

Article

Not peer-reviewed version

---

# Early Screening of Sleep-Disordered Breathing Using Metaheuristic-Optimized Extreme Learning Machines

---

[Thaer Thaher](#)\*, [Alaa Sheta](#)\*, Huthaifa I. Ashqar, [Hamouda Chantar](#), [Salim Surani](#)

Posted Date: 1 May 2026

doi: 10.20944/preprints202604.2232.v1

Keywords: obstructive sleep apnea; extreme learning machine (ELM); metaheuristics; machine learning; optimization







Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC, OpenAlex.

Copyright: This open access article is published under a [Creative Commons CC BY 4.0 license](#), which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

# Early Screening of Sleep-Disordered Breathing Using Metaheuristic-Optimized Extreme Learning Machines

Thaer Thaher<sup>1,\*</sup> , Alaa Sheta<sup>2,\*</sup> , Huthaifa I. Ashqar<sup>3,4</sup> , Hamouda Chantar<sup>5</sup>   
and Salim Surani<sup>6</sup>

<sup>1</sup> Department of Computer Systems Engineering, Arab American University, Jenin P.O. Box 240, Palestine

<sup>2</sup> Computer Science Department, Southern Connecticut State University, New Haven, CT 06514, USA

<sup>3</sup> Department of AI and Data Science, Arab American University, Jenin P.O. Box 240, Palestine

<sup>4</sup> AI Program, Columbia University, New York, NY 10027, USA

<sup>5</sup> Faculty of Information Technology, Sebha University, Sebha 18758, Libya

<sup>6</sup> Department of Pharmacy & Medicine, College Station, Texas A&M University, Texas, USA

\* Correspondence: thaer.thaher@aaup.edu (T.T.); shetaa1@southernct.edu (A.S.)

## Abstract

**Background/Objectives:** Obstructive sleep apnea (OSA) is a common and serious sleep-related disorder that causes repeated interruptions in breathing during sleep. Traditional diagnostic methods, such as polysomnography, are accurate but costly, time-consuming, and unsuitable for large-scale screening. This study proposes and evaluates a lightweight diagnostic framework based on an Extreme Learning Machine (ELM) optimized by a set of basic and advanced metaheuristic optimizers (GA, RUN, MEO, CL-PSO, HI-WOA, GWO, HGS, HHO, SeaHO, MGO, and the hybrid GWO-WOA). The model aims to improve early detection of OSA using demographic and clinical data. **Methods:** Two real datasets were employed to train and evaluate the proposed framework: (i) a clinical OSA dataset with 274 subjects and 31 demographic/anthropometric and sleep-related predictors, and (ii) a public strongly imbalanced Sleep-Disordered Breathing (SDB) dataset with 500 subjects and 10 structured predictors. Metaheuristic algorithms are used to optimize ELM weights and biases, addressing the instability of random initialization and improving model generalization. The optimized models are evaluated against eight baseline classifiers, including Logistic Regression (LR), k-nearest neighbours (KNN), Decision Tree (DT), Random Forest (RF), Support Vector Machine (SVM), Multilayer Perceptron (MLP), XGBoost (XGB), and a standard ELM classifier. **Results:** Results show that metaheuristic optimization improves ELM on the OSA dataset, increasing ROC-AUC from 0.6527 to about 0.73 and accuracy from 0.6573 to about 0.69–0.70, while on the highly imbalanced SDB dataset, it yields modest ROC-AUC gains (from 0.5132 to about 0.544–0.548) with small decreases in accuracy and F1-score. **Conclusions:** The proposed framework provides a fast, lightweight, and cost-effective screening tool for large-scale, resource-limited healthcare settings, enabling early OSA detection and preventive intervention.

**Keywords:** obstructive sleep apnea; Extreme Learning Machine (ELM); metaheuristics; machine learning; optimization

## 1. Introduction

Obstructive sleep apnea (OSA) is a common and serious sleep-related breathing disorder characterized by repeated episodes of partial or complete obstruction of the upper airway during sleep [1]. Its symptoms include loud snoring, morning headaches, excessive daytime sleepiness, and difficulty concentrating [2,3]. Globally, more than 100 million individuals are estimated to suffer from sleep apnea, affecting both adults and children [4,5]. In the United States alone, nearly 22 million people are diagnosed with moderate to severe forms of OSA [6]. The apnea-hypopnea index (AHI) is widely used to classify the severity of OSA, with thresholds distinguishing mild, moderate, and severe cases [6].

Beyond its impact on sleep quality, untreated OSA is strongly associated with cardiovascular disease, hypertension, diabetes, depression, and an increased risk of workplace and traffic accidents [7]. The American Academy of Sleep Medicine has reported that the economic burden of undiagnosed OSA in the U.S. exceeds \$140 billion annually due to lost productivity, healthcare costs, and accident-related expenses [7]. These figures highlight the urgent need for accurate and accessible diagnostic solutions.

While many OSA diagnostic models basically depend on physiological signals such as respiratory rate (RR), spirometry, and electrocardiograms (ECG), these approaches encounter vital challenges. For instance, ECG-based diagnostic methods require qualified staff and specialized equipment, which makes them costly and difficult to adopt. Signal quality across different sleep phases can reduce the robustness of these methods, particularly for large-scale monitoring. Moreover, the variance of signal quality during different sleep phases can lower the robustness of these methods, in particular when controlled clinical settings are unavailable. Such shortcomings underscore the need for different options that are simple, low-cost, and easy to implement in larger populations. In this regard, clinical and demographic data are feasible sources of information. Machine learning approaches can then be applied to this valuable data to build predictive models that improve screening efficiency and robustness.

Classic diagnostic approaches such as polysomnography (PSG) remain the clinical standard but are expensive, time-consuming, and impractical for large-scale screening. Thus, alternative techniques have been investigated to promote and simplify early detection [8]. Demographic and clinical variables such as age, body mass index (BMI), snoring, neck circumference, and excessive daytime sleepiness have been shown to correlate strongly with OSA and are commonly used for screening and risk assessment [9–11]. Many previous studies have applied machine learning algorithms to these variables to develop predictive models, achieving favorable results while reducing reliance on costly physiological signals.

One machine learning model, the extreme learning machine (ELM), has shown promising performance due to its rapid training and strong generalization [12,13]. Due to its lightweight nature, the ELM model is particularly suitable for inexpensive, large-scale OSA screening tools where computational resources are limited. Despite ELM's significant strengths, its performance heavily relies on the initial configurations of hidden biases and input weights. In the standard ELM, these parameters are randomly initialized, which may lead to several problems, such as inconsistent results, poor conditioning, or overfitting [14,15]. For this reason, several works have proposed coupling ELM with global optimization techniques that search for better hidden-layer weights and biases, rather than relying on a single random initialization [16].

The effectiveness of ELM highly relies on the initial configuration of the hidden-layer weights and biases. Random initialization of its parameters leads to unsatisfactory solutions, unsteady convergence, and contradictory accuracy across runs. To overcome this problem, metaheuristic optimization techniques can be employed to drive the initialization and training of ELMs [16]. Metaheuristic optimization algorithms are designed to explore complex search spaces and can avoid local minima in non-convex problems. Examples include evolutionary algorithms, swarm intelligence methods, physics-based optimizers, and math-inspired strategies [17]. When combined with ELM, these methods have improved accuracy and training stability in different domains. However, most existing studies on OSA or sleep-disordered breathing (SDB) use a single optimizer, a limited set of baseline models, and often only a single dataset. As a result, there is still limited evidence about which metaheuristics are most suitable for ELM in this application and how much they help under different data conditions, especially in imbalanced datasets.

Motivated by these considerations, this paper proposes a metaheuristic-optimized ELM framework for early detection of OSA and SDB using low-cost demographic, anthropometric, and physiological features. We use two real datasets: a clinical OSA dataset with rich demographic and polysomnographic variables, and a public Sleep-Disordered Breathing (SDB) dataset that includes demographic information and short-term cardiorespiratory signals. A diverse set of metaheuristic optimizers from different families and integrated with ELM and compared against several standard ML classifiers. The main contributions are summarized as follows:

1. We design a lightweight diagnostic framework based on ELM that uses readily collected demographic and clinical features, aiming to enable practical, scalable screening for OSA and SDB in resource-constrained settings.
2. We systematically integrate eleven metaheuristic algorithms covering evolutionary, math-based, physics-based, and swarm-based families to optimize ELM hidden-layer weights and biases under a common objective and protocol.
3. We perform an extensive empirical study on two real datasets, benchmarking all metaheuristic-optimized ELM variants against standard ML baselines in terms of accuracy, F1-score, ROC-AUC, generalization gap, convergence behaviour, and computational time.
4. We show that metaheuristic-optimized ELM models achieve consistent improvements over plain ELM and several baselines on the OSA dataset and provide moderate but meaningful gains in discrimination on the more imbalanced SDB dataset.

The remainder of this paper is organized as follows: Section 2 reviews related works. Section 3 describes the datasets, preprocessing steps, the proposed optimized-ELM framework, and the experimental setup and evaluation measures. Section 4 presents and discusses the experimental results for both baseline classifiers and metaheuristic-optimized ELM models. Finally, Section 5 summarizes the main findings, outlines practical implications, and suggests directions for future research.

## 2. Related Works

### 2.1. OSA Detection Using Demographic and Clinical Data

Traditional early identification of obstructive sleep apnea conditions depends on clinical evaluations and demographic information, which provide accessibility and practicality to these features. It has been discovered that factors such as neck circumference, age, body mass index (BMI), and gender, along with other lifestyle indicators including smoking or alcohol consumption, have been highly correlated with the risk of OSA problems [18,19]. Clinically observed symptoms such as habitual snoring, excessive daytime sleepiness, and apneas during sleep are also common indicators used in screening processes for diagnosing OSA. In contrast to the monitoring of physiological signals, easily collected clinical and demographic information provides a low-cost, non-invasive alternative for stratifying the risk of OSA [20–22]. These variables offer advantages over large-scale screening and resource-constrained contexts, where access to sleep laboratories or specialized tools is limited. Although clinical and demographic features offer precious and valuable predictors, they have difficulty in discrimination between OSA and healthy cases when utilized separately [23,24]. This drawback has motivated the authors of this work to integrate machine learning models to enhance the accuracy of early prediction of OSA by identifying complex, non-linear interactions and dependencies among OSA features. This finding aligns with recent work in interpretable machine learning, where researchers have used clinical features and questionnaire data to predict severe OSA risk and, at the same time, pinpoint the risk factors that matter most [25].

### 2.2. Machine Learning and Deep Learning Models in OSA

The widespread availability of medical data has encouraged researchers to adopt deep learning and machine learning techniques for OSA detection [26–29]. Conventional machine learning algorithms, such as support vector machines, logistic regression, and decision trees, as well as ensemble learning models, have been used to classify OSA severity based on physiological, demographic, and clinical features [10,30–32]. These algorithms can efficiently handle multidimensional data, learn feature interactions, and deliver accurate classification results. Also, the adoption of deep learning and neural networks has been increasing rapidly, providing automatic extraction of valuable hierarchical patterns from structured data and basic physiological signals [21,33,34]. Traditional and recurrent architectures, such as Recurrent Neural Networks (RNNs), are promising for analysing airflow, electrocardiogram, and oxygen saturation signals. Despite their robust predictive power, deep learning models often require substantial computational resources, large amounts of training data, and careful

hyperparameter tuning. Moreover, the inability to interpret physiological signals and the high cost of acquiring them hinder the clinical adoption of deep learning models. In this regard, lightweight learning models such as the extreme learning machine (ELM) have been adopted as alternatives to deep learning models due to their effective balance between prediction accuracy and computational cost, making them appropriate for highly scalable screening applications. [35–37].

### 2.3. Metaheuristic-Based Optimization of ELM

Extreme learning machines are preferred due to their fast training speed and robust generalization power. However, their performance heavily relies on the initial configuration of hidden-layer weights and biases [35,38,39]. Random initialization of weights and biases may lead to unsatisfactory solutions, unsteady convergence, and contradictory accuracy across runs. To overcome this problem, metaheuristic optimization techniques can be employed to guide the initialization and training of ELMs. Metaheuristics, such as swarm intelligence techniques, are designed to efficiently explore the search space, avoid premature convergence, and find near-optimal solutions for complex optimization problems. Incorporating optimization techniques with ELM models improves learning stability, avoids overfitting, and boosts classification accuracy [35,38,39]. For OSA identification, metaheuristic-optimized ELM models hold strong promise. They merge the efficiency and scalability of ELM models with the power and high capability of global optimization techniques. This integration offers strong potential to build a lightweight, robust, and accurate AI-based screening model that can be used across various healthcare applications, especially when clinical infrastructure and required computational resources are unavailable.

