results and performance analysis concerning various evaluation metrics and report the results obtained through this research endeavor.

II. Related Works

Machine learning has emerged as a preeminent paradigm for classifying medical data owing to its ability to manage

and extract insights from voluminous and intricate datasets effectively. For instance, in the context of OSA diagnosis, traditional methods like drug-induced sleep endoscopy (DISE) rely on subjective observer evaluations, as seen in the VOTE classification system proposed by Altintas et al. [13], which showed only moderate to fair agreement among observers. This highlights the potential of machine learning models to provide objective, consistent, and accurate diagnoses by automating the analysis of complex data, thereby addressing the limitations of observer-dependent methods.

Various studies [14]–[16] have utilized machine learning algorithms to improve diagnostic precision and patient outcomes in various diseases. The authors in [17], highlighted the utility of support vector machines (SVM) in classifying brain MRI images for neurological disorders, which aligns with its potential for OSA diagnosis. However, the authors in [18] explored the use of various ML methods to predict obstructive sleep apnea syndrome (OSAS) severity using demographic, clinical, and spirometric data from 313 patients. Their study demonstrated that SVMs and Random Forests (RFs) showed the best classification performance.

Furthermore, ensemble learning methods, such as Gradient Boosting Classifiers (GBCs) and RF classifiers, have demonstrated robust performance in medical diagnostics due to their ability to handle imbalanced and high-dimensional data. Ramesh et al. [8] applied GBCs to classify OSA from electronic health records, achieving an accuracy of 68.06%. The authors in [11] further validated the performance of RFCs for OSA detection, leveraging feature selection algorithms to improve model interpretability and accuracy.

Integrating the STOP-BANG questionnaire with machine learning models is employed in [12] to enhance OSA diagnosis. Among the four algorithms tested, the K-Nearest Neighbor (K-NN) model demonstrated the best performance, achieving 94% accuracy. The results highlight the potential of combining ML with traditional tools to improve the reliability and efficiency of OSA screening. However, in [19], the SLEEPS model, a machine learning-based questionnaire using nine items, accurately predicts OSA, COMISA, and insomnia without polysomnography. The model trained on over 4,600 participants using XGBoost, achieved AUROC values above 0.89, outperforming tools like STOP-BANG.

In recent years, the authors in [20] have used the Swedish National Study on Aging and Care electronic health data to predict sleep apnea with a ML model. The XGBoost and Bidirectional Long Short-Term Memory Networks modules give the model 97% accuracy with 75 features and 10,765 samples. Furthermore, pre-screening symptoms were employed to diagnose OSA [21], and the experimental findings revealed that the Decision Tree Classifier (DTC) and RF outperformed other comparable algorithms, achieving the highest classification accuracies. Similarly to the authors in [22], the RF classifier technique is utilized to predict sleep disorders and has achieved the highest accuracy. The potential of ML models for cost-effective OSA screening, with RF and LightGBM showing the most promise for clinical use, is discussed in [23].

A concise overview of the utilization of machine learning techniques in diagnosing, classifying, and treating sleep-related respiratory problems is presented in [24]. The effectiveness of machine learning-based classifiers in OSA classification is highly affected by the quality and quantity of input data and the selection of the machine learning approach.

More advanced techniques, like deep learning models, have demonstrated superior accuracy in sleep apnea and related disorder detection. Studies like [10] combined ECG signals with machine learning and deep learning to achieve 86.25% accuracy, while advanced architectures like multi-resolution residual network (MR-ResNet) [6] and CNN-based approaches [9] reached accuracies of 90.8% and 79.61%, respectively, leveraging polysomnographic (PSG) data. Wearable systems [25] and single-lead electrocardiogram (ECG) classifiers [7] achieved notable performance with accuracies up to 88.2% and 93.0%. Other studies, such as [26], employed EEG and/or electrooculogram (EOG) signals for sleep staging, achieving up to 84.5% accuracy, while [27] explored microelectromechanical system (MEMS)-based solutions. However, the computational demands and reliance on specialized data limit the practicality of these methods in resource-constrained environments.

Despite substantial progress in sleep apnea diagnosis, pursuing more accurate, efficient, and accessible methods remains an active area of research. Many existing studies rely on costly diagnostic tools like PSG, need help with imbalanced datasets, and focus on computationally intensive deep-learning models that lack practicality and generalizability. This study addresses these challenges by using easily obtainable physical parameters, applying oversampling to balance data, and evaluating resource-efficient algorithms, offering scalable and accessible solutions for diverse populations.

III. OSA Data Collection

Data were collected from adult individuals who were referred to a community sleep center due to suspected obstructive sleep apnea (OSA) and had not received a previous diagnosis. Each participant provided demographic details such as age, gender, and ethnicity, and completed sleep-related questionnaires, including the Epworth Sleepiness Scale. Prior to undergoing polysomnography, a physical examination was conducted, which included an assessment of the airway using a modified Friedman grade, measurement of body mass index (BMI), and neck circumference (NC). OSA was diagnosed when the apnea-hypopnea index (AHI) was equal to or greater than 15. Incomplete questionnaires or polysomnography records with inadequate technical quality or insufficient total sleep time were excluded from the analysis.

Retrospective data analysis and review were conducted after the administrative approval of the data from the Torr Sleep Center executive and institutional committee. The study was conducted at Torr Sleep Center in their Corpus Christi, Texas Location. The patients undergoing the second or split night (titration night) were excluded. The patients have undergone the first (diagnostic night were determined by the computerized search using the CPT code 98510. The patient discussed the study with the registered polysomnographic therapist during the presentation. Baseline demographic information was collected. Height, weight, Modified Friedman, Waist circumference, and diameter were assessed, and Body Mass Index (BMI) was computed. The patient underwent overnight polysomnography. The nocturnal polysomnogram (NPSG) included the

recording of electroencephalogram (EEG): F4-M1, F3-M2, C4-M1, C3-M2, O2-M1, O1-M2, electrooculogram (EOG), submental, intercostal and anterior tibialis electromyogram (EMG), electrocardiogram (EKG), airflow by nasal pressure, and oral thermistor, abdominal and chest wall excursion using impedance plethysmography and oxygen saturation by pulse oximetry attended by a sleep technologist. The sleep study was staged and scored using AASM standards.

All methods described in this study adhere to relevant guidelines and regulations. The research protocol obtained ethical approval from the Research Ethics Committee of NTUH. (protocol number 201603113RIND), and the committee waived the requirement for participant consent. Table I presents dataset statistics, including sample size, mean values, and ranges. Additionally, Table II displays randomly selected data samples to provide insight into the dataset's structure.

Fig. 1 depicts the distribution of the dataset across OSA, Sex, and BMI attributes. These attributes play a crucial role in the modeling process. Notably, Body Mass Index (BMI) is one of these attributes. BMI is a measure to estimate body fat using an individual's height and weight. It is a standard adult screening tool for weight-related health problems. According to the classifications given in Table III, it sorts people into four groups: underweight, normal weight, overweight, or obese.

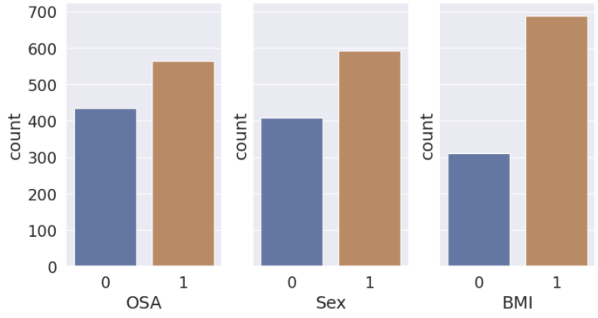


Fig. 1. Distribution of some attributes of the data: OSA, Sex, and BMI.

Fig. 2 shows the dataset box plot, and Fig. 3 shows a correlation between dataset parameters; it can be seen that the most correlated parameter with OSA is Neck circumference and then weight, which means that these two parameters will play a significant role in classification results.

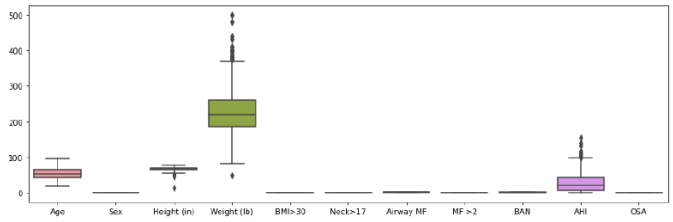


Fig. 2. Dataset Box plot.

