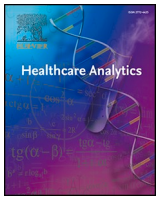
Healthcare Analytics 7 (2025) 100374

Contents lists available at ScienceDirect

Healthcare Analytics

journal homepage: www.elsevier.com/locate/health



An integrated stacked convolutional neural network and the levy flight-based grasshopper optimization algorithm for predicting heart disease

Syed Muhammad Salman Bukhari [a,1], Muhammad Hamza Zafar [b,1], Syed Kumayl Raza Moosavi [b], Majad Mansoor [c], Filippo Sanfilippo [b,d,*]

[a] Department of Electrical Engineering, Capital University of Science and Technology, Islamabad, Pakistan
[b] Department of Engineering Sciences, University of Agder, Grimstad, Norway
[c] Department of Automation, University of Science and Technology of China, Hefei, China
[d] Department of Software Engineering, Kaunas University of Technology, Kaunas, Lithuania

ARTICLE INFO

Handling editor: Madijd Tavana

Keywords:
Heart disease prediction
Convolutional neural network
Grasshopper optimization
Deep learning

ABSTRACT

Cardiovascular disease is the leading cause of death worldwide, including critical conditions such as blood vessel blockage, heart failure, and stroke. Accurate and early prediction of heart disease remains a significant challenge due to the complexity of symptoms and the variability of contributing factors. This study proposes a novel hybrid model integrating a Stacked Convolutional Neural Network (SCNN) with the Levy Flight-based Grasshopper Optimization Algorithm (LFGOA) to address this challenge. The SCNN provides robust feature extraction, while LFGOA enhances the model by optimizing hyperparameters, improving classification accuracy, and reducing overfitting. The proposed approach is evaluated using four publicly available heart disease datasets, each representing diverse clinical and demographic features. Compared to traditional classifiers, including Regression Trees, Support Vector Machine, Logistic Regression, K-Nearest Neighbors, and standard Neural Networks, the SCNN-LFGOA consistently outperforms these methods. The results highlight that the SCNN-LFGOA achieves an average accuracy of 99%, with significant improvements in specificity, sensitivity, and F1-Score, showcasing its adaptability and robustness across datasets. This study highlights the SCNN-LFGOA's potential as a transformative tool for early and accurate heart disease prediction, contributing to improved patient outcomes and more efficient healthcare resource utilization. By combining deep learning with an advanced optimization technique, this research introduces a scalable and effective solution to a critical healthcare problem.

1. Introduction

Heart disease is a leading cause of death worldwide, encompassing critical conditions such as blood vessel blockage, heart failure, and stroke. The number of deaths attributed to heart disease has risen sharply in recent years, with approximately 17.9 million individuals worldwide dying from the disease in 2016 alone, representing nearly 30% of global fatalities [1]. Despite advancements in medical technology, the early and accurate prediction of heart disease remains a substantial challenge due to the complex, multifactorial nature of cardiovascular conditions and the subtle, often asymptomatic progression in early stages. This issue is compounded by the economic burden of heart disease, which consumes around 4% of annual healthcare expenditures, with over half of diagnosed patients passing away within three years [2,3].

Effective heart disease diagnosis typically relies on two approaches: non-invasive and invasive diagnostics. Non-invasive methods, including electrocardiography (EKG), phonocardiography (PCG), active cardiac monitoring, and ultrasonography, are safer and more cost-effective, facilitating their widespread use in early detection efforts [4]. However, invasive procedures are often preferred for advanced diagnostics in severe cases to obtain comprehensive insights. The significant and growing incidence of heart disease underscores the need for rapid and accurate diagnostic tools that can enhance early detection, reduce

* Corresponding author. Department of Engineering Sciences, University of Agder, Grimstad, Norway.
E-mail address: filippo.sanfilippo@uia.no (F. Sanfilippo).
[1] These authors contributed equally to this work.

https://doi.org/10.1016/j.health.2024.100374
Received 15 July 2023; Received in revised form 30 November 2024; Accepted 6 December 2024
Available online 7 December 2024

mortality rates, and optimize healthcare resource allocation.

Despite advancements in medical technology, the early and accurate prediction of heart disease remains a substantial challenge due to the complex, multifactorial nature of cardiovascular conditions and their subtle, often asymptomatic progression in early stages. Traditional diagnostic methods, while effective in some cases, struggle with limitations such as overfitting in predictive models, handling noisy or imbalanced data, and adapting to the variability of patient demographics across datasets. These challenges necessitate the development of advanced predictive models capable of robustly handling diverse clinical data while maintaining high accuracy and reliability. In response, this study proposes a hybrid approach leveraging the strengths of a Stacked Convolutional Neural Network (SCNN) and the Levy Flight-based Grasshopper Optimization Algorithm (LFGOA). The SCNN excels in extracting meaningful patterns from high-dimensional data, while LFGOA optimizes hyperparameters, reducing overfitting and enhancing adaptability across datasets. Together, these components create a robust solution designed specifically to address the aforementioned challenges in heart disease prediction.

The main contributions of this study are summarized as follows