Merging these perspectives, it is confirmed that OSA detection can be improved using three complementary aspects. Firstly, despite their limited predictive ability when applied alone, clinical and demographic features provide a foundation for early identification of OSA. Secondly, machine learning and deep learning techniques can improve classification performance by detecting complex, hidden interactions between features. However, they face challenges such as interpretability, scalability, and computational cost. Thirdly, using meta-heuristic techniques to optimize extreme learning machine models provides a promising solution through exploiting the capability of lightweight models and the power of global optimization approaches. This combination highlights a research gap at the intersection of accuracy, accessibility, and computational capability; that is, a field where metaheuristic-optimized ELM models can deliver significant improvements for resource-limited and large-scale OSA screening.

## 3. Materials and Methods

### 3.1. Dataset Description

This study relies on two real clinical datasets to train and evaluate the proposed models for early detection of sleep apnea and sleep-disordered breathing. Both datasets include a binary diagnosis label and a set of demographic, anthropometric, and physiological features, but they differ in sample size, number of predictors, and class balance. In the following subsections, we describe each dataset, its main characteristics, and the feature space considered in our experiments.

#### 3.1.1. Obstructive Sleep Apnea (OSA) Dataset

The first dataset is a clinical dataset of obstructive sleep apnea (OSA). The initial unprocessed dataset comprises 620 patients, including 366 males and 254 females. The age range for males spans 19 to 88 years, while for females, it spans 20 to 96 years. Notably, the prevalence of snoring was 92.6% among males and 91.7% among females. Each patient underwent comprehensive full-night monitoring as part of the study. [40]. The processed version employed in this study cohort with 274 patients and 31 input variables, together with a binary outcome indicating the presence of OSA (class = 1) or the absence of OSA (class = 0). As summarized in Table 1, the dataset contains 125 positive and 149 negative cases, so it is only moderately imbalanced. The features cover demographic and anthropometric information (e.g., age, race, sex, BMI category, waist, hip and neck circumferences), comorbidities (diabetes, hypertension, coronary artery disease, cerebrovascular accident), questionnaire-based indicators

(Epworth Sleepiness Scale, Berlin questionnaire, daytime sleepiness), and polysomnographic indices (RDI, AHI by sleep stage and body position, arousal and awakening indices, oxygen saturation statistics, and periodic limb movement index). Most variables are numerical; some are binary or ordinal but are encoded as integers. The average age is around the mid-50s, and BMI scores indicate that many patients fall in the overweight or obese categories.

### 3.1.2. Sleep-Disordered Breathing (SDB) Detection Dataset

The second dataset is the Sleep Disordered Breathing (SDB) detection dataset, which contains 500 subjects collected from a publicly available resource on kaggle<sup>1</sup> [41]. The original dataset includes 18 variables, combining structured attributes with free-text fields such as physician notes and patient symptoms. In this study, we restrict the analysis to 10 structured predictors and a binary outcome, as shown in Table 1. The selected predictors are age, gender, BMI, snoring, average oxygen saturation, apnea-hypopnea index (AHI), ECG-derived heart rate, average SpO<sub>2</sub>, nasal airflow, and chest movement (see Table 2). The target variable `Diagnosis_of_SDB` is recoded into a binary label, where all mild, moderate, and severe SDB cases are mapped to the positive class, and the remaining subjects are treated as negative. This results in 381 positive and 119 negative samples, corresponding to a strong class imbalance with about 76% SDB cases. The features are all numeric or binary-coded and capture both anthropometric risk factors and short-term cardiorespiratory signals.

**Table 1.** Summary of the two datasets used in this study.

Dataset	No. samples	No. features	Negative samples	Positive samples	Positive rate (%)
OSA	274	31	149	125	45.6
SDB	500	10	119	381	76.2

### 3.1.3. Summary and Feature Characteristics

Table 1 summarizes the main characteristics of the two datasets, including the number of samples, number of features, and class distribution. The OSA dataset is smaller but richer in clinical and polysomnographic variables. In contrast, the SDB dataset is larger but uses a more compact set of 10 structured predictors and exhibits a more pronounced class imbalance. Table 2 provides a unified overview of all 38 unique predictor variables appearing in either dataset, indicating for each feature whether it is present in the OSA dataset, the SDB dataset, or both. The table also reports the maximum number of distinct values observed across datasets and classifies each variable as continuous numeric or categorical/ordinal, together with a brief description. In both datasets, all predictors are ultimately treated as numeric inputs to the models, while the details of scaling, encoding, and any sampling strategies are described in Section 3.2.

## 3.2. Data Processing

For both datasets, we used a simple, consistent preprocessing pipeline. We separated predictors  $X$  and labels  $y$ , cleaned the data, removed non-informative fields, and standardized all input features using z-score scaling. All steps were applied identically for baseline models and the metaheuristic-optimized ELM.

### 3.2.1. OSA Dataset Preprocessing

For the OSA dataset, we used the processed version provided by the original authors [40], which already contains numeric predictors and a binary label (class = 1 for OSA, 0 for non-OSA). No missing values were present, and all 274 records and 31 input features were kept for analysis. Before model training, each feature was standardized on the training split using the z-score transform.

<sup>1</sup> <https://www.kaggle.com/datasets/ziya07/sleep-disordered-breathing-detection>

$$\tilde{x}_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j}, \quad (1)$$

Where  $\mu_j$  and  $\sigma_j$  are the mean and standard deviation of the feature  $j$  computed from the training data only, to avoid data leakage and preserve a strict train-test separation (i.e., preventing the preprocessing step from using any information from the evaluation split). The same transformation was then applied to the corresponding test data using these fixed parameters, which also reflects the real-world setting where future/unseen data are unavailable when estimating scaling statistics. Binary and ordinal variables were kept in their numeric form and scaled in the same way.

**Table 2.** Overview of predictor variables across the OSA and SDB datasets.

Feature	Type	In OSA	In SDB	Description
Race	Categorical	Yes	No	Race / ethnic group category.
Age	Numeric (continuous)	Yes	Yes	Patient age in years.
Sex	Categorical (binary)	Yes	Yes	Patient sex.
BMI*	Numeric / categorical	Yes	Yes	Body mass index or BMI category.
Epworth	Numeric (ordinal)	Yes	No	Epworth Sleepiness Scale total score.
Wast	Numeric (continuous)	Yes	No	Waist circumference (cm).
Hip	Numeric (continuous)	Yes	No	Hip circumference (cm).
RDI	Numeric (continuous)	Yes	No	Respiratory Disturbance Index per hour.
Neck	Numeric (continuous)	Yes	No	Neck circumference (cm).
M.Friedman	Ordinal categorical	Yes	No	Friedman tongue position grade (1–4).
Co-morbid	Categorical	Yes	No	Presence or count of comorbidities.
Snoring	Categorical (binary)	Yes	Yes	Snoring indicator.
Daytime sleepiness	Categorical (binary)	Yes	No	Self-reported daytime sleepiness.
DM	Categorical (binary)	Yes	No	Diabetes mellitus status.
HTN	Categorical (binary)	Yes	No	Hypertension status.
CAD	Categorical (binary)	Yes	No	Coronary artery disease status.
CVA	Categorical (binary)	Yes	No	History of cerebrovascular accident.
TST	Numeric (continuous)	Yes	No	Total sleep time.
Sleep Effic	Numeric (continuous)	Yes	No	Sleep efficiency (%).
REM AHI	Numeric (continuous)	Yes	No	Apnea–Hypopnea Index in REM sleep.
NREM AHI	Numeric (continuous)	Yes	No	Apnea–Hypopnea Index in NREM sleep.
Supine AHI	Numeric (continuous)	Yes	No	Apnea–Hypopnea Index in supine position.
Apnea Index	Numeric (continuous)	Yes	No	Number of apnea events per hour.
Hypopnea Index	Numeric (continuous)	Yes	No	Number of hypopnea events per hour.
Berlin Q	Categorical	Yes	No	Berlin questionnaire risk category.
Arousal index	Numeric (continuous)	Yes	No	Number of arousals per hour.
Awakening Index	Numeric (continuous)	Yes	No	Number of awakenings per hour.
PLM Index	Numeric (continuous)	Yes	No	Periodic limb movement index per hour.
Mins.SaO2	Numeric (continuous)	Yes	No	Minutes below oxygen saturation threshold.
Mins.SaO2Desats	Numeric (continuous)	Yes	No	Minutes with oxygen desaturation events.
Lowest SaO2	Numeric (continuous)	Yes	No	Lowest oxygen saturation recorded.
Oxygen_Saturation	Numeric (continuous)	No	Yes	Average oxygen saturation during recording.
AHI	Numeric (continuous)	No	Yes	Overall Apnea–Hypopnea Index.
ECG_Heart_Rate	Numeric (continuous)	No	Yes	Heart rate derived from ECG.
SpO2	Numeric (continuous)	No	Yes	Average peripheral oxygen saturation (%).
Nasal_Airflow	Numeric (continuous)	No	Yes	Normalized nasal airflow signal.
Chest_Movement	Numeric (continuous)	No	Yes	Normalized chest movement signal.

\* BMI is treated as a categorical variable in the OSA dataset and as a numeric variable in the SDB dataset.

### 3.2.2. SDB Dataset Preprocessing

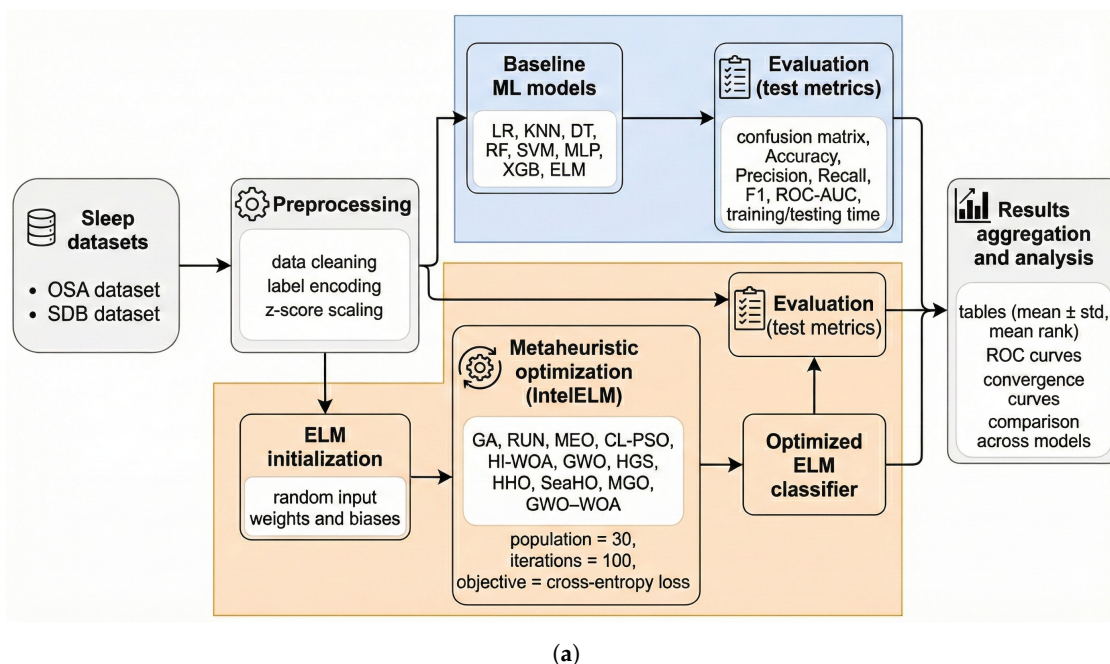
For the SDB dataset, we started from the publicly available Kaggle file [41]. The original data contain 500 subjects, 18 columns, and no missing values. We removed identifier, treatment, and free-text columns and retained ten structured predictors: Age, Gender, BMI, Snoring, Oxygen\_Saturation, AHI, ECG\_Heart\_Rate, SpO<sub>2</sub>, Nasal\_Airflow, and Chest\_Movement.

The original Diagnosis\_of\_SDB field includes four categories (Mild, Moderate, Severe, and no SDB). We converted it into a binary label, mapping Mild/Moderate/Severe to the positive class and the remaining subjects to the negative class. In the final processed file, Diagnosis\_of\_SDB = 1 indicates SDB and 0 indicates non-SDB, giving 381 positive and 119 negative samples.

We then standardized all ten predictors using the same z-score formula in Eq. (1), with  $\mu_j$  and  $\sigma_j$  estimated on the training data and reused for the test data.