A. Oversampling

The OSA data set is imbalanced; we adopted a Random oversampling technique to balance the data. RandomOver

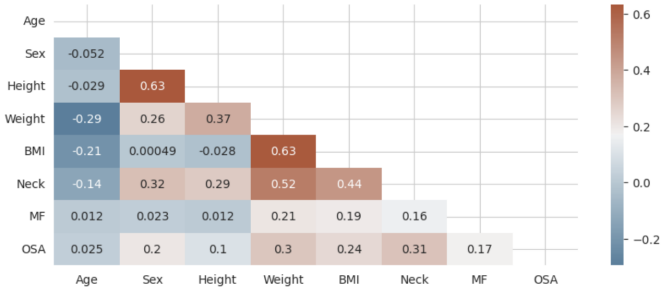


Fig. 3. Correlation heatmap.

Sampler is a machine learning technique used to handle imbalanced datasets. It addresses class imbalance by creating a more equally distributed dataset by randomly oversampling the minority class as follows:

- First, the minority class is identified in the dataset. This is the class with fewer instances than the majority class.
- Then, the "RandomOverSampler" algorithm randomly selects instances from the minority class. It oversamples the minority class by duplicating its instances to match the population of the majority class.
- After the oversampling, the resulting dataset is balanced or nearly balanced, with an equal number of instances for each class.
- Finally, a machine learning model is trained using the balanced dataset; the model should outperform its counterpart trained on an unbalanced dataset.

Oversampling techniques, such as Random Oversampling and Synthetic Minority Oversampling Technique (SMOTE), have been widely used to address this issue. The authors in [24] employed SMOTE to enhance the performance of ML models in detecting OSA, demonstrating its effectiveness in reducing classification errors for minority classes. It's worth noting that the Oversampling can be prone to overfitting, especially if the minority class is oversampled too much. Tuning the oversampling ratio is essential to finding the optimal balance between reducing class imbalance and avoiding overfitting.

B. Data Scaling

Data scaling is an essential pre-processing stage for the machine learning classification process. The purpose of scaling the data in this way is to make it easier to compare features with different units and scales. This can potentially enhance the efficiency of machine learning algorithms, especially those like k-nearest neighbors and support vector machines that are sensitive to input data size.

The "StandardScaler" is a preprocessing technique used in machine learning to scale numerical data. Standardization is achieved by transforming the data to exhibit a zero mean and unit variance. Specifically, it can be done according to the following steps.

- First, the mean of each feature (column) in the data is calculated.

TABLE I. STATISTICS OF THE OSA DATA SET

	Age	Sex	Height (in)	Weight (lb)	BMI>30	Neck>17	Airway MF
count	1000	1000	1000	1000	1000	1000	1000
mean	54.031	0.592	67.037	227.653	0.689	0.661	2.722
std	14.320	0.492	4.624	58.218	0.463	0.474	1.008
min	19.000	0.00	15.000	49.000	0.000	0.000	0.000
25%	44.000	0.000	64.000	186.750	0.000	0.000	2.000
50%	54.000	1.000	67.000	220.000	1.000	1.000	3.000
75%	65.000	1.000	70.000	262.000	1.000	1.000	4.000
max	96.000	1.000	79.000	500.000	1.000	1.000	4.000

TABLE II. SAMPLE OF THE OSA DATA SET

Sample No.	Age	Sex	Height (in)	Weight (lb)	BMI>30	Neck>17	Airway MF
652	45	1	73.0	284	1	1	1
939	71	0	59.0	192	1	0	0
319	51	1	77.0	280	1	1	1
626	41	1	66.0	180	0	1	0
808	85	0	59.0	178	1	1	1

TABLE III. BMI CATEGORIES

BMI Category	BMI Range
$Under\_weight$	< 18.5
$Normal$	$18.5 - 24.9$
$Over\_weight$	$25 - 29.9$
$Obese$	≥ 30

- Then, the StandardScaler subtracts the mean from each value in the feature. This centers the data around zero.
- Next, the StandardScaler divides each value in the feature by its standard deviation. This scales the data to have a standard deviation of 1.
- After the scaling is done, the resulting dataset has a mean value of zero and a standard deviation value of one.

IV. PROPOSED METHODOLOGY

The machine learning process consists of multiple steps, beginning with data collection. The collected data might be raw and unstructured, so the subsequent step involves pre-processing. In this step, missing data is eliminated and thoroughly cleaned to ensure its suitability for analysis.

Following the pre-processing stage, the next step is feature extraction, which involves identifying and extracting pertinent features from the data. This step is significant since the ML model's performance relies heavily on the quality of the extracted features. After extracting the features, the dataset was divided into several subsets for training and testing. The training subset was used to develop the model, while the testing subset was used to evaluate how well the model can be applied to new data. The subsequent step is classification, where the machine learning algorithm utilizes the extracted features to classify new data into distinct categories. Various classification algorithms, such as RF, SVM, or ANN, can be employed to train the model on the training set.

Finally, precision, recall, and F1-score metrics evaluate the model's performance. The model exhibiting the highest performance is chosen and employed for classifying new data. It should be emphasized that this process is iterative and typically requires multiple iterations of adjustments and enhancements to attain optimal performance. Algorithm 1 summarizes all the process stages of machine learning. Fig. 4 shows the proposed methodology utilized for OSA detection using various machine-learning techniques.

Algorithm 1: Machine Learning Process

```
Input: Raw data
Output: Trained model
Step 1: Data Collection
    Collect raw data from various sources
Step 2: Data Pre-processing
    Remove missing data and outliers
    Normalize or scale the data if necessary
Step 3: Feature Extraction
    Extract relevant features from the pre-processed data
    Reduce the dimensionality of the data if necessary
Step 4: Model Selection
    Choose appropriate machine learning algorithms
    Select hyperparameters for the algorithms
Step 5: Model Implementation
    Train the selected models on the pre-processed data
    Evaluate the performance of the trained models
Step 6: Model Evaluation
    Test the trained models on new data
    Evaluate the performance of the tested models
Step 7: Model Deployment
    Deploy the best-performing model in production
```

V. ML METHODS

A. Logistic Regression (LR)

Logistic regression is a statistical technique that shares similarities with linear regression and is utilized for predicting binary outcomes. Unlike the Mantel-Haenszel odds ratio, which is limited to discrete explanatory variables, logistic regression can simultaneously handle continuous and

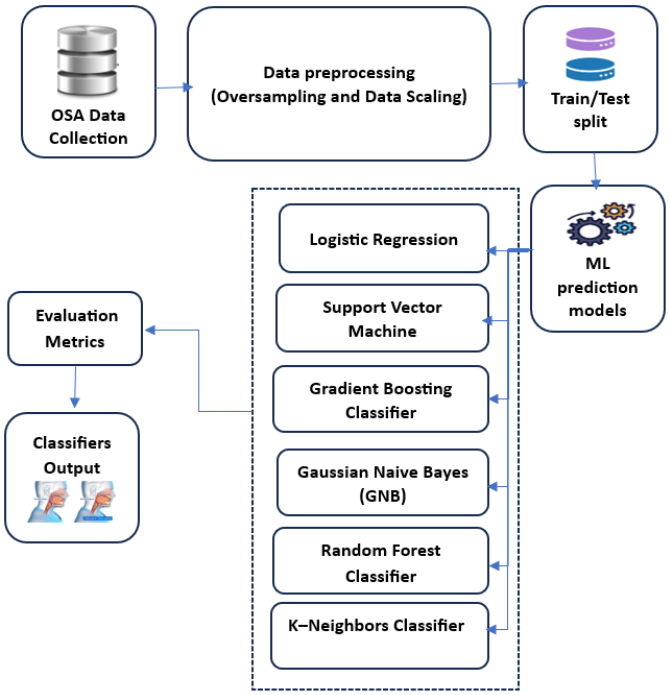


Fig. 4. The proposed methodology for utilizing various ML models to detect OSA.

multiple explanatory variables. This capability is essential when studying the impact of various factors on the response variable. Logistic regression models the probability of an outcome by considering the covariance among variables and accounting for individual characteristics. This approach helps address confounding effects when analyzing multiple variables independently. The logarithm of the odds is used in modeling, as odds represent a ratio, as explained by Sperandei [28].

$$log\left(\frac{\pi}{1-\pi}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_m x_m \qquad (1)$$

The symbol π represents the likelihood of an event, such as the incidence of OSA. The regression coefficients β_i correspond to the reference group and the explanatory factors x_i.