### 3.3. Proposed Optimized-ELM Framework

The proposed pipeline begins with the two datasets (OSA and SDB), followed by data cleaning and preprocessing (label encoding, and z-score scaling). The processed data are then used in two branches. In the first branch, baseline classifiers are trained and evaluated to provide a reference performance level. In the second branch, the same preprocessed data are passed to the IntelELM framework, where different metaheuristics search for improved ELM weights and biases. In both branches, models are trained and tested under repeated experiments, and the resulting metrics are collected for analysis. Figure 1 illustrates the overall workflow from data collection to model evaluation and result analysis.



**Figure 1.** Schematic view of the pipeline and system flow.

#### 3.3.1. Basic ELM Classifier and Mathematical Formulation

The basic ELM is a popular type of single hidden-layer neural network (SLNN). Its variants are commonly used for different learning scenarios: sequential, batch, and incremental learning due to their rapid and efficient learning speed, suitable generalization ability, swift convergence rate, and ease of implementation [15]. In contrast to traditional learning algorithms, the main objective of ELM is to enhance generalization performance by minimizing both the norm of the output weights and the training error. According to Bartlett's theory of feedforward neural networks [42], a smaller weight norm correlates with improved generalization performance.

Single-hidden-layer feedforward neural networks (SLFNs) are among the most common neural network architectures. These networks are usually trained using gradient descent methods like backpropagation. However, these Gradient descent-based methods have significant drawbacks: they depend heavily on the network's starting weights, often get stuck in local minima, and have sluggish convergence rates. To address these limitations, ELM was initially introduced by Huang et al. [43]. In ELM, the weights linking the input layer to the hidden layer, along with hidden biases, are initialized randomly; subsequently, the connection weights between the hidden layer and the output layer are analytically computed using a straightforward mathematical approach called the Moore-Penrose generalized inverse (a least-squares solution) [44]. This method avoids the pitfalls of gradient descent while maintaining efficiency.

ELM initially assigns random weights and biases to the input layer and subsequently determines the output-layer weights based on these random values. Typically, the SLNN model has  $n$  input-layer neurons,  $h$  hidden-layer neurons, and  $k$  output-layer neurons. The architecture of this model is illustrated in Figure 2. As presented in [15], the activation function of SLNN can be written as in Eq. (2).

$$E_i = \left( \sum_{j=1}^h (S_j f(w_j, b_j, X_j)) \right) \quad (2)$$

where  $w_j$  indicates the input weight and  $b_j$  denotes the bias of  $j$ th the hidden neuron. The inputs are represented by  $X_j$ , while  $E$  indicates the output of the SLNN model. Eq.(3) represents the matrix of Eq. (2).

$$E^T = OS \quad (3)$$

where  $S = [S_1, S_2, \dots, S_h]^T$ .  $E^T$  denotes the transpose of matrix  $E$ .  $O$  denotes the output matrix of the hidden layer. It is calculated as follows:

$$O = \begin{bmatrix} f(w_1, b_1, X_1) & f(w_2, b_2, X_1) & \dots & f(w_h, b_h, X_1) \\ \dots & \dots & \dots & \dots \\ f(w_1, b_1, X_\beta) & f(w_2, b_2, X_\beta) & \dots & f(w_h, b_h, X_\beta) \end{bmatrix}_{\beta \times h} \quad (4)$$

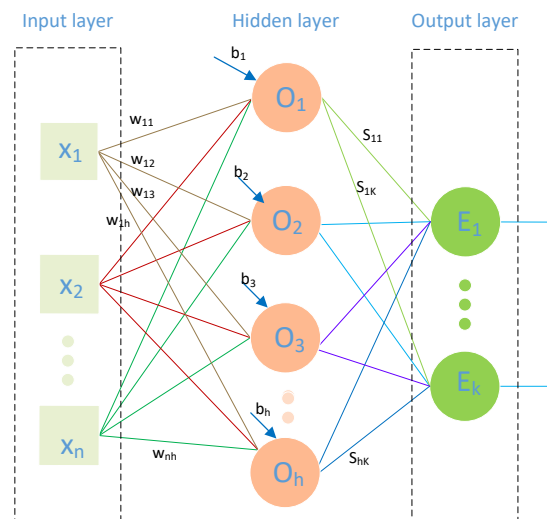


Figure 2. Basic Architecture of a Single Hidden Layer ELM Model

The principal objective of training is to reduce the error or variance of the ELM. The activation function must be infinitely differentiable in the standard ELM; nonetheless, ELM training results in the determination of the output weight ( $S$ ) by improving the least-squares function as specified in Eq. (5). The corresponding output weights are calculated analytically using the Moore-Penrose generalized inverse, as implemented in ELM (see Eq. (6)), rather than through iterative adjustment.

$$\min_S ||OS - E^T|| \quad (5)$$

$$\hat{E} = O^\dagger E^T \quad (6)$$

where  $O^\dagger$  denotes the generalized Moore-Penrose inverse of the  $O$  matrix.

### 3.3.2. Optimization Methodology

Metaheuristic optimization techniques can be used to optimize the weights and biases of the hidden layer in ELM, rather than relying on random assignment of their values. The integration of optimization aims to boost the accuracy and generalization ability of the ELM, especially in cases where the random determination of hidden layer parameters in might affect their ability to reach the desired solutions. A metaheuristic optimizer (e.g., PSO, GA) can be used to find the best or near-optimal values

for the ELM's input-to-hidden layer weights and hidden layer biases. After seeing the optimized values of the hidden-layer parameters, the ELM model proceeds with its standard training procedure, which involves computing the hidden-layer outputs and then using the Moore-Penrose inverse to allocate the hidden-to-output-layer weights.

To optimize the weights and biases of the hidden layer in ELM using a meta-heuristic algorithm, two main design aspects must be considered. Firstly, the design of the search agent, which solves the ELM's parameter fine-tuning problem. Then, the choice of an appropriate objective function for assessing the quality of the generated solutions (weights and biases).

### Solution Encoding

To optimize the ELM, we encode all input weights and hidden biases into a single continuous vector  $\mathbf{z} \in \mathbb{R}^D$ , where  $D$  is the total number of trainable ELM parameters. For a given candidate  $\mathbf{z}$ , the hidden-layer matrix  $H(\mathbf{z})$  is constructed, the output weights  $\beta(\mathbf{z})$  are computed in closed form, and the resulting model is evaluated on the training data. In all experiments, each component of  $\mathbf{z}$  is constrained to lie in the interval  $[-1, 1]$ , which defines the lower and upper bounds of the search space for the metaheuristics. Figure 3 depicts the encoding schema of the search agent. The search agent has  $D$  components based on the number of input weights and the number of hidden biases, as shown in Eq. (7):

$$D = R \times Y + Y \quad (7)$$

where  $R$  denotes the number of weights, and  $Y$  refers to the number of hidden biases.

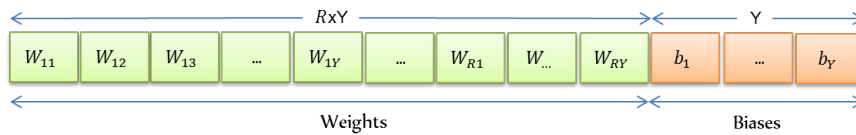


Figure 3. Search Agent Structure.

### Objective Function

The fitness (objective) function used by all metaheuristics is the average binary cross-entropy loss on the training set:

$$\mathcal{L}(\mathbf{z}) = -\frac{1}{N} \sum_{i=1}^N [y_i \log p_i(\mathbf{z}) + (1 - y_i) \log(1 - p_i(\mathbf{z}))], \quad (8)$$

where  $N$  is the number of training samples,  $y_i \in \{0, 1\}$  is the true label of sample  $i$ , and  $p_i(\mathbf{z})$  is the predicted probability of the positive class produced by the ELM with parameters  $\mathbf{z}$ .

The goal of each metaheuristic is to find the parameter vector that minimizes the training loss:

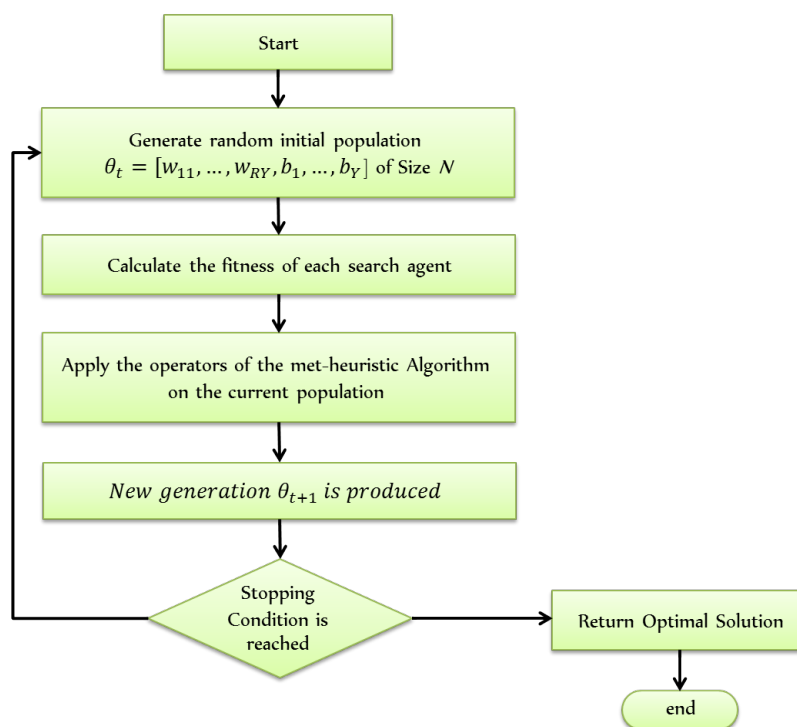
$$\mathbf{z}^* = \arg \min_{\mathbf{z} \in \mathbb{R}^D} \mathcal{L}(\mathbf{z}). \quad (9)$$

This common objective allows a fair comparison of different metaheuristics under the same training criterion. At the same time, the final generalization ability is assessed on the separate test sets as described in Section 3.4.

#### 3.3.3. Integration of Metaheuristics with ELM

The integration of metaheuristics with ELM follows a unified procedure. Each algorithm starts by generating an initial population of candidate vectors  $\mathbf{z}$ . In each iteration, the current population is decoded into ELM parameters, the cross-entropy loss  $L(\mathbf{z})$  is computed for each candidate, and the metaheuristic's update rules are applied to produce a new population. This process is repeated for a fixed number of iterations (100), with population size 30 in all experiments. At the end of the run, the candidate with the best fitness is selected, and the corresponding ELM (with its optimized weights and

biases) is used as the final classifier for that run. The inference phase remains as efficient as a standard ELM, since only a single forward pass through the optimized network is required for prediction. Figure 4 depicts the general flowchart of applying any metaheuristic algorithm for optimizing the parameters of the ELM model.



**Figure 4.** Integrating ELM model with meta-heuristic Algorithm.

### 3.4. Experimental Setup and Evaluation Protocol

This subsection describes how the datasets were split, how the baseline and metaheuristic-based models were configured, and how the evaluation procedure was carried out.

#### 3.4.1. Data Splitting and Repeated Experiments

For each dataset, we used a stratified train–test split to preserve the original class proportions, with 80% of the samples for training and 20% for testing in each run. All experiments were repeated 20 times with different random seeds, and the same splits were reused across all models within a run to ensure fair comparison. Feature scaling (z-score standardization) was applied only to the training set and then to the corresponding test set; no resampling or data augmentation methods were used.

#### 3.4.2. Baseline Classifiers and Hyperparameters

We evaluated eight baseline models: logistic regression (LR), k-nearest neighbours (KNN), decision tree (DT), random forest (RF), support vector machine (SVM) with Radial Basis Function (RBF), multilayer perceptron (MLP), XGBoost (XGB), and a standard ELM classifier. All models were implemented using scikit-learn and XGBoost libraries, except ELM, which was implemented via the Intelligent Metaheuristic-based ELM (IntelELM) framework [45,46]. Hyperparameters were set to standard recommended values and lightly tuned, then kept fixed across all runs; probability outputs were enabled where needed to compute ROC-AUC. The primary internal hyperparameters are summarized in Table 3. It is worth mentioning that only parameters explicitly set in the implementation are reported, while all other options follow the default values of the corresponding libraries.

**Table 3.** Baseline classifiers and main hyperparameters used in the experiments.

Model	Key hyperparameters
ELM	layer_sizes = (50); act_name = relu
LR (LogReg)	max_iter = 200
RF	n_estimators = 20
SVM (RBF)	kernel = rbf; probability = True
XGB	n_estimators = 20; max_depth = 4; learning_rate = 0.05; subsample = 0.8; colsample_bytree = 0.8; eval_metric = logloss
MLP	hidden_layer_sizes = (100,); activation = relu; solver = adam; max_iter = 200
KNN	n_neighbors = 5; weights = distance; metric = minkowski
DT	max_depth = None

### 3.4.3. Metaheuristic-Optimized ELM Configuration

For the optimized models, we used IntelELM to configure a single-hidden-layer ELM with fixed architecture (number of hidden neurons and activation function) across datasets. Eleven metaheuristic algorithms were employed to optimize the ELM weights and biases. These algorithms include a diverse set of basic and advanced metaheuristics drawn from different families: an evolutionary-based method, Genetic Algorithm (GA) [47]; a math-based optimizer, RUNge Kutta optimizer (RUN) [48]; a physics-based method, Modified Equilibrium Optimizer (MEO) [49]; and a rich group of swarm-based algorithms, including Comprehensive Learning Particle Swarm Optimization (CL-PSO) [50], Hybrid Improved Whale Optimization Algorithm (HI-WOA) [51], standard Grey Wolf Optimizer (GWO) [52], Hunger Games Search (HGS) [53], Harris Hawks Optimization (HHO) [54], Sea-Horse Optimization (SeaHO) [55], Mountain Gazelle Optimizer (MGO) [56], and the hybrid Grey Wolf–Whale Optimization Algorithm (GWO-WOA) [57].

Each optimizer was run for 100 iterations with a population size of 30, and the cross-entropy loss on the training data was used as the objective function. Algorithm-specific parameters were set to either IntelELM defaults [45] or values suggested in the original papers and were kept constant across all runs.

### 3.4.4. Evaluation Measures

We evaluated the models using standard classification quality metrics and computational time. All metrics were computed on the training and test sets and then averaged across repeated runs.

#### Confusion matrix for binary diagnosis

For both datasets, the problem is binary:

- Positive class: patient with obstructive sleep apnea or sleep-disordered breathing (OSA/SDB present).
- Negative class: patient without OSA/SDB (no sleep apnea).

For a given test set, predictions can be summarized in a confusion matrix as shown in Figure 5, where:

- True Positive (TP): correctly predicted OSA/SDB cases.
- True Negative (TN): correctly predicted non-OSA/non-SDB cases.
- False Positive (FP): predicted OSA/SDB, but the patient is actually non-OSA/non-SDB.
- False Negative (FN): predicted non-OSA/non-SDB, but the patient actually has OSA/SDB.

Actual	Predicted	
	OSA/SDB (+)	No OSA/SDB (-)
OSA/SDB (+)	TP	FN
No OSA/SDB (-)	FP	TN

**Figure 5.** Schematic confusion matrix for the binary diagnosis task, where TP, TN, FP, and FN denote true positives, true negatives, false positives, and false negatives, respectively.

### Classification Quality Metrics

From the confusion matrix, we compute the following measures.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}. \quad (10)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \quad (11)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \quad (12)$$

$$\text{F1} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (13)$$

We also report the receiver operating characteristic area under the curve (ROC-AUC). ROC-AUC summarizes the trade-off between true positive rate and false positive rate across all possible decision thresholds. Higher AUC values indicate a better ability to distinguish between OSA/SDB and non-OSA/non-SDB patients.

### Computational Time

To assess efficiency, we measure training time and testing time for each model. Training time is the time required to fit the model on the training data (including metaheuristic optimization steps when used). In contrast, testing time is the time needed to generate predictions for the test set using the trained model. Both times are recorded for every run and then summarized by their mean and standard deviation. This allows us to compare not only predictive performance but also computational cost across baseline and metaheuristic-optimized ELM models.

#### 3.4.5. Environment and Tools

All experiments were implemented in a Python framework developed based on IntelELM and other standard scientific libraries. The experiments were conducted on a Windows 64-bit system with an 11th Gen Intel<sup>®</sup> Core<sup>™</sup> i7-1165G7 CPU @ 2.80 GHz and 16 GB of RAM. All runs were performed on the same machine and under the same software environment to ensure consistent timing measurements and reproducible results.

## 4. Experimental Results

This section presents the empirical evaluation of the proposed framework in two phases. First, in Section 4.1, we compare and analyze a set of baseline ML models, including the standard ELM, to assess their ability to detect obstructive sleep apnea and sleep-disordered breathing from the two datasets. In the second phase, we focus on improving ELM by optimizing its weights and biases using different metaheuristic algorithms (Section 4.2), and we examine how these optimized variants affect classification performance, generalization, and computational cost.

### 4.1. Results and Analysis of Baseline Models

This section focuses on the analysis of baseline classifiers applied to the OSA dataset. It presents and discusses the test-set performance in Table 4 and Figure 6, the training-set performance in Table

A1, and the generalization analysis in Table 5 and Figure 7. Together, these results show how eight models behave on the OSA data and how well they generalize from training to unseen cases.

#### 4.1.1. Results on OSA Dataset

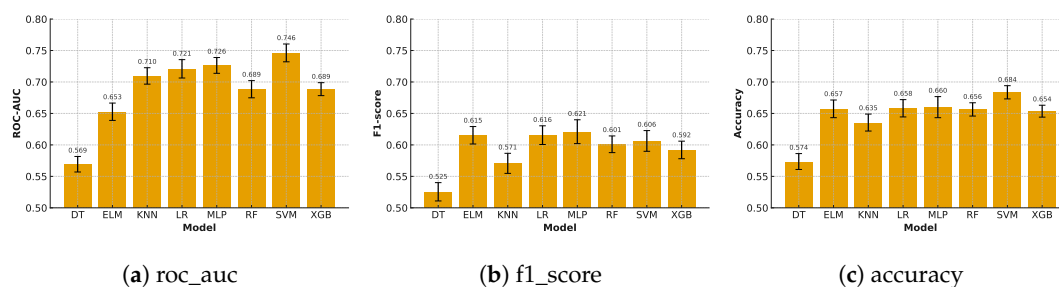
##### Testing Performance

As shown in Table 4 and Figure 6, there are noticeable differences in the effectiveness of the compared models. The DT is the weakest baseline, with an ROC-AUC of 0.5693, an F1 of 0.5255, and an accuracy of 0.5736. On the other hand, among the stronger models, SVM and MLP stand out. SVM achieves the highest ROC-AUC (0.7462) and accuracy (0.6836), while MLP obtains the best F1-score (0.6210) and recall (0.6160). Compared with DT as a baseline, SVM improves ROC-AUC by about 31% and accuracy by about 19%, while MLP improves ROC-AUC by about 28% and accuracy by about 15%. The F1-score gains are also notable: around 18% for MLP and 15% for SVM relative to DT. These gains indicate a clear benefit of more advanced models for OSA risk prediction.

**Table 4.** Test-set classification performance of baseline models on the OSA dataset. Mean rank is the average ranking of each model across all evaluation metrics, with lower values indicating better overall performance.

model	Measure	accuracy	precision	recall	f1	roc_auc	mean rank
DT	Avg	0.5736	0.5345	0.5220	0.5255	0.5693	8.0
	Std	0.0564	0.0684	0.0794	0.0651	0.0552	
ELM	Avg	0.6573	0.6332	0.6020	0.6153	0.6527	4.4
	Std	0.0628	0.0751	0.0680	0.0620	0.0616	
KNN	Avg	0.6355	0.6175	0.5340	0.5707	0.7096	6.4
	Std	0.0603	0.0865	0.0749	0.0713	0.0584	
LR	Avg	0.6582	0.6331	0.6040	0.6155	0.7209	3.2
	Std	0.0615	0.0720	0.0850	0.0664	0.0653	
MLP	Avg	0.6600	0.6334	<b>0.6160</b>	<b>0.6210</b>	0.7263	2.0
	Std	0.0749	0.0943	0.1025	0.0847	0.0561	
RF	Avg	0.6564	0.6423	0.5740	0.6010	0.6885	4.4
	Std	0.0472	0.0785	0.0902	0.0593	0.0611	
SVM	Avg	<b>0.6836</b>	<b>0.6973</b>	0.5440	0.6063	<b>0.7462</b>	2.6
	Std	0.0470	0.0758	0.1017	0.0736	0.0633	
XGB	Avg	0.6536	0.6393	0.5600	0.5920	0.6886	5.0
	Std	0.0423	0.0609	0.0954	0.0624	0.0460	

Considering the other models, LR, ELM, RF, and XGB form a middle group. LR has balanced performance with accuracy 0.6582, F1-score 0.6155, and ROC-AUC 0.7209, which is close to MLP and SVM but slightly lower. ELM and RF have similar accuracies (around 0.656–0.657) but somewhat lower AUC values (0.6527 and 0.6885). KNN is weaker than these models, mainly due to lower recall (0.5340). The mean rank values, which denote the average ranking of each model across all evaluation metrics, confirm these findings: MLP has the best overall rank (2.0), followed by SVM (2.6), LR (3.2), ELM and RF (4.4), XGB (5.0), KNN (6.4), while DT has the worst rank (8.0).



**Figure 6.** Test performance of the eight classifiers: (a) ROC-AUC, (b) F1-score, and (c) accuracy. Bars represent the mean test value over 20 runs, and the vertical lines on each bar show  $\pm 1$  standard error (SE). The SE is computed as  $SE = s / \sqrt{n}$ , where  $s$  is the std of the metric over the 20 runs ( $n = 20$ ).

To demonstrate the variability across the 20 runs and how much the performance fluctuates from run to run, the standard error (SE) lines are visualized in Figure 6. Consistently, SEs are relatively small for all models, which means the results are stable under repeated random splits. However, variability is slightly larger for MLP.

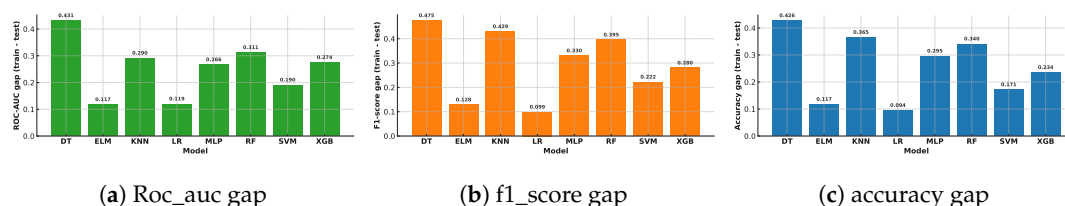
### Training Performance and Overfitting

To discuss overfitting and generalization, quantifying gaps between training and testing performance for each metric are reported in Table A1 and visualized in Figure 7. From Table A1, we observe that several models fit the training data almost perfectly. DT, KNN, and RF reach training accuracies near 1.0 and ROC-AUC close to 1.0 with zero or very low standard deviation. MLP and XGB also report very high training scores (e.g., MLP training AUC 0.9923, XGB 0.9627). In contrast, ELM, LR, and, to a lesser extent, SVM keep more moderate training scores, which indicates better regularization or less capacity to memorize the data.

For further clarification, Table 5 and Figure 7 summarize these differences through the generalization gap  $\Delta = \text{train} - \text{test}$ . DT, KNN, RF, and MLP show large gaps in all metrics. For example, DT has an AUC gap of 0.4307 and an accuracy gap of 0.4264, meaning that more than 40% points of its training performance are lost on the test set. KNN and RF show similar patterns, with AUC gaps around 0.29–0.31 and F1 gaps around 0.39–0.43. MLP also suffers from notable overfitting, with F1 and AUC gaps of around 0.33 and 0.27. These models are powerful but tend to memorize the training set, especially in a relatively small and somewhat imbalanced sample.

**Table 5.** Training and test performance of baseline classifiers on the OSA dataset. It presents the generalization gap ( $\Delta$ ) for accuracy, F1-score, and ROC-AUC, where  $\Delta = \text{train} - \text{test}$

Model	accuracy			f1-score			roc_auc		
	train	test	$\Delta$	train	test	$\Delta$	train	test	$\Delta$
DT	1.0000	0.5736	0.4264	1.0000	0.5255	0.4745	1.0000	0.5693	0.4307
ELM	0.7740	0.6573	0.1167	0.7435	0.6153	0.1282	0.7696	0.6527	0.1169
KNN	1.0000	0.6355	0.3645	1.0000	0.5707	0.4293	1.0000	0.7096	0.2904
LR	0.7521	0.6582	0.0939	0.7142	0.6155	0.0987	0.8402	0.7209	0.1193
MLP	0.9553	0.6600	0.2953	0.9509	0.6210	0.3299	0.9923	0.7263	0.2660
RF	0.9966	0.6564	0.3402	0.9962	0.6010	0.3952	0.9999	0.6885	0.3114
SVM	0.8543	0.6836	0.1707	0.8282	0.6063	0.2219	0.9364	0.7462	0.1902
XGB	0.8879	0.6536	0.2343	0.8719	0.5920	0.2799	0.9627	0.6886	0.2741



**Figure 7.** Generalization gaps (train – test) for baseline classifiers on the OSA dataset: (a) ROCAUC gap, (b) F1\_score gap, and (c) accuracy gap.