B. Support Vector Machines (SVM)

The Support Vector Machine (SVM) is a widely used supervised learning model for prediction and classification tasks. It was developed in 1995 by Vladimir Vapnik and his team at AT&T Bell Laboratory. SVM utilizes a nonlinear mapping function to transform the training data set into a higher dimensional space. It then employs linear regression to separate the data within this transformed space. This approach has demonstrated effectiveness across various applications, enabling the learning of complex decision boundaries and improving classification accuracy. The author in [29] described this process of SVM as approximating the training data set within a higher dimensional space and employing linear regression to separate the data. Fig. 5 illustrates the SVM model.

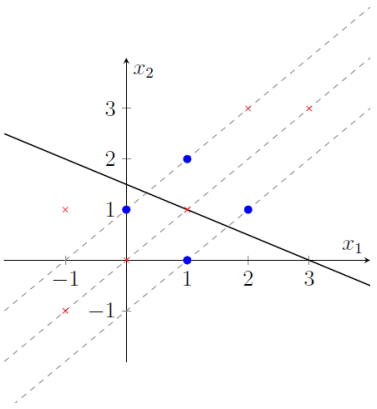


Fig. 5. Optimal hyperplane in support vector machine.

C. Gradient Boosting Classifier (GBC)

As an ensemble machine learning approach, the Gradient Boosting Classifier combines many weak models into one more potent model, increasing the prediction power of the combined model [30]. It operates iteratively by training decision trees on the residuals of the preceding tree, utilizing gradient descent optimization to minimize the loss function. This technique enables the algorithm to learn more intricate decision boundaries, improving prediction accuracy. The specific structure of the algorithm, including its formulas, is highly influenced by the chosen designs of $\Phi\,(y, f)$ and $h\,(x, \theta)$. More detailed examples of these algorithms can be found in the work of Friedman [30].

Algorithm 2: Friedman's Gradient Boost Algorithm

Input: Training Dataset $\mathcal{D} = (\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$, number of iterations M, learning rate α, base model $h_0(\mathbf{x})$, loss function $L(y, F(\mathbf{x}))$

Output: Ensemble model $F(\mathbf{x}) = \sum m = 1^M \beta_m h_m(\mathbf{x})$

Initialize ensemble model $F_0(\mathbf{x}) = h_0(\mathbf{x})$;

for $m \in 1, \ldots, M$ **do**

  Calculate the negative gradient $r_{im} = -\left[\frac{\partial L(y_i, F_{m-1}(\mathbf{x}i))}{\partial Fm-1(\mathbf{x}i)}\right] i = 1^n$ for each training instance $\mathbf{x}i$

  Fit a base model $h_m(\mathbf{x})$ to the negative gradient r_{im}

  Compute the optimal step size $\beta_m = \arg\min_\beta \sum_{i=1}^n L(y_i, F_{m-1}(\mathbf{x}_i) + \beta h_m(\mathbf{x}i))$

  Update the ensemble model: $F_m(\mathbf{x}) = Fm - 1(\mathbf{x}) + \alpha\beta_m h_m(\mathbf{x})$;

end

return *Ensemble model* $F(\mathbf{x}) = \sum_{m=1}^M \beta_m h_m(\mathbf{x})$

The Gradient Boosting is used for creating an ensemble model on a training set $\mathcal{D}$ consisting of n instances, where each instance has a pair of features $\mathbf{x}i$ and label y_i. It requires several iterations M, a learning rate α, a base model $h_0(\mathbf{x})$, a loss function $L(y, F(\mathbf{x}))$ to evaluate the quality of the ensemble model, and a set of hyper-parameters for the base model. The algorithm initializes the ensemble model to $F_0(\mathbf{x}) = h_0(\mathbf{x})$, and then iteratively improves it by fitting

a base model $h_m(\mathbf{x})$ to the negative gradient of the loss function concerning the current ensemble model $Fm-1(\mathbf{x})$. The optimal step size β_m is computed using line search, and the ensemble model is updated by adding a scaled version of the new base model $h_m(\mathbf{x})$ to the previous ensemble model $F_{m-1}(\mathbf{x})$. The final output of the algorithm is the resulting ensemble model $F(\mathbf{x})$.

D. Gaussian Naive Bayes (GNB)

One machine learning algorithm based on the concepts of Bayes' Theorem is the Naive Bayes Classifier [31]. These classifiers rely on the assumption of strong independence among the features used for predictions.

Under this premise, it is assumed that the value of one characteristic does not affect the value of any other feature. A notable advantage of Naive Bayes Classifiers is their ability to be efficiently trained in supervised learning scenarios, even when working with limited training data. Moreover, their straightforward design and ease of implementation make them popular for various real-world applications.

In machine learning, continuous data is frequently assumed to adhere to a normal (Gaussian) distribution, mainly when dealing with classification tasks. This assumption suggests that the continuous values corresponding to each class follow a normal distribution. By making this assumption, it becomes possible to estimate the likelihood of the features using the Gaussian probability density function:

$$P\left(x_i|y\right) = \frac{1}{\sqrt{2\pi\sigma_y^2}} exp\left(-\frac{\left(x_i-\mu_y\right)^2}{2\sigma_y^2}\right) \tag{2}$$

One strategy for constructing a simple model is to assume that a Gaussian distribution with no covariance among dimensions can characterize the data. In other words, each dimension is considered independent of the others. This type of model can be easily built by calculating the mean and standard deviation of the data points within each label, as these parameters define the distribution.

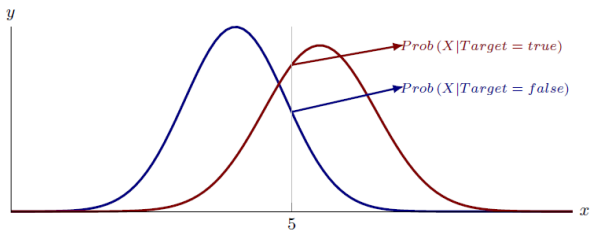


Fig. 6. Demonstration of working with a continuous variable in naïve bayes.

The provided illustration, as shown in Fig. 6 demonstrates the functioning of a Gaussian Naive Bayes (GNB) classifier. For each data point, the classifier calculates the z-score distance, which is the difference between the data point and the mean of each class divided by the standard deviation of that class.

E. Random Forest Classifier (RFC)

The term "random forest" refers to an ensemble of tree predictors. A randomly distributed vector, which is sampled individually and distributed uniformly among all trees in the forest, is relied upon by each individual tree [32]. As the random forest expands in terms of the number of trees, its generalization error stabilizes. The predictive capacity of individual trees and their interrelationships impact the ultimate level of this error.

Random forests use a random feature selection method for splitting nodes, which leads to error rates comparable to Adaboost while being more resilient to noise. The forest's internal estimates monitor various factors such as error, strength, and correlation and can assess the impact of increasing the number of features used for splitting. These estimates are also useful in determining the importance of different variables, and the approach is applicable to regression tasks as well.

RFC constructs decision trees by randomly selecting subsets of the training data and features for each node [33], [34]. Finding the optimal feature to divide the data at each node is how the trees are iteratively developed until a stopping requirement is satisfied [35]–[37]. During prediction, the ensemble of trees votes on the class label for a new input instance, with the class receiving the most votes being predicted as the output. This approach mitigates overfitting and enhances classification accuracy by leveraging the collective ability of the tree ensemble to capture diverse patterns and relationships within the data. RFC has demonstrated successful applications in various domains, including sleep apnea research [11], [38]. Refer to Fig. 7 for an illustrative example of the RFC mechanism.