In contrast, LR and ELM have the fewest gaps. LR shows an accuracy gap of 0.0939 and an AUC gap of 0.1193; ELM has a similar AUC gap (0.1169) and a slightly larger accuracy gap (0.1167). SVM lies between these two groups, with moderate gaps (AUC gap: 0.1902; accuracy gap: 0.1707). Figure 7 makes this pattern clear: LR and ELM have the shortest bars, SVM sits in the middle, while DT, KNN, RF, MLP, and XGB have much taller bars. This means that LR and ELM offer the best balance between test accuracy and robustness, while SVM trades a bit more overfitting for the highest test-set AUC.

From an OSA screening perspective, this trade-off is essential. SVMs and MLPs achieve the best raw test performance and can be attractive when the focus is on maximizing discrimination. However, their significant gaps suggest that their decisions are more sensitive to the specific training sample. LR

and ELM provide slightly lower test metrics but more stable generalization, which may be safer when the model is deployed in new clinical settings.

### *Training and Testing Time*

In terms of efficiency, Table 6 reports the average training time (time needed to fit the model on the training data) and testing time (time needed to generate predictions for the test set) over the 20 runs, together with their standard deviations and an overall mean rank (lower is better). We can observe that ELM is the fastest model, with the lowest training time (0.0046 s) and testing time (0.0001 s). DT and LR are also swift, with training and testing times of a few milliseconds. In contrast, MLP, KNN, RF, and SVM are noticeably slower, especially during training for MLP (0.3751 s on average) and during testing for KNN, RF, and SVM, which take longer to produce predictions. Overall, these results indicate that ELM offers a favorable trade-off between predictive performance and computational cost, which is essential for real-time or resource-constrained screening scenarios.

**Table 6.** Training and testing times (in seconds) of baseline classifiers on the OSA dataset; mean rank indicates the average efficiency ranking across models (lower values mean faster overall performance).

model	Measure	train_time	test_time	mean rank
DT	Avg	0.0063	0.0005	2.0
	Std	0.0019	0.0003	
ELM	Avg	<b>0.0046</b>	<b>0.0001</b>	1.0
	Std	0.0054	0.0001	
KNN	Avg	0.1295	0.0051	6.5
	Std	0.5372	0.0013	
LR	Avg	0.0076	0.0006	3.0
	Std	0.0037	0.0004	
MLP	Avg	0.3751	0.0009	6.0
	Std	0.0939	0.0004	
RF	Avg	0.0622	0.0066	6.5
	Std	0.0218	0.0027	
SVM	Avg	0.0507	0.0070	6.5
	Std	0.0178	0.0033	
XGB	Avg	0.0286	0.0014	4.5
	Std	0.0065	0.0006	

#### 4.1.2. Results on SDB Dataset

For the SDB dataset, Table 7 shows that most models achieve relatively high accuracy (0.72–0.76), very high recall (often above 0.9), and strong F1 Scores, but their ROC-AUC values remain low, mainly in the 0.49–0.58 range. This pattern reflects the strong class imbalance (only 119 of 500 samples are “no sleep apnea”): the classifiers tend to label most cases as having sleep apnea, boosting recall and F1 but offering poor discrimination between the positive and negative classes. XGB and MLP provide the best overall trade-off according to the mean rank, with XGB achieving the highest AUC (0.5804) and tied-best accuracy and F1 with SVM (0.7600 and 0.8636).

Training results in Table A2 confirm that several models almost perfectly fit the training data. When these training results are contrasted with the much lower test AUC values, it becomes clear that many models heavily overfit to the imbalanced SDB dataset: they learn the majority class very well but fail to generalize to the minority “no sleep apnea” cases. Overall, the SDB results highlight that, under severe imbalance, high recall and F1 scores alone can be misleading, and ROC-AUC, together with training–testing differences, should be used to assess how reliable the models really are.

**Table 7.** Test classification performance of baseline models on the SDB dataset

model	Measure	accuracy	precision	recall	f1	roc_auc	mean rank
DT	Avg	0.6345	0.7695	0.7408	0.7542	0.5194	6.4
	Std	0.0467	0.0258	0.0544	0.0369	0.0521	
ELM	Avg	0.7410	0.7650	0.9513	0.8479	0.5132	4.8
	Std	0.0295	0.0116	0.0355	0.0194	0.0301	
KNN	Avg	0.7190	0.7629	0.9145	0.8317	0.5426	5.6
	Std	0.0279	0.0142	0.0275	0.0176	0.0508	
LR	Avg	0.7590	0.7598	0.9987	0.8630	0.5380	4.2
	Std	0.0031	0.0007	0.0040	0.0020	0.0527	
MLP	Avg	0.7460	<b>0.7697</b>	0.9500	0.8502	0.5430	3.2
	Std	0.0237	0.0094	0.0327	0.0160	0.0582	
RF	Avg	0.7300	0.7625	0.9368	0.8405	0.5329	5.6
	Std	0.0192	0.0104	0.0288	0.0130	0.0532	
SVM	Avg	<b>0.7600</b>	0.7600	<b>1.0000</b>	<b>0.8636</b>	0.4855	3.4
	Std	0.0000	0.0000	0.0000	0.0000	0.0568	
XGB	Avg	<b>0.7600</b>	0.7600	<b>1.0000</b>	<b>0.8636</b>	<b>0.5804</b>	2.0
	Std	0.0000	0.0000	0.0000	0.0000	0.0511	

Overall, the results on the SDB dataset are broadly consistent with the findings on the OSA dataset, but the impact of class imbalance is much more substantial. In both datasets, simple models such as DT perform worst, while more advanced models (LR, MLP, SVM, XGB) achieve the highest test accuracy and F1 scores, confirming that the relative ranking of the classifiers is stable across data sources. However, compared with OSA, all models achieve lower ROC-AUC on SDB, despite higher recall and F1, indicating that the strong skew towards the apnea class makes it harder to distinguish the minority “no apnea” cases. The training results also follow the same pattern: DT, KNN, and RF almost memorize the training set; MLP and XGB achieve very high training scores, while ELM and LR show more moderate training performance, again suggesting better generalization.

#### 4.2. Results of Metaheuristic-Optimized ELM

After analysing the baseline results in Section 4.1, where ELM was compared with several conventional classifiers on both the OSA and SDB datasets, we now focus on improving ELM itself using metaheuristic optimization. The goal is to refine the ELM parameters (weights and biases) to enhance predictive performance and robustness, especially under the imbalanced and limited-sample conditions observed in the datasets.

To this end, we evaluate a diverse set of basic and advanced metaheuristic algorithms drawn from different families: an evolutionary-based method, GA; a math-based optimizer, RUN; a physics-based method, MEO; and a rich group of swarm-based algorithms, including CL-PSO, HI-WOA, GWO, HGS, HHO, SeaHO, MGO, and the GWO-WOA. Each metaheuristic is executed for 20 independent runs, with 100 iterations and a population size of 30, under the same experimental environment to ensure a fair comparison.

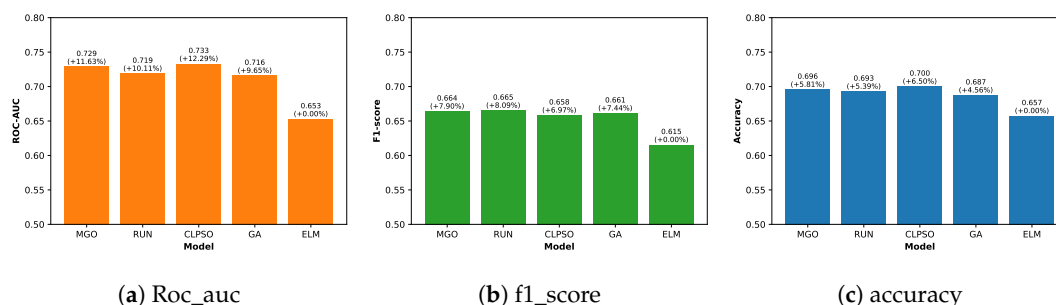
##### 4.2.1. Results on OSA Dataset

Table 8 reports the test-set classification performance of metaheuristic-optimized ELM models on the OSA dataset. The last row reports the baseline (non-optimized) ELM. It is clear from Table 8 and Figure 8 that optimizing ELM with metaheuristics consistently improves test performance compared with the standard ELM. For example, CL-PSO, MGO, RUN, and GA all raise test accuracy from 0.6573 for plain ELM to about 0.69–0.70 (roughly 5–7% improvement) and increase F1 from 0.6153 to about 0.66 (around 7–8% gain). Their gains in ROC-AUC are more pronounced: CL-PSO and MGO reach 0.7329 and 0.7286, respectively, compared with 0.6527 for ELM, corresponding to an improvement of about 11–12%. Even the weaker optimizers, such as HI-WOA and HHO, still provide higher AUC than the non-optimized ELM, although their accuracy and F1 remain closer to the baseline. The mean rank values confirm these trends: MGO, RUN, and CL-PSO achieve the best overall ranks (2.2, 2.6, and 3.0), while the plain ELM and HI-WOA appear among the least competitive methods. Overall, these

results indicate that metaheuristic optimization provides clear and consistent benefits for ELM on the OSA dataset, especially in terms of discrimination ability, as measured by ROC-AUC.

**Table 8.** Test classification performance of metaheuristic-optimized ELM models on the OSA dataset. ELM denotes the baseline (non-optimized) version.

model	Measure	accuracy	precision	recall	f1	roc_auc	mean rank
CLPSO	Avg	<b>0.7000</b>	<b>0.6798</b>	0.6420	0.6582	<b>0.7329</b>	3.0
	Std	0.0423	0.0457	0.0904	0.0602	0.0382	
GA	Avg	0.6873	0.6541	<b>0.6740</b>	0.6611	0.7157	4.0
	Std	0.0415	0.0513	0.0771	0.0462	0.0408	
GWO	Avg	0.6818	0.6500	0.6560	0.6515	0.7164	5.8
	Std	0.0492	0.0561	0.0716	0.0552	0.0519	
GWOWOA	Avg	0.6827	0.6573	0.6460	0.6483	0.7223	4.8
	Std	0.0406	0.0556	0.0737	0.0446	0.0381	
HGS	Avg	0.6818	0.6533	0.6480	0.6468	0.7074	6.6
	Std	0.0499	0.0561	0.0968	0.0650	0.0485	
HHO	Avg	0.6745	0.6464	0.6300	0.6343	0.7066	9.6
	Std	0.0627	0.0687	0.1153	0.0812	0.0505	
HIWOA	Avg	0.6573	0.6304	0.5960	0.6112	0.6839	11.6
	Std	0.0597	0.0705	0.0860	0.0733	0.0550	
MEO	Avg	0.6755	0.6506	0.6220	0.6333	0.7214	8.0
	Std	0.0418	0.0528	0.0846	0.0589	0.0559	
MGO	Avg	0.6955	0.6689	0.6640	0.6639	0.7286	2.2
	Std	0.0368	0.0497	0.0679	0.0419	0.0478	
RUN	Avg	0.6927	0.6590	<b>0.6740</b>	<b>0.6651</b>	0.7187	2.6
	Std	0.0504	0.0547	0.0771	0.0581	0.0539	
SeaHO	Avg	0.6791	0.6486	0.6440	0.6445	0.7111	8.0
	Std	0.0614	0.0726	0.0917	0.0744	0.0594	
ELM	Avg	0.6573	0.6332	0.6020	0.6153	0.6527	11.2
	Std	0.0628	0.0751	0.0680	0.0620	0.0616	



**Figure 8.** Comparison of the four best-performing metaheuristic-based models (MGO, RUN, CLPSO, GA) against ELM on the OSA dataset in terms of (a) ROC-AUC, (b) F1-score, and (c) accuracy. Bars show the actual metric values, and the labels above each bar indicate the corresponding percentage improvement (or decline) relative to ELM.