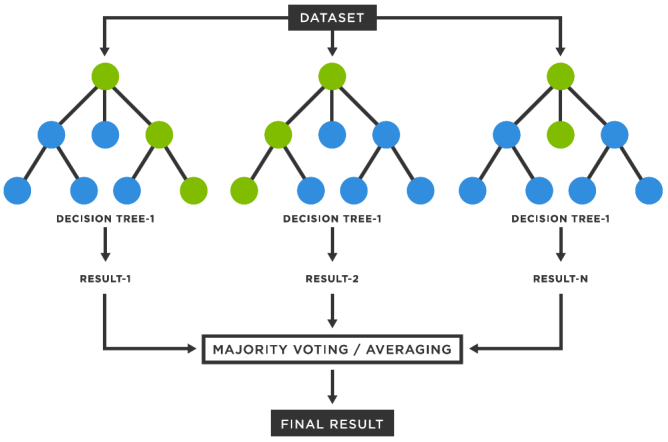


Fig. 7. Random forest mechanism.

To train individual trees, the Random Forest Algorithm requires a training set $\mathcal{D}$, a certain number of trees to be included in the forest (T), a random number of features (K) to be chosen for each tree, and a decision tree algorithm $\mathcal{A}$. To build the forest, the algorithm iteratively constructs T trees. At each iteration t, the algorithm randomly selects K features from the available features and draws a bootstrap sample $\mathcal{D}_t$ from the training set $\mathcal{D}$. A decision tree f_t is then trained on the sampled features and the bootstrap sample $\mathcal{D}_t$, using

Algorithm 3: Random Forest Classifier Algorithm

Input: Training Dataset $\mathcal{D}$, number of trees T, number of features K, decision tree algorithm $\mathcal{A}$
Output: Random Forest $\mathcal{F}$
for $t \in 1, \ldots, T$ **do**
  Sample K features from the p available features without replacement
  Draw a bootstrap sample $\mathcal{D}_t$ from $\mathcal{D}$
  Train a decision tree f_t on $\mathcal{D}_t$ using the selected features and $\mathcal{A}$
end
return *Random Forest* $\mathcal{F} = f_1, \ldots, f_T$

the given decision tree algorithm $\mathcal{A}$. The final output of the algorithm is the resulting random forest $\mathcal{F}$, which consists of the T decision trees $f_1, \ldots, f_T$ (see Algorithm 3).

F. K–Neighbors Classifier (KNN)

One supervised machine learning technique that is commonly employed for classification problems is the K-Nearest Neighbors (KNN) classifier. It sorts unlabeled data points according to the similarity principle, which states that it should consider the class of nearby data points in the training dataset. The number of neighbors to consider is represented by the "K" in KNN.

The algorithm computes the Euclidean distance between the unclassified data instance and each labeled training instance to inform the classification decision. After that, it uses the distances to choose the K closest neighbors. The unlabeled data point's class identification is decided by a majority vote among its K nearest neighbors. KNN is a simple and intuitive algorithm that does not require training. It uses the entire training dataset for classification. The KNN algorithm is easy to understand, and its pseudocode is provided in Algorithm 4.

In this algorithm, the input is a training set $\mathcal{D}$ consisting of labeled instances, a test instance x that we want to classify, and the number of neighbors K to consider. The output is the predicted class label for the test instance.

Algorithm 4: K-Nearest Neighbors Classifier Algorithm

Input: Training Dataset $\mathcal{D} = (\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$, new instance $\mathbf{x}$, number of neighbors K
Output: Predicted label $\hat{y}$ for $\mathbf{x}$
for $i \in 1, \ldots, n$ **do**
  Compute the Euclidean distance $d(\mathbf{x}, \mathbf{x}_i)$ between $\mathbf{x}$ and each training instance $\mathbf{x}_i$;
end
Identify the K training instances with the smallest distances to $\mathbf{x}$; Assign the majority class label among these K instances as the predicted label $\hat{y}$ for $\mathbf{x}$;
return *Predicted label* $\hat{y}$

Each instance in the training set $\mathcal{D}$ has a pair of features $\mathbf{x}_i$ and a label y_i, and the K-Nearest Neighbors Classifier Algorithm is applied to this set. The algorithm calculates the Euclidean distance between each training instance $\mathbf{x}_i$ and $\mathbf{x}$ when given a fresh instance $\mathbf{x}$. Next, it chooses the K training examples that are closest to $\mathbf{x}$, and the projected label $\hat{y}$ for $\mathbf{x}$ is the majority class label among these K examples. The algorithm returns the predicted label $\hat{y}$.

VI. Experimental Results and Performance Analysis

The following section presents the findings from utilizing the proposed machine learning models in the OSA dataset. The performance of each model is evaluated using a comprehensive set of metrics, and a comparative analysis is conducted to identify the most effective approach. These metrics provide quantitative measures of model quality. Most of them mainly depend on calculating TP (i.e. count of model-correct positives), FP (i.e. number of positive cases misclassified as negative), TN (i.e. the number of true negatives the model identified), and FN (i.e. the model predicts a negative result while it is positive). The total number of instances is $Total$ (i.e. $Total = TP + TN + FP + FN$). These metrics include accuracy, precision, recall, and F1-score, as computed using the following formulas.

$$Accuracy = \frac{TP + TN}{Total} \tag{3}$$

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

$$Recall = \frac{TP}{TP + FN} \tag{5}$$

$$F1 = 2 \times \frac{precision \times recall}{precision + recall} \tag{6}$$

Table IV provides a comparative analysis of various classification algorithms (LR, SVM, GBC, GNB, RFC, and KNC) based on their performance metrics: accuracy, precision, recall, and F1-score, calculated for both training and testing datasets. Our findings indicate that:

- The analysis results show that RFC has the highest performance on the training set across all metrics, with a training accuracy of 86.78%, precision of 92.64%, and F1-score of 87.80%. However, its testing performance drops significantly (accuracy of 65.02%).
- The GBC achieves relatively balanced performance between training and testing datasets, with a testing accuracy of 66.08% and the highest test F1-score of 67.79% among the models, indicating better generalization compared to RFC and other models except KNC.
- The SVM shows testing accuracy (64.66%) and moderate performance across metrics. However, its performance on the training dataset is slightly lower than that of RFC, KNC, and GBC.
- KNC demonstrates balanced and consistent performance, achieving a training accuracy of 74.14% and testing accuracy of 67.14%. However, its recall on the test set decreases to 62.59%.

TABLE IV. CALCULATED PERFORMANCE CRITERIA BASED ON VARIOUS ML MODELS

Method	Train Accuracy	Train Precision	Train Recall	Train F1	Test Accuracy	Test Precision	Test Recall	Test F1
LR	0.665880	0.666667	0.698851	0.682379	0.650177	0.596273	0.738462	0.659794
SVM	0.704841	0.706935	0.726437	0.716553	0.646643	0.594937	0.723077	0.652778
GBC	0.822904	0.807775	0.85977	0.832962	0.660777	0.60119	0.776923	0.677852
GNB	0.645809	0.675862	0.649007	0.662162	0.650177	0.707692	0.601307	0.650177
KNC	0.741440	0.763218	0.741071	0.751982	0.671378	0.707692	0.625850	0.664260
RFC	0.867769	0.926437	0.834369	0.877996	0.650177	0.753846	0.593939	0.664407

- LR achieves moderate training accuracy (66.59%) and a balanced F1 score (68.24%). However, in the test, it maintains a consistent performance accuracy of 65.02% but suffers from lower precision (59.63%), indicating a higher false positive rate compared to algorithms such as KNC and GBC.
- The GNB has the lowest accuracy on the training dataset, with a value of 64.58%, but has a high test precision of 70.77%. This highlights the model's ability to correctly classify positive cases, despite lower overall accuracy and recall.

The results align well with existing literature on applying ML models for OSA diagnosis. Consistent with prior studies, the RFC emerged as the best-performing model regarding training accuracy (87%), reflecting its robustness in handling complex datasets and its effectiveness as highlighted in studies like [8], [10]. Similarly, the GBC demonstrated strong generalization capabilities, achieving a balanced performance across metrics, which is in line with findings in [8], where GBC was noted for its ability to capture intricate patterns in data. Overall, the alignment between this study's findings and existing research underscores the validity of these ML models for OSA prediction.

Furthermore, confusion matrices were generated to gain a comprehensive understanding of each machine learning algorithm's predictive capabilities. These visual representations offer a detailed breakdown of correct and incorrect classifications. Fig. 8 through 13 provide a graphical depiction of these training and testing data results, enabling a thorough analysis of each model's performance characteristics.

From Fig. 8, the total number of correctly predicted OSA cases is 564 out of 847 (i.e. in the case of training) and 184 out of 283 (i.e., in the case of testing). However, the percentage of incorrectly classified instances is 33.41% and 34.98% in training and testing, respectively; the LR model shows a moderate performance.

Fig. 9 demonstrates the strong overall performance of the SVM model, with a significantly higher number of correct predictions (diagonal elements) compared to incorrect predictions (off-diagonal elements). The model seems balanced in predicting classes 0 and 1, with a relatively even distribution of correct predictions for each class. The values in the off-diagonal (119 and 131) in training and (36 and 64) in testing indicate relatively low rates of false positives and false negatives, suggesting that the model effectively distinguishes between the two classes.

According to the GBC confusion matrix (as shown in Fig. 10), the GBC model exhibits strong overall performance, with a significantly higher number of correct predictions of OSA instances equal to 697 (i.e. 82.92%) and 187 (i.e. 66.08%) in training and testing, respectively.

The GNB classifier demonstrates solid performance on the training and testing sets, as shown in Fig. 11. The GNB model correctly classified 294 cases as positive OSA and 253 as negative cases in the case of training. In testing, the overall number of correctly classified cases is 184 out of 283 (i.e. 65.02%). While there's a minor decrease in performance on the testing set, the model still maintains a good balance in predicting both classes.

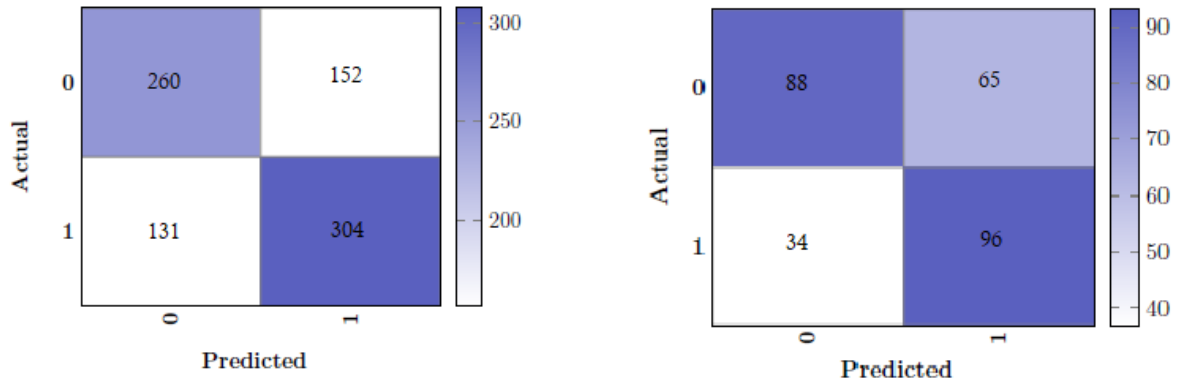
According to the confusion matrix of the random forest classifier (as shown in Fig. 12), the number of correctly classified instances is 735 (TP+TN) in training and 184 (TP+TN) in testing, while the total number of misclassified is 112 (FP+FN) in training and 99 (FP+FN) in testing. However, in Fig. 13, the K–Neighbors classifier, the model Correctly predicted 332 instances as class 1, 296 as class 0, 116 incorrectly predicted as class 1, and 103 incorrectly predicted as class 0 in training. In testing, The model achieves moderate results of 67.14%, indicating that it correctly predicts the class in 67.14% of cases.

Each methodology's efficacy depends on complex factors, including the problem domain, dataset characteristics (size and quality), and computational constraints. Moreover, the receiver operating characteristic (ROC) curve visually shows a binary classifier's performance. The area under the ROC curve (AUC) estimates the general model performance. A higher AUC denotes better discriminative capability; a perfect model achieves an AUC of 1.0, while a random classifier generates an AUC of 0.5. Fig. 14 and 15 present the ROC curves and box plots for the respective classification algorithms.

A. Statistical Test Analysis

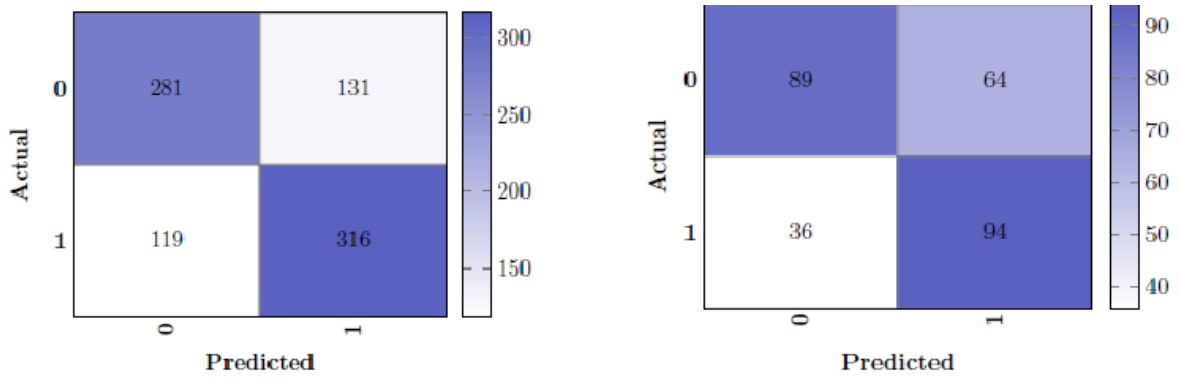
Friedman's statistical test, a non-parametric test technique, was employed to identify the classification technique that outperformed other competing classifiers to conduct a more detailed investigation of the performance of the classification techniques. Table V presents the mean ranks derived from a Friedman test conducted to statistically compare the classification performance of the competing algorithms across accuracy, precision, recall, and F1-measure.

The lowest ranking finding reflects a higher level of performance, as seen in Table V. Friedman's test was used to determine the p-value, displayed in Table V. Some of the p-values found by Friedman's statistical test were less than the significance level, identified as $\alpha = 0.5$. The alternative hypothesis is supported, while the null hypothesis is refuted. The alternative hypothesis contends that there are different



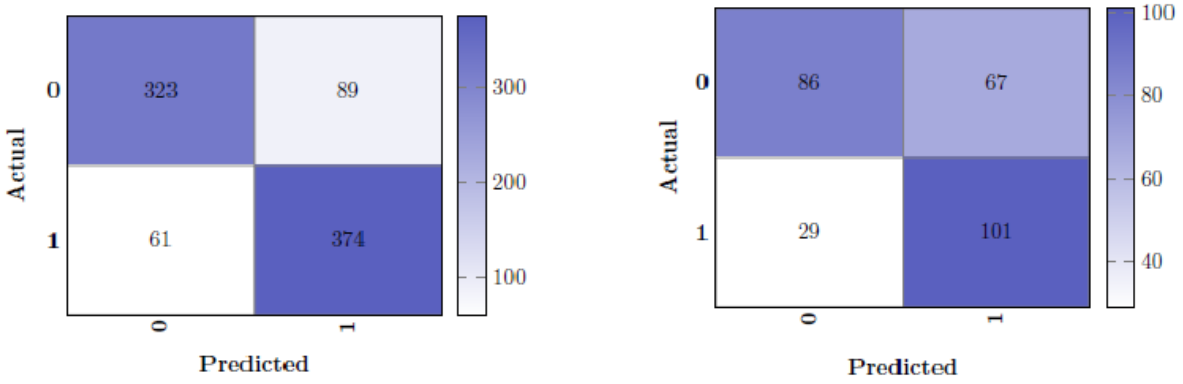
(a) Training. (b) Testing.

Fig. 8. LRC confusion matrices.



(a) Training. (b) Testing.