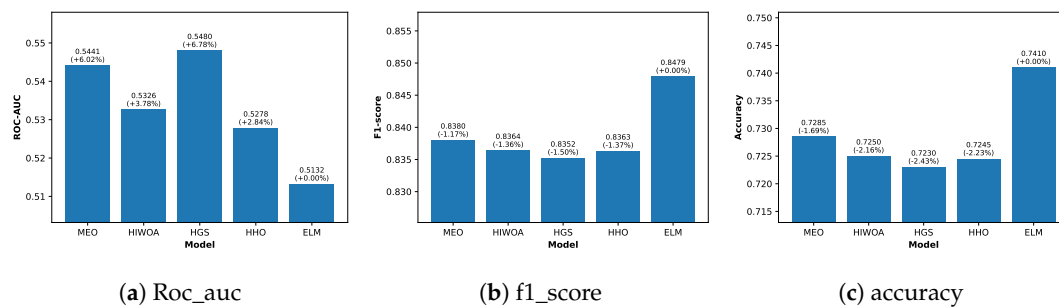
#### 4.2.2. Results on SDB Dataset

Table 9 presents the test performance of metaheuristic-optimized ELM models on the SDB dataset. Overall, almost all optimized variants achieve similar or slightly lower accuracy and F1 Scores than the plain ELM, but several offer modest gains in ROC-AUC. For example, MEO, HGS, GWO, and SeaHO increase AUC from 0.5132 for ELM to about 0.544–0.548 (roughly 6–7% improvement), while their accuracies remain around 0.72–0.73 compared with 0.7410 for ELM. The mean rank values reflect this trade-off: ELM remains among the top methods due to its strong accuracy and F1, whereas MEO and HI-WOA rank well because they balance small drops in accuracy with improved discriminatory power on this highly imbalanced dataset.

**Table 9.** Test classification performance of metaheuristic-optimized ELM models on the SDB dataset.

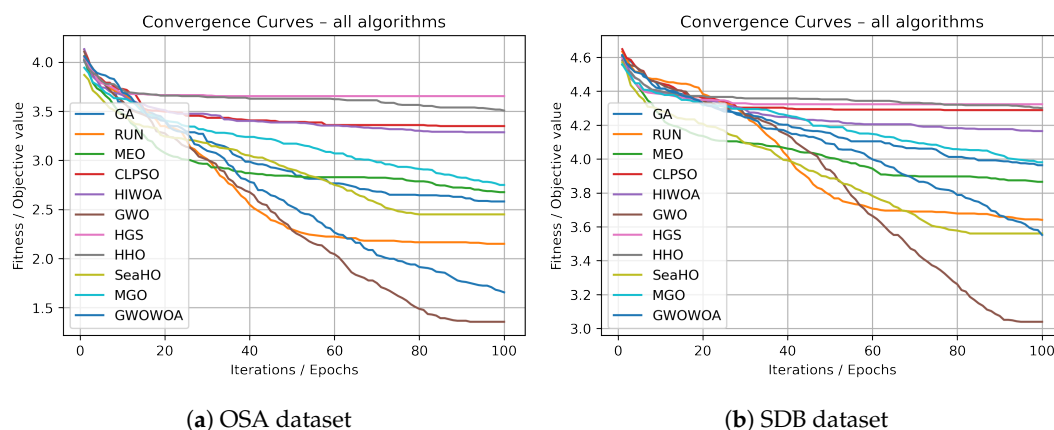
model	Measure	accuracy	precision	recall	f1	roc_auc	mean rank
CLPSO	Avg	0.7175	0.7579	0.9230	0.8323	0.4985	8.6
	Std	0.0183	0.0077	0.0215	0.0121	0.0516	
GA	Avg	0.7140	0.7649	0.9007	0.8271	0.5300	8.2
	Std	0.0190	0.0095	0.0251	0.0130	0.0571	
GWO	Avg	0.7120	0.7673	0.8914	0.8246	0.5454	7.2
	Std	0.0253	0.0134	0.0280	0.0168	0.0518	
GWOWOA	Avg	0.7180	<b>0.7681</b>	0.9013	0.8293	0.5404	6.0
	Std	0.0164	0.0106	0.0184	0.0104	0.0506	
HGS	Avg	0.7230	0.7624	0.9237	0.8352	<b>0.5480</b>	5.0
	Std	0.0187	0.0118	0.0207	0.0116	0.0493	
HHO	Avg	0.7245	0.7620	0.9270	0.8363	0.5278	5.6
	Std	0.0295	0.0143	0.0306	0.0189	0.0722	
HIWOA	Avg	0.7250	0.7631	0.9257	0.8364	0.5326	4.6
	Std	0.0164	0.0097	0.0231	0.0109	0.0447	
MEO	Avg	0.7285	0.7666	0.9243	0.8380	0.5441	3.0
	Std	0.0278	0.0152	0.0276	0.0174	0.0647	
MGO	Avg	0.6935	0.7545	0.8842	0.8139	0.5094	11.6
	Std	0.0303	0.0112	0.0404	0.0216	0.0667	
RUN	Avg	0.7165	0.7635	0.9086	0.8296	0.5215	8.0
	Std	0.0320	0.0174	0.0294	0.0201	0.0803	
SeaHO	Avg	0.7205	0.7648	0.9132	0.8323	0.5450	5.6
	Std	0.0190	0.0092	0.0223	0.0126	0.0329	
ELM	Avg	<b>0.7410</b>	0.7650	<b>0.9513</b>	<b>0.8479</b>	0.5132	3.4
	Std	0.0295	0.0116	0.0355	0.0194	0.0301	

For further illustration, Figure 9 focuses on the four best metaheuristic variants on SDB (MEO, HI-WOA, HGS, HHO) and compares them directly with ELM. We can observe that all four optimizers improve ROC-AUC by approximately 3–7% relative to ELM, but they lose about 1–2% in both accuracy and F1-score. This means that optimization slightly reduces the overall number of correctly classified samples but makes the classifier more sensitive to the minority “no sleep apnoea” class, which is reflected in better ranking quality. In a clinical screening context, this trade-off can be acceptable when the priority is to better distinguish risky patients from safe ones rather than maximizing raw accuracy.

**Figure 9.** Comparison of the four best-performing metaheuristic-based models (MEO, HIWOA, HGS, HHO) against ELM on SDB dataset

The convergence curves in Figure 10 demonstrate how each metaheuristic reduces the training fitness (cross-entropy loss) over 100 iterations on the OSA and SDB datasets. We can see that every metaheuristic reduces the training loss, but with apparent differences in efficiency and stability. GWO stands out with a smooth, almost monotonic descent and consistently reaches the lowest final fitness among the methods, while RUN, SeaHO, and MGO also attain strong but slightly higher minima. In contrast, algorithms such as HHO, HGS, CL-PSO, and HI-WOA exhibit early stagnation at relatively high loss values, which suggests weaker exploration and a higher risk of premature convergence to local optima. The curves further indicate that the SDB landscape is more challenging to optimize than the OSA landscape, with higher objective values. Overall, these convergence patterns confirm that some metaheuristics are more effective optimizers of the ELM objective. Still, they also highlight that lower training loss does not necessarily yield the best test AUC or F1, so convergence behaviour must be interpreted alongside generalization results.

Specifically, in our results, GWO often achieves the smallest cross-entropy values on the training data. However, methods such as GA or MGO sometimes obtain similar or better ROC-AUC and F1 on the test set (see Tables 8 and 9). This happens because the optimized objective is the training loss, while the primary evaluation criteria are based on classification quality, and the dataset, especially SDB, is imbalanced. Very aggressive minimization of loss can push the model toward solutions that fit majority-class patterns or noise, improving the training objective but harming generalization. For this reason, the convergence curves should be read together with the test metrics: they show that the optimizer is working, but the final choice of metaheuristic must be guided by how well the corresponding ELM model performs on unseen cases, not only by how low the training loss becomes.



**Figure 10.** Convergence behavior of the metaheuristic algorithms on (a) the OSA dataset and (b) the SDB dataset. Each curve shows the mean fitness value over 20 independent runs across 100 training epochs.

#### 4.2.3. Computational Cost of Optimization

Table 10 reports the training times of all metaheuristic-based ELM variants on both datasets and highlights the computational cost of optimization. Plain ELM is extremely fast (on the order of milliseconds), while the optimized versions require 10–20 seconds for methods such as GA, GWO, HGS, and SeaHO, and 40–50 seconds for CL-PSO and MGO. These results confirm that metaheuristic optimization is best viewed as an offline tuning step. Once an optimized set of weights and biases is found, the deployed ELM keeps the low prediction time of the baseline model, while benefiting from the improved classification performance reported in Tables 8 and 9.

**Table 10.** Training times (in seconds) of baseline ELM and metaheuristic-based ELM versions on OSA and SDB datasets.

model	Measure	OSA	SDB
CLPSO	Avg	56.7140	30.8554
	Std	1.4979	3.4613
GA	Avg	12.2153	13.8607
	Std	0.3092	3.4478
GWO	Avg	11.4065	10.5191
	Std	0.2875	0.3267
GWOWOA	Avg	11.4524	14.6931
	Std	0.5361	2.7607
HGS	Avg	10.6527	10.5610
	Std	0.5867	0.9460
HHO	Avg	18.6361	18.7242
	Std	0.7000	1.3378
HIWOA	Avg	12.9272	12.9110
	Std	0.8009	0.7845
MEO	Avg	17.5411	22.7018
	Std	0.5319	4.3979
MGO	Avg	40.1385	46.5233
	Std	1.1942	4.0455
RUN	Avg	20.2531	24.6024
	Std	0.7883	9.1322
SeaHO	Avg	17.4395	17.7751
	Std	0.5797	2.9496
ELM	Avg	<b>0.0046</b>	<b>0.0911</b>
	Std	0.0054	0.2988

### 4.3. Discussion and Limitations

Taken together, the results across both datasets reveal a recurring trade-off between predictive performance, generalization, and computational cost. On the OSA dataset, SVM and MLP delivered the strongest test ROC-AUC and F1 scores among the baselines, yet their training scores sat near 1.0, leaving sizable train–test gaps. This pattern is the classic signature of overfitting in small clinical samples. LR and the plain ELM tell a different story: their test metrics are slightly lower, but the gap to training performance is much narrower, which points to more stable behaviour when the model is later applied to new patients. The picture on the SDB dataset is shaped by class imbalance. Accuracy, recall, and F1 all look reassuring at first glance, but ROC-AUC stays modest, a reminder that when one class dominates, aggregate metrics can mask poor discrimination. Any honest assessment under such skew has to weigh ranking ability and train–test consistency, not headline accuracy.

Optimizing ELM with metaheuristics sharpens this picture. On OSA, MGO, RUN, and CL-PSO produced steady improvements across accuracy, F1, and ROC-AUC, with AUC gains of roughly 10–12% over the plain ELM and no added cost at inference time. The SDB results were less dramatic. Most optimized variants nudged ROC-AUC upward by a few percentage points but gave back small amounts of accuracy and F1 in return — the optimizers were evidently shifting attention toward the minority “no apnea” cases, but the imbalance and modest sample size kept these gains constrained. The convergence curves reinforce this reading. GWO drove the training loss down most aggressively, yet the highest test ROC-AUC and F1 sometimes came from methods that settled at higher loss values. Pushing the training objective too hard, in other words, did not always pay off on unseen data.

Read together, these observations point to a practical conclusion: a properly tuned ELM strikes a workable balance between accuracy, robustness, and computational cost for early OSA screening from demographic and clinical inputs. Metaheuristic optimization adds the most value when the data are reasonably balanced, as in the OSA dataset. Under heavy imbalance, as in the SDB case, it mainly buys better discrimination at the margin rather than large gains in aggregate accuracy.

#### 4.3.1. Limitations of the Study

Despite the acceptable results, this study has several limitations that should be acknowledged when interpreting the results:

- The experiments rely on two datasets from a specific clinical context, each with a relatively small sample size. Accordingly, the generality of the conclusions to other populations or acquisition protocols is not guaranteed.
- The SDB dataset is highly imbalanced, and although appropriate metrics are used, residual bias toward the majority class may still affect the reported performance.
- The analysis focuses on global performance metrics. However, model interpretability and feature importance, which are essential for clinical adoption, were not explored in this study.
- The metaheuristic algorithms, the ELM architecture, and the investigated ML models were configured using reasonable but fixed hyperparameter settings. More exhaustive hyperparameter tuning could further improve performance or change the relative rankings of the methods.
- Metaheuristic optimization introduces a non-negligible offline training cost, which may limit its use when frequent re-training is required or when computational resources are very constrained

## 5. Conclusion and Future Work

This study investigated a lightweight, metaheuristic-optimized Extreme Learning Machine (ELM) framework for early diagnosis of obstructive sleep apnea (OSA) and sleep-disordered breathing (SDB) using demographic/clinical predictors. After standardizing features with a strict train–test separation, we evaluated eight baseline classifiers and then optimized the ELM hidden-layer weights and biases using eleven metaheuristic algorithms. On the OSA dataset, baseline results showed that SVM and MLP achieved the strongest overall discrimination among conventional models (e.g., SVM ROC-AUC = 0.7462, MLP ROC-AUC = 0.7263). At the same time, the plain ELM provided competitive but lower

test performance (accuracy = 0.6573, F1 = 0.6153, ROC-AUC = 0.6527). Importantly, metaheuristic tuning produced clear and consistent gains for ELM on OSA: for example, CL-PSO improved test ROC-AUC to 0.7329 and accuracy to 0.7000, while RUN improved F1 to 0.6651 (from 0.6153). These results confirm that optimizing the hidden-layer parameters can significantly enhance ELM's discrimination and overall stability for early screening tasks on moderately balanced clinical data.