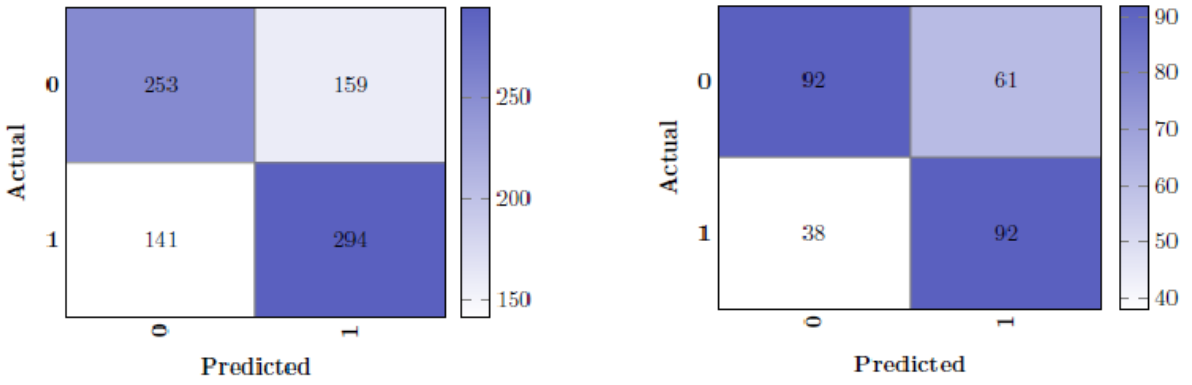
Fig. 9. SVM confusion matrices.



(a) Training. (b) Testing.

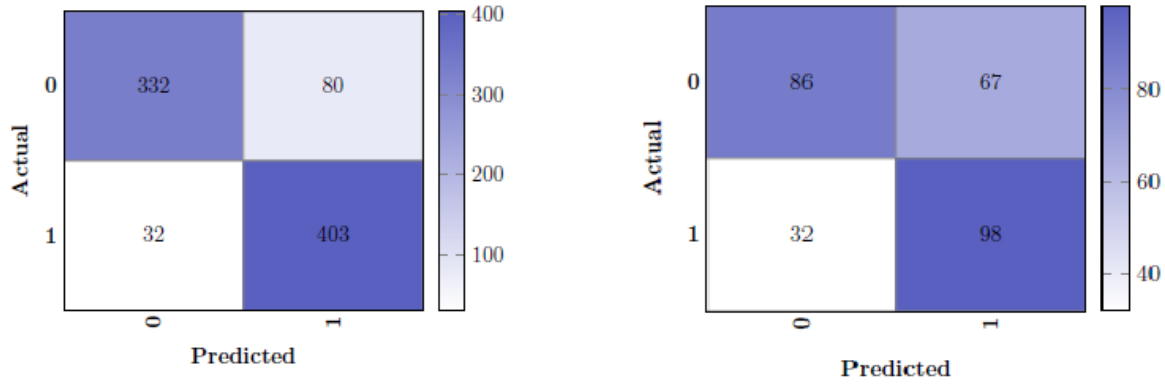
Fig. 10. GBC confusion matrices.

margins in the performance behaviors of the classification techniques, in contrast to the null hypothesis, which holds that all classification techniques have the same performance behavior when employed to address classification problems. According to the statistical findings shown in Table V, the GBC method is the most accurate classifier. This shows that the GBC technique came in first for classification accuracy shared with the KNC classifier while coming in third rank for precision rate, behind RFC and KNC classifiers, first in recall rate, and first in F1 metric measure shared with the RFC classifier. These findings demonstrate the GBC classification method's breadth is better than that of its competitors. In drawing things to a close, it is



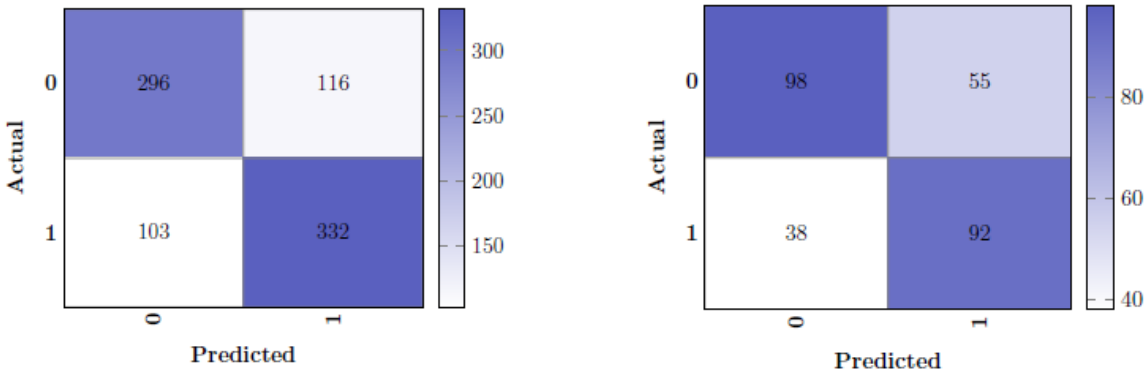
(a) Training. (b) Testing.

Fig. 11. GNB confusion matrices.



(a) Training. (b) Testing.

Fig. 12. RFC confusion matrices.



(a) Training. (b) Testing.

Fig. 13. KNC confusion matrices.

evident that the GBC classification method received the most excellent rank in terms of the total quantity of recall rate, with a rank of 1.0, the lowest rank of all the ranks that the other classifiers have.

The difference between the control classifier and its rivals was then demonstrated using Holm's statistical test as a *post-hoc* statistical method. Friedman's test findings confirm this, demonstrating that the control classification technique outperforms all others in each evaluation metric measure. The statistical findings from Holm's statistical process are shown

TABLE V. AVERAGE RANKING RESULTS OF ALL RIVAL CLASSIFICATION TECHNIQUES REGARDING ACCURACY, PRECISION, RECALL, AND F1 METRIC MEASURES USING FRIEDMAN'S TEST

Classifier	Accuracy	Precision	Recall	F1	Total ranking
LR	5.25	5.5	4.0	5.0	19.75
SVM	4.75	5.0	3.0	4.0	16.75
GBC	2.0	3.0	1.0	1.5	7.5
GNB	4.75	3.75	5.5	6.0	20
KNC	2.0	2.75	3.5	3.0	11.25
RFC	2.25	1.0	4.0	1.5	8.75
p-value	0.220640	0.177047	0.279401	0.083747	

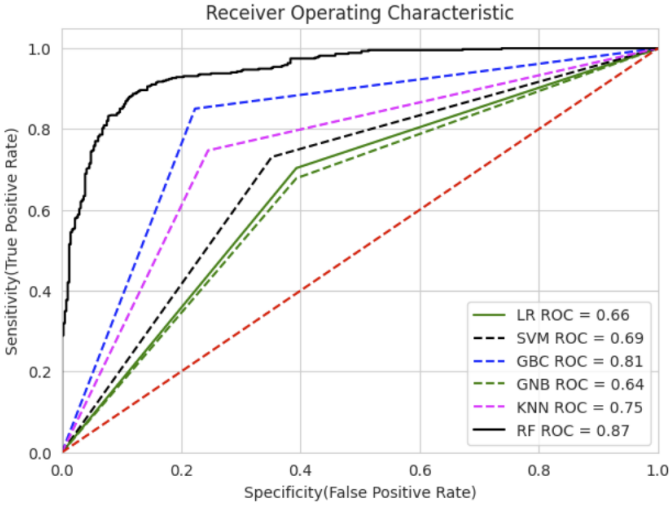


Fig. 14. The Receiver Operating Characteristic curve for the different techniques.

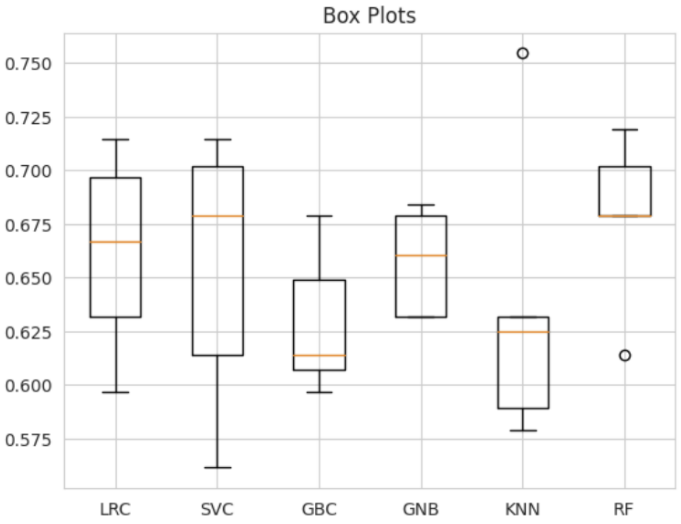


Fig. 15. Box-and-whisker plot for the used techniques.

in Table VI. As per the data reported in Table VI, the control classifier's rank is R_0, the ith classifier's rank is R^i, the effect size of the control classifier's classification technique on the ith classifier is ES, and the statistical difference between two classification techniques is z.