The SDB dataset, with its heavy class skew, told a different story. Baseline models produced high accuracy and F1 figures but only modest ROC-AUC values, and the plain ELM ranked among the stronger baselines on accuracy and F1 (0.7410 and 0.8479, respectively) while its ROC-AUC remained limited at 0.5132. Here, metaheuristic optimization mostly shifted the balance toward better discrimination rather than higher aggregate scores. HGS, for example, lifted ROC-AUC to 0.5480 and MEO to 0.5441, even as accuracy slipped a little (MEO at 0.7285 against 0.7410 for the plain ELM). The takeaway is straightforward: under severe imbalance, optimization can sharpen the model's ability to separate minority cases, but those AUC gains tend to come at a small cost in accuracy and F1. The timing analysis adds one more practical note. Optimization is best treated as an offline step. Training the plain ELM takes around 0.0046 s on OSA, while the optimized variants need roughly 10–20 s for most algorithms and as much as 40–56 s for the heavier ones, yet none of this affects the low-cost inference once the optimized weights are in place.

Future work should focus on improving clinical readiness and generalization. First, broader validation is needed across larger, more diverse cohorts (including external sites) to confirm robustness across populations and acquisition settings, especially given the limited sample sizes used here. Second, the SDB setting requires stronger imbalance-aware learning (e.g., cost-sensitive objectives, calibrated threshold selection, or resampling strategies) to reduce residual bias toward the majority class while preserving AUC gains. Third, interpretability should be incorporated through feature importance and explanation methods to support clinical adoption and to verify that optimized ELM decisions align with medical knowledge. Fourth, a more systematic exploration of hyperparameters (ELM architecture choices and optimizer settings) and potentially multi-objective optimization (optimizing AUC/F1 and calibration jointly, rather than only the training loss) may further improve the ranking and consistency of the optimized model.

**Acknowledgments:** During the preparation of this manuscript, the author(s) used Claude (Anthropic, Claude Opus 4.7) for the purposes of language refinement and improving the readability of the text. The authors have reviewed and edited the output and take full responsibility for the content of this publication.

**Author Contributions:** Conceptualization, T.T. and A.S.; methodology, T.T., A.S., and H.C.; software, T.T.; validation, T.T., A.S., H.I.A., and H.C.; formal analysis, T.T. and A.S.; investigation, T.T., A.S., H.I.A., and H.C.; resources, A.S. and S.S.; data curation, T.T., A.S., and S.S.; writing—original draft preparation, T.T., H.I.A., and H.C.; writing—review and editing, T.T., A.S., H.I.A., H.C., and S.S.; visualization, T.T. and H.I.A.; supervision, A.S. and S.S.; project administration, A.S. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The Sleep-Disordered Breathing (SDB) dataset used in this study is publicly available on Kaggle at <https://www.kaggle.com/datasets/ziya07/sleep-disordered-breathing-detection>. The processed Obstructive Sleep Apnea (OSA) dataset is derived from a previously published study [40] and is available from the corresponding author upon reasonable request.

**Conflicts of Interest:** The authors declare that there is no conflict of interest regarding the publication of this paper.

## Appendix A

**Table A1.** Training-set classification performance of baseline models on the OSA dataset.

model	Measure	accuracy	precision	recall	f1	roc_auc	mean rank
DT	Avg	<b>1.0000</b>	<b>1.0000</b>	<b>1.0000</b>	<b>1.0000</b>	<b>1.0000</b>	1.0
	Std	0.0000	0.0000	0.0000	0.0000	0.0000	
ELM	Avg	0.7740	0.7708	0.7195	0.7435	0.7696	7.2
	Std	0.0218	0.0239	0.0454	0.0293	0.0230	
KNN	Avg	<b>1.0000</b>	<b>1.0000</b>	<b>1.0000</b>	<b>1.0000</b>	<b>1.0000</b>	1.0
	Std	0.0000	0.0000	0.0000	0.0000	0.0000	
LR	Avg	0.7521	0.7536	0.6790	0.7142	0.8402	7.8
	Std	0.0133	0.0151	0.0229	0.0173	0.0108	
MLP	Avg	0.9553	0.9516	0.9505	0.9509	0.9923	4.0
	Std	0.0100	0.0089	0.0224	0.0115	0.0024	
RF	Avg	0.9966	0.9975	0.9950	0.9962	0.9999	3.0
	Std	0.0036	0.0044	0.0061	0.0039	0.0002	
SVM	Avg	0.8543	0.8959	0.7710	0.8282	0.9364	6.0
	Std	0.0164	0.0172	0.0377	0.0224	0.0072	
XGB	Avg	0.8879	0.9112	0.8365	0.8719	0.9627	5.0
	Std	0.0152	0.0224	0.0268	0.0178	0.0057	

**Table A2.** Training-set classification performance of baseline models on the SDB dataset.

model	Measure	accuracy	precision	recall	f1	roc_auc	mean rank
DT	Avg	<b>1.0000</b>	<b>1.0000</b>	<b>1.0000</b>	<b>1.0000</b>	<b>1.0000</b>	1.0
	Std	0.0000	0.0000	0.0000	0.0000	0.0000	
ELM	Avg	0.7795	0.7860	0.9770	0.8711	0.5612	7.0
	Std	0.0101	0.0089	0.0088	0.0053	0.0226	
KNN	Avg	<b>1.0000</b>	<b>1.0000</b>	<b>1.0000</b>	<b>1.0000</b>	<b>1.0000</b>	1.0
	Std	0.0000	0.0000	0.0000	0.0000	0.0000	
LR	Avg	0.7621	0.7624	0.9995	0.8650	0.6098	6.0
	Std	0.0009	0.0002	0.0012	0.0006	0.0136	
MLP	Avg	0.8208	0.8172	0.9856	0.8935	0.8698	5.0
	Std	0.0090	0.0098	0.0065	0.0046	0.0124	
RF	Avg	0.9965	0.9958	0.9997	0.9977	1.0000	1.0
	Std	0.0024	0.0028	0.0015	0.0015	0.0001	
SVM	Avg	0.7656	0.7649	<b>1.0000</b>	0.8668	0.5389	8.0
	Std	0.0034	0.0026	0.0000	0.0017	0.4601	
XGB	Avg	0.7626	0.7626	<b>1.0000</b>	0.8653	0.9324	4.0
	Std	0.0006	0.0004	0.0000	0.0003	0.0106	

## References

- Lee, Y.C.; Lu, C.T.; Chuang, L.P.; Lee, L.A.; Fang, T.J.; Cheng, W.N.; Li, H.Y. Pharmacotherapy for obstructive sleep apnea - A systematic review and meta-analysis of randomized controlled trials. *Sleep Medicine Reviews* **2023**, *70*, 101809. <https://doi.org/10.1016/j.smrv.2023.101809>.
- Meyer, E.J.; Wittert, G.A. Approach the Patient With Obstructive Sleep Apnea and Obesity. *The Journal of Clinical Endocrinology & Metabolism* **2024**, *109*, e1267–e1279, [<https://academic.oup.com/jcem/article-pdf/109/3/e1267/56680639/dgad572.pdf>]. <https://doi.org/10.1210/clinem/dgad572>.
- Piriyajitakonkij, M.; Warin, P.; Lakhan, P.; Leelaarporn, P.; Kumchaiseemak, N.; Suwajanakorn, S.; Pianpanit, T.; Niparnan, N.; Mukhopadhyay, S.C.; Wilaiprasitporn, T. SleepPoseNet: Multi-View Learning for Sleep Postural Transition Recognition Using UWB. *IEEE Journal of Biomedical and Health Informatics* **2020**, pp. 1–1. <https://doi.org/10.1109/JBHI.2020.3025900>.
- Kaditis, A.G.; Alonso Alvarez, M.L.; Boudewyns, A.; Alexopoulos, E.I.; Ersu, R.; Joosten, K.; Larramona, H.; Miano, S.; Narang, I.; Trang, H.; et al. Obstructive sleep disordered breathing in 2- to 18-year-old children: diagnosis and management. *European Respiratory Journal* **2016**, *47*, 69–94. <https://doi.org/10.1183/13993003.00385-2015>.
- Banluesombatkul, N.; Ouppaphan, P.; Leelaarporn, P.; Lakhan, P.; Chaitusaney, B.; Jaimchariya, N.; Chuangsuwanich, E.; Chen, W.; Phan, H.; Dilokthanakul, N.; et al. MetaSleepLearner: A Pilot Study on Fast Adaptation of Bio-signals-Based Sleep Stage Classifier to New Individual Subject Using Meta-Learning. *IEEE Journal of Biomedical and Health Informatics* **2020**, pp. 1–1. <https://doi.org/10.1109/JBHI.2020.3037693>.

6. American Sleep Apnea Association. <https://www.sleepapnea.org/learn/sleep-apnea-information-clinicians/>. Accessed: 2019-04-22.
7. AASM. American Academy of Sleep Medicine: Economic burden of undiagnosed sleep apnea in U.S. is nearly \$150B per year, 2023. Accessed: Aug. 8, 2016.
8. Kuo, N.Y.; Tsai, H.J.; Tsai, S.J.; Yang, A.C. Efficient Screening in Obstructive Sleep Apnea Using Sequential Machine Learning Models, Questionnaires, and Pulse Oximetry Signals: Mixed Methods Study. *J Med Internet Res* **2024**, *26*, e51615. <https://doi.org/10.2196/51615>.
9. Sheta, A.; Turabieh, H.; Braik, M.; Surani, S.R. Diagnosis of Obstructive Sleep Apnea Using Logistic Regression and Artificial Neural Networks Models. In Proceedings of the Proceedings of the Future Technologies Conference (FTC) 2019; Arai, K.; Bhatia, R.; Kapoor, S., Eds., Cham, 2020; pp. 766–784.
10. Aiyer, I.; Shaik, L.; Sheta, A.; Surani, S. Review of Application of Machine Learning as a Screening Tool for Diagnosis of Obstructive Sleep Apnea. *Medicina* **2022**, *58*. <https://doi.org/10.3390/medicina58111574>.
11. Surani, S.; Sheta, A.; Turabieh, H.; Park, J.; Mathur, S.; Katangur, A. Diagnosis of Sleep Apnea Using artificial Neural Network and binary Particle Swarm Optimization for Feature Selection. *Chest* **2019**, *156*, A136.
12. Markowska-Kaczmar, U.; Kosturek, M. Extreme learning machine versus classical feedforward network: Comparison from the usability perspective. *Neural Computing and Applications* **2021**, *33*, 15121–15144. <https://doi.org/10.1007/s00521-021-06402-y>.
13. Abu Al-Haija, Q.; Altamimi, S.; AlWadi, M. Analysis of Extreme Learning Machines (ELMs) for intelligent intrusion detection systems: A survey. *Expert Systems with Applications* **2024**, *253*, 124317. <https://doi.org/https://doi.org/10.1016/j.eswa.2024.124317>.
14. Albadr, M.; Tiun, S.; Ayob, M.; Al-Dhief, F. Particle Swarm Optimization-Based Extreme Learning Machine for COVID-19 Detection. *Cognitive Computation* **2022**, *16*, 1–16. <https://doi.org/10.1007/s12559-022-10063-x>.
15. Wang, J.; Lu, S.; Wang, S.H.; Zhang, Y.D. A review on extreme learning machine. *Multimedia Tools and Applications* **2022**, *81*, 41611–41660. <https://doi.org/10.1007/s11042-021-11007-7>.
16. Eshtay, M.; Faris, H.; Obeid, N. Metaheuristic-based extreme learning machines: a review of design formulations and applications. *International Journal of Machine Learning and Cybernetics* **2019**, *10*, 1543–1561. <https://doi.org/10.1007/s13042-018-0833-6>.
17. Thaher, T.; Sheta, A.; Awad, M.; Aldasht, M. Enhanced variants of crow search algorithm boosted with cooperative based island model for global optimization. *Expert Systems with Applications* **2024**, *238*, 121712. <https://doi.org/https://doi.org/10.1016/j.eswa.2023.121712>.
18. Qian, Y.; Dharmage, S.C.; Hamilton, G.S.; Lodge, C.J.; Lowe, A.J.; Zhang, J.; Bowatte, G.; Perret, J.L.; Senaratna, C.V. Longitudinal risk factors for obstructive sleep apnea: A systematic review. *Sleep Medicine Reviews* **2023**, *71*, 101838. <https://doi.org/https://doi.org/10.1016/j.smrv.2023.101838>.
19. Chang, J.L.; Goldberg, A.N.; Alt, J.A.; Mohammed, A.; Ashbrook, L.; Auckley, D.; Ayappa, I.; Bakhtiar, H.; Barrera, J.E.; Bartley, B.L.; et al. International Consensus Statement on Obstructive Sleep Apnea. *International Forum of Allergy & Rhinology* **2023**, *13*, 1061–1482, [<https://onlinelibrary.wiley.com/doi/pdf/10.1002/alr.23079>]. <https://doi.org/https://doi.org/10.1002/alr.23079>.
20. Duarte, M.; Pereira-Rodrigues, P.; Ferreira-Santos, D. The role of novel digital clinical tools in the screening or diagnosis of obstructive sleep apnea: systematic review. *Journal of Medical Internet Research* **2023**, *25*, e47735.
21. Maniaci, A.; Riela, P.M.; Iannella, G.; Lechien, J.R.; La Mantia, I.; De Vincentiis, M.; Cammaroto, G.; Calvo-Henriquez, C.; Di Luca, M.; Chiesa Estomba, C.; et al. Machine learning identification of obstructive sleep apnea severity through the patient clinical features: a retrospective study. *Life* **2023**, *13*, 702.
22. Srivastava, G.; Chauhan, A.; Kargeti, N.; Pradhan, N.; Dhaka, V.S. ApneaNet: A hybrid 1DCNN-LSTM architecture for detection of Obstructive Sleep Apnea using digitized ECG signals. *Biomedical Signal Processing and Control* **2023**, *84*, 104754.
23. Brennan, H.L.; Kirby, S.D. Barriers of artificial intelligence implementation in the diagnosis of obstructive sleep apnea. *Journal of Otolaryngology-Head & Neck Surgery* **2022**, *51*, 16.
24. Yeh, E.; Wong, E.; Tsai, C.W.; Gu, W.; Chen, P.L.; Leung, L.; Wu, I.C.; Strohl, K.P.; Folz, R.J.; Yar, W.; et al. Detection of obstructive sleep apnea using Belun Sleep Platform wearable with neural network-based algorithm and its combined use with STOP-Bang questionnaire. *PLoS One* **2021**, *16*, e0258040.
25. Shi, E.; Zhang, Y.; Cao, Z.; Ma, L.; Yuan, Y.; Niu, X.; Su, Y.; Xie, Y.; Chen, X.; Xing, L.; et al. Application and interpretation of machine learning models in predicting the risk of severe obstructive sleep apnea in adults. *BMC Medical Informatics and Decision Making* **2023**, *23*. <https://doi.org/10.1186/s12911-023-02331-z>.

26. Banluesombatkul, N.; Rakthanmanon, T.; Wilaiprasitporn, T. Single Channel ECG for Obstructive Sleep Apnea Severity Detection Using a Deep Learning Approach. In Proceedings of the TENCON 2018 - 2018 IEEE Region 10 Conference, 2018, pp. 2011–2016. <https://doi.org/10.1109/TENCON.2018.8650429>.
27. L., Z.; D., F.; R., U.; D., K. 0311 Automated Apnea and Hypopnea Event Detection Using Deep Learning. *Sleep* **2018**, *41*, A119–A120. Copyright - Copyright © 2018 Sleep Research Society; Last updated - 2018-07-09.
28. Faust, O.; Barika, R.; Shenfield, A.; Ciaccio, E.J.; Acharya, U.R. Accurate detection of sleep apnea with long short-term memory network based on RR interval signals. *Knowledge-Based Systems* **2021**, *212*, 106591. <https://doi.org/https://doi.org/10.1016/j.knosys.2020.106591>.
29. Haberfeld, C.; Sheta, A.; Hossain, M.S.; Turabieh, H.; Surani, S. SAS Mobile Application for Diagnosis of Obstructive Sleep Apnea Utilizing Machine Learning Models. In Proceedings of the 2020 11th IEEE Annual Ubiquitous Computing, Electronics Mobile Communication Conference (UEMCON), 2020, pp. 0522–0529. <https://doi.org/10.1109/UEMCON51285.2020.9298041>.
30. Azimi, H.; Xi, P.; Bouchard, M.; Goubran, R.; Knoefel, F. Machine Learning-Based Automatic Detection of Central Sleep Apnea Events From a Pressure Sensitive Mat. *IEEE Access* **2020**, *8*, 173428–173439. <https://doi.org/10.1109/ACCESS.2020.3025808>.
31. Huang, W.; Lee, P.; Liu, Y.; Lai, F. 0495 Prediction Of Obstructive Sleep Apnea Using Machine Learning Technique. *Sleep* **2018**, *41*, A186–A186. <https://doi.org/10.1093/sleep/zsy061.494>.
32. K., T.; K., J.W.; L., K. Detection of sleep disordered breathing severity using acoustic biomarker and machine learning techniques. *BioMedical Engineering OnLine* **2018**, *17*, 16. <https://doi.org/10.1186/s12938-018-0448-x>.
33. Alshaer, H.; Hummel, R.; Mendelson, M.; Marshal, T.; Bradley, T.D. Objective Relationship Between Sleep Apnea and Frequency of Snoring Assessed by Machine Learning. *Journal of Clinical Sleep Medicine* **2019**, *15*, 463–470. <https://doi.org/10.5664/jcsm.7676>.
34. Bozkurt, F.; Uçar, M.K.; Bozkurt, M.R.; Bilgin, C. Detection of abnormal respiratory events with single channel ECG and hybrid machine learning model in patients with obstructive sleep apnea. *IRBM* **2020**.
35. Sheta, A.; Turabieh, H.; Thaher, T.; Too, J.; Mafarja, M.; Hossain, M.S.; Surani, S.R. Diagnosis of obstructive sleep apnea from ECG signals using machine learning and deep learning classifiers. *Applied Sciences* **2021**, *11*, 6622.
36. Ferreira-Santos, D.; Amorim, P.; Silva Martins, T.; Monteiro-Soares, M.; Pereira Rodrigues, P. Enabling early obstructive sleep apnea diagnosis with machine learning: systematic review. *Journal of Medical Internet Research* **2022**, *24*, e39452.
37. Bahrami, M.; Forouzanfar, M. Sleep apnea detection from single-lead ECG: A comprehensive analysis of machine learning and deep learning algorithms. *IEEE Transactions on Instrumentation and Measurement* **2022**, *71*, 1–11.
38. Kohzadi, Z.; Safdari, R.; Haghighi, K.S. Evaluation of the PSO metaheuristic algorithm in different types of sleep apnea diagnosis using RR intervals. *Journal of Biomedical Physics & Engineering* **2023**, *13*, 147.
39. Pouramirarsalani, S.; Maleki, S.E.; Rajebi, S.; Manaf, N.V.; Roohany, A. Diagnosis of sleep apnea by optimal fuzzy system based on respiratory signals. In Proceedings of the 2024 10th International Conference on Artificial Intelligence and Robotics (QICAR). IEEE, 2024, pp. 100–105.
40. Sheta, A.; Thaher, T.; Surani, S.R.; Turabieh, H.; Braik, M.; Too, J.; Abu-El-Rub, N.; Mafarjah, M.; Chantar, H.; Subramanian, S. Diagnosis of Obstructive Sleep Apnea Using Feature Selection, Classification Methods, and Data Grouping Based Age, Sex, and Race. *Diagnostics* **2023**, *13*. <https://doi.org/10.3390/diagnostics13142417>.
41. ziya07. Sleep Disordered Breathing Detection. Kaggle dataset. Accessed: 2025-12-13.
42. Golowich, N.; Rakhlin, A.; Shamir, O. Size-independent sample complexity of neural networks. *Information and Inference: A Journal of the IMA* **2020**, *9*, 473–504, [<https://academic.oup.com/imaiai/article-pdf/9/2/473/33321322/iaz007.pdf>]. <https://doi.org/10.1093/imaiai/iaz007>.
43. Huang, G.B.; Zhu, Q.Y.; Siew, C.K. Extreme learning machine: a new learning scheme of feedforward neural networks. In Proceedings of the 2004 IEEE international joint conference on neural networks (IEEE Cat. No. 04CH37541). Ieee, 2004, Vol. 2, pp. 985–990.
44. Eshtay, M.; Faris, H.; Obeid, N. Metaheuristic-based extreme learning machines: a review of design formulations and applications. *International Journal of Machine Learning and Cybernetics* **2019**, *10*, 1543–1561.
45. Van Thieu, N.; Houssein, E.H.; Oliva, D.; Hung, N.D. IntelELM: A python framework for intelligent metaheuristic-based extreme learning machine. *Neurocomputing* **2025**, *618*, 129062. <https://doi.org/10.1016/j.neucom.2024.129062>.

46. Nguyen, T.; Hoang, B.; Nguyen, G.; Nguyen, B.M. A new workload prediction model using extreme learning machine and enhanced tug of war optimization. *Procedia Computer Science* **2020**, *170*, 362–369. <https://doi.org/10.1016/j.procs.2020.03.063>.
47. Whitley, D. A genetic algorithm tutorial. *Statistics and Computing* **1994**, *4*, 65–85. <https://doi.org/10.1007/BF00175354>.
48. Ahmadianfar, I.; Heidari, A.A.; Gandomi, A.H.; Chu, X.; Chen, H. RUN beyond the metaphor: An efficient optimization algorithm based on Runge Kutta method. *Expert Systems with Applications* **2021**, *181*, 115079. <https://doi.org/https://doi.org/10.1016/j.eswa.2021.115079>.
49. Gupta, S.; Deep, K.; Mirjalili, S. An efficient equilibrium optimizer with mutation strategy for numerical optimization. *Applied Soft Computing* **2020**, *96*, 106542. <https://doi.org/https://doi.org/10.1016/j.asoc.2020.106542>.
50. Liang, J.; Qin, A.; Suganthan, P.; Baskar, S. Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. *IEEE Transactions on Evolutionary Computation* **2006**, *10*, 281–295. <https://doi.org/10.1109/TEVC.2005.857610>.
51. Tang, C.; Sun, W.; Wu, W.; Xue, M. A hybrid improved whale optimization algorithm. In Proceedings of the 2019 IEEE 15th International Conference on Control and Automation (ICCA), 2019, pp. 362–367. <https://doi.org/10.1109/ICCA.2019.8900003>.
52. Lou, L.; Xia, W.; Sun, Z.; Quan, S.; Yin, S.; Gao, Z.; Lin, C. COVID-19 mortality prediction using ensemble learning and grey wolf optimization. *PeerJ Computer Science* **2023**, *9*, e1209. <https://doi.org/10.7717/peerj-cs.1209>.
53. Yang, Y.; Chen, H.; Heidari, A.A.; Gandomi, A.H. Hunger games search: Visions, conception, implementation, deep analysis, perspectives, and towards performance shifts. *Expert Systems with Applications* **2021**, *177*, 114864. <https://doi.org/https://doi.org/10.1016/j.eswa.2021.114864>.
54. Heidari, A.A.; Mirjalili, S.; Faris, H.; Aljarah, I.; Mafarja, M.; Chen, H. Harris hawks optimization: Algorithm and applications. *Future Generation Computer Systems* **2019**, *97*, 849–872. <https://doi.org/https://doi.org/10.1016/j.future.2019.02.028>.
55. Zhao, S.; Zhang, T.; Ma, S.; Wang, M. Sea-horse optimizer: a novel nature-inspired meta-heuristic for global optimization problems. *Applied Intelligence* **2022**, *53*, 11833–11860. <https://doi.org/10.1007/s10489-022-03994-3>.
56. Abdollahzadeh, B.; Gharehchopogh, F.S.; Khodadadi, N.; Mirjalili, S. Mountain Gazelle Optimizer: A new Nature-inspired Metaheuristic Algorithm for Global Optimization Problems. *Advances in Engineering Software* **2022**, *174*, 103282. <https://doi.org/https://doi.org/10.1016/j.advengsoft.2022.103282>.
57. Obadina, O.O.; Thaha, M.A.; Althoefer, K.; Shaheed, M.H. Dynamic characterization of a master–slave robotic manipulator using a hybrid grey wolf–whale optimization algorithm. *Journal of Vibration and Control* **2022**, *28*, 1992–2003, [<https://doi.org/10.1177/10775463211003402>]. <https://doi.org/10.1177/10775463211003402>.

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.