Holm's test was utilized to assess the competing classification systems. This test eliminates hypotheses with p-values of $\leq$ 0.010000 for classification accuracy, $\leq$ 0.010000 for precision, $\leq$ 0.010000 for recall, and $\leq$ 0.010000 for F1 rates, respectively. Table VI demonstrates that GBC outperforms LR, SVM, GNB, KNC, and RFC in terms of classification accuracy, even if there is no statistically significant difference between GBC and the other classification techniques (i.e. LR, SVM, GNB, KNC, and RFC). Friedman's and Holm's test-based statistical precision findings show that the RFC technique and the GNB, KNC, and GBC classification methods do not vary significantly from one another. The RFC approach, however, differs dramatically from the two classification techniques (LR and SVM). The findings in terms of the recall rate show that while GBC differs significantly from the GNB classification technique, it does not differ from the other three competing classifiers (i.e. LR, SVM, KNC, and RFC).

Tables VI and V display the results of Holm's and Friedman's tests, respectively, which demonstrate that the GBC classification method outperforms the other competing methods in achieving promising accuracy and precision rates for the datasets being studied. Overall, the GBC classification method outperformed a number of cutting-edge classification methods disclosed in the literature, including LR, SVM, KNC, and RFC, according to the results of the statistical study discussed above. This demonstrates the GBC method's consistent performance and attests to the fact that it successfully solves classification issues with low, medium, and high dimensions. In addition, GBC's performance degree is close to that of RFC and KNC classifiers as per the average rankings of classification techniques regarding precision rate. Still, the performance level of GNB and LR classifiers falls far short of its RFC and KNC competitors. This leads one to the conclusion that the GBC as a classification model gives it such exceptional capacity to handle classification difficulties. These statistical analysis tests demonstrate the reliability and suitability of the GBC tool as a classification method. These findings provide compelling justifications for employing the GBC classifier to classify difficult datasets progressively.

VII. CONCLUSIONS AND FUTURE WORK

This research aims to examine the feasibility of using machine learning techniques to identify cases of OSA. Traditional techniques of diagnosing OSA are costly and logistically difficult, even though they impact a large percentage of adults. Methods for diagnosing OSA in this study included Logistic Regression, Support Vector Machines, Gradient Boosting Classifier, Random Forest Classifier, Gaussian Naive Bayes,

TABLE VI. RESULTS OF HOLM'S TEST BETWEEN SEVERAL CLASSIFICATION TECHNIQUES

Classification accuracy (GBC is the control classifier)					
i	Algorithm	$z = \frac{(R_0 - R^i)}{SE}$	p-value	$\alpha \div i$	Hypothesis
5	LR	1.737198	0.0823522	0.010000	Not_Rejected
4	SVM	1.469936	0.141578	0.012500	Not_Rejected
3	GNB	1.469936	0.141578	0.016666	Not_Rejected
2	RFC	0.133630-	0.893694	0.025000	Not_Rejected
1	KNC	0.000000	1.000000	0.050000	Not_Rejected

Precision (RFC is the control classifier)					
i	Algorithm	$z = \frac{(R_0 - R^i)}{SE}$	p-value	$\alpha \div i$	Hypothesis
5	LR	2.405351	0.016156	0.010000	Rejected
4	SVM	2.138089	0.032509	0.012500	Rejected
3	GNB	1.469936	0.141578	0.016666	Not_Rejected
2	GBC	1.069044	0.285049	0.025000	Not_Rejected
1	KNC	0.935414	0.349574	0.050000	Not_Rejected

Recall (GBC is the control classifier)					
i	Algorithm	$z = \frac{(R_0 - R^i)}{SE}$	p-value	$\alpha \div i$	Hypothesis
5	GNB	2.405351	0.016156	0.010000	Rejected
4	LR	1.603567	0.108809	0.012500	Not_Rejected
3	RFC	1.603567	0.108809	0.016666	Not_Rejected
2	KNC	1.336306	0.181449	0.025000	Not_Rejected
1	SVM	1.069044	0.285049	0.050000	Not_Rejected

F1 (GBC is the control classifier)					
i	Algorithm	$z = \frac{(R_0 - R^i)}{SE}$	p-value	$\alpha \div i$	Hypothesis
5	GNB	2.405351	0.016156	0.010000	Rejected
4	LR	1.870828	0.061368	0.012500	Not rejected
3	SVM	1.336306	0.181449	0.016666	Not rejected
2	KNC	0.801783	0.422678	0.025000	Not rejected
1	RFC	0.000000	1.000000	0.050000	Not rejected

and K-Nearest Neighbors Classifier. Results showed that the Random Forest Classifier performed the best, with an accuracy of 0.87 during training and 0.65 during testing. The ROC curve produced a score of 0.87. The proposed work achieved classification accuracy comparable to other related studies. However, unlike most existing pieces that utilize PSG or ECG, which can be costly and time-consuming for physicians and patients, we employed physical parameters that are easy to obtain. Future research may explore the potential of other machine learning (ML) techniques, including artificial neural networks (ANN) and decision trees (DT), to address the problem at hand. By exploring these different ML techniques, it may be possible to improve the accuracy and generalizability of the model and gain new insights into the problem domain.

ACKNOWLEDGMENT

The authors extend their appreciation to Taif University, Saudi Arabia, for supporting this work through project number (TU-DSPP-2024-201)

DECLARATIONS

- Funding: This research was funded by Taif University, Taif, Saudi Arabia (TU-DSPP-2024-201).
- Data Availability Statements: The data supporting this study's findings are available on request from the corresponding author.
- Ethical Approval: The authors declare that ethical standards have been followed and that no human participants or animals were involved in this research.
- Conflict of Interest: The authors have no competing interests to declare relevant to this article's content.

REFERENCES

[1] S. P. Patil, H. Schneider, A. R. Schwartz, and P. L. Smith, "Adult obstructive sleep apnea: pathophysiology and diagnosis," *Chest*, vol. 132, no. 1, pp. 325–337, 2007.

[2] A. Sheta, H. Turabieh, M. Braik, and S. R. Surani, "Diagnosis of obstructive sleep apnea using logistic regression and artificial neural networks models," in *Proceedings of the Future Technologies Conference (FTC) 2019: Volume 1*. Springer, 2020, pp. 766–784.

[3] T. D. Bradley and J. S. Floras, "Obstructive sleep apnoea and its cardiovascular consequences," *The Lancet*, vol. 373, no. 9657, pp. 82–93, 2009.

[4] V. K. Somers, D. P. White, R. Amin, W. T. Abraham, F. Costa, A. Culebras, S. Daniels, J. S. Floras, C. E. Hunt, L. J. Olson *et al.*, "Sleep apnea and cardiovascular disease: An american heart association/american college of cardiology foundation scientific statement from the american heart association council for high blood pressure research professional education committee, council on clinical cardiology, stroke council, and council on cardiovascular nursing in collaboration with the national heart, lung, and blood institute national center on sleep disorders research (national institutes of health)," *Circulation*, vol. 118, no. 10, pp. 1080–1111, 2008.

[5] C. V. Senaratna, J. L. Perret, C. J. Lodge, A. J. Lowe, B. E. Campbell, M. C. Matheson, G. S. Hamilton, and S. C. Dharmage, "Prevalence of

obstructive sleep apnea in the general population: a systematic review," *Sleep medicine reviews*, vol. 34, pp. 70–81, 2017.

[6] H. Yue, Y. Lin, Y. Wu, Y. Wang, Y. Li, X. Guo, Y. Huang, W. Wen, G. Zhao, X. Pang *et al.*, "Deep learning for diagnosis and classification of obstructive sleep apnea: A nasal airflow-based multi-resolution residual network," *Nature and Science of Sleep*, pp. 361–373, 2021.

[7] E. Urtnasan, J.-U. Park, and K.-J. Lee, "Multiclass classification of obstructive sleep apnea/hypopnea based on a convolutional neural network from a single-lead electrocardiogram," *Physiological measurement*, vol. 39, no. 6, p. 065003, 2018.

[8] J. Ramesh, N. Keeran, A. Sagahyroon, and F. Aloul, "Towards validating the effectiveness of obstructive sleep apnea classification from electronic health records using machine learning," in *Healthcare*, vol. 9, no. 11. MDPI, 2021, p. 1450.

[9] L. Cen, Z. L. Yu, T. Kluge, and W. Ser, "Automatic system for obstructive sleep apnea events detection using convolutional neural network," in *2018 40th annual international conference of the IEEE engineering in medicine and biology society (EMBC)*. IEEE, 2018, pp. 3975–3978.

[10] A. Sheta, H. Turabieh, T. Thaher, J. Too, M. Mafarja, M. S. Hossain, and S. R. Surani, "Diagnosis of obstructive sleep apnea from ecg signals using machine learning and deep learning classifiers," *Applied Sciences*, vol. 11, no. 14, 2021. [Online]. Available: https://www.mdpi.com/2076-3417/11/14/6622

[11] M. Deviaene, D. Testelmans, P. Borzée, B. Buyse, S. V. Huffel, and C. Varon, "Feature selection algorithm based on random forest applied to sleep apnea detection," in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2019, pp. 2580–2583.

[12] M.-S. Choi, D.-H. Han, J.-W. Choi, and M.-S. Kang, "A study on improving sleep apnea diagnoses using machine learning based on the stop-bang questionnaire," *Applied Sciences*, vol. 14, no. 7, 2024. [Online]. Available: https://www.mdpi.com/2076-3417/14/7/3117

[13] A. Altintas, Y. Yegin, M. Çelik, K. H. Kaya, A. K. Koç, and F. T. Kayhan, "Interobserver consistency of drug-induced sleep endoscopy in diagnosing obstructive sleep apnea using a vote classification system," *Journal of Craniofacial Surgery*, vol. 29, no. 2, pp. e140–e143, 2018.

[14] K. Bond and A. Sheta, "Medical data classification using machine learning techniques," *International Journal of Computer Applications*, vol. 183, no. 6, pp. 1–8, Jun 2021. [Online]. Available: http://www.ijcaonline.org/archives/volume183/number6/31928-2021921339

[15] W. H. Elashmawi, A. Djellal, A. Sheta, S. Surani, and S. Aljahdal, "Machine learning for enhanced copd diagnosis: A comparative analysis of classification algorithms," *Diagnostics*, vol. 14, no. 24, 2024. [Online]. Available: https://www.mdpi.com/2075-4418/14/24/2822

[16] A. Sheta, W. H. Elashmawi, A. Al-Qerem, and E. S. Othman, "Utilizing various machine learning techniques for diabetes mellitus feature selection and classification," *International Journal of Advanced Computer Science and Applications*, vol. 15, no. 3, 2024. [Online]. Available: http://dx.doi.org/10.14569/IJACSA.2024.01503134

[17] R. Cuingnet, E. Gerardin, J. Tessieras, G. Auzias, S. Lehericy, M.-O. Habert, M. Chupin, H. Benali, and O. Colliot, "Automatic classification of patients with alzheimer's disease from structural mri: A comparison of ten methods using the adni database," *NeuroImage*, vol. 56, no. 2, pp. 766–781, 2011.

[18] C. Mencar, C. Gallo, M. Mantero, P. Tarsia, G. Carpagnano, M. P. Foschino Barbaro, and D. Lacedonia, "Application of machine learning to predict obstructive sleep apnea syndrome (osas) severity," *Health Informatics Journal*, pp. 1–20, 03 2020.

[19] S. Ha, S. J. Choi, S. Lee, R. H. Wijaya, J. H. Kim, E. Y. Joo, and J. K. Kim, "Predicting the risk of sleep disorders using a machine learning–based simple questionnaire: Development and validation study," *J Med Internet Res*, vol. 25, p. e46520, Sep 2023. [Online]. Available: https://www.jmir.org/2023/1/e46520

[20] A. Javeed, J. Berglund, A. Luiza, M. Saleem, and Peter, "Predictive power of xgboost_bilstm model: A machine-learning approach for accurate sleep apnea detection using electronic health data," *International Journal of Computational Intelligence Systems*, p. 188, 11 2023.

[21] A. Sheta, S. Subramanian, S. R. Surani, and M. Braik, "Diagnosis of obstructive sleep apnea using machine learning," in *2023 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*, 2023, pp. 12–17.

[22] W. Z. T. Tareq, *Sleep Disorders Detection and Classification Using Random Forests Algorithm*. Cham: Springer International Publishing, 2024, pp. 257–266.

[23] K. Liu, S. Geng, P. Shen, L. Zhao, P. Zhou, and W. Liu, "Development and application of a machine learning-based predictive model for obstructive sleep apnea screening," *Frontiers in Big Data*, vol. 7, 2024. [Online]. Available: https://www.frontiersin.org/journals/big-data/articles/10.3389/fdata.2024.1353469

[24] G. Bazoukis, S. C. Bollepalli, C. T. Chung, X. Li, G. Tse, B. L. Bartley, S. Batool-Anwar, S. F. Quan, and A. A. Armoundas, "Application of artificial intelligence in the diagnosis of sleep apnea," *Journal of Clinical Sleep Medicine*, vol. 19, no. 7, pp. 1337–1363, 2023. [Online]. Available: https://jcsm.aasm.org/doi/abs/10.5664/jcsm.10532

[25] G. Surrel, A. Aminifar, F. Rincón, S. Murali, and D. Atienza, "Online obstructive sleep apnea detection on medical wearable sensors," *IEEE transactions on biomedical circuits and systems*, vol. 12, no. 4, pp. 762–773, 2018.

[26] H. Korkalainen, J. Aakko, S. Nikkonen, S. Kainulainen, A. Leino, B. Duce, I. O. Afara, S. Myllymaa, J. Töyräs, and T. Leppänen, "Accurate deep learning-based sleep staging in a clinical population with suspected obstructive sleep apnea," *IEEE journal of biomedical and health informatics*, vol. 24, no. 7, pp. 2073–2081, 2019.

[27] A. H. Yüzer, H. Sümbül, M. Nour, and K. Polat, "A different sleep apnea classification system with neural network based on the acceleration signals," *Applied Acoustics*, vol. 163, p. 107225, 2020.

[28] S. Sperandei, "Understanding logistic regression analysis," *Biochemia medica*, vol. 24, no. 1, pp. 12–18, 2014.

[29] R. J. Bacue, "An analytic overview of estes' statistical learning theory," *IEEE Transactions on Neural Networks*, pp. 988–999, 1999.

[30] J. H. Friedman, "Greedy function approximation: A gradient boosting machine." *The Annals of Statistics*, vol. 29, no. 5, pp. 1189 – 1232, 2001. [Online]. Available: https://doi.org/10.1214/aos/1013203451

[31] I. Rish, "An empirical study of the naive bayes classifier," in *IJCAI 2001 workshop on empirical methods in artificial intelligence*, vol. 3, no. 22. IBM New York, 2001, pp. 41–46.

[32] M. Braik, H. Al-Zoubi, and H. Al-Hiary, "Pedestrian detection using multiple feature channels and contour cues with census transform histogram and random forest classifier," *Pattern Analysis and Applications*, vol. 23, no. 2, pp. 751–769, 2020.

[33] T. Hastie, R. Tibshirani, and J. Friedman, *Random Forests*. New York, NY: Springer New York, 2009, pp. 587–604.

[34] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, p. 5–32, oct 2001.

[35] X. Li, M. Li, Y. Zhang, and X. Deng, "A new random forest method based on belief decision trees and its application in intention estimation," in *2021 33rd Chinese Control and Decision Conference (CCDC)*, 2021, pp. 6008–6012.

[36] B. Chakradhar, I. S. Siva Rao, V. Jhansy Archana, and C. V. M. K. Hari, "Detection of malignancy on dermis using j48 and random forest classifiers," in *2020 International Conference on Computer Science, Engineering and Applications (ICCSEA)*, 2020, pp. 1–6.

[37] U. N. A and K. Dharmarajan, "Diabetes prediction using random forest classifier with different wrapper methods," in *2022 International Conference on Edge Computing and Applications (ICECAA)*, 2022, pp. 1705–1710.

[38] R. Hummel, T. D. Bradley, G. R. Fernie, S. I. Chang, and H. Alshaer, "Estimation of sleep status in sleep apnea patients using a novel head actigraphy technique," in *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2015, pp. 5416–5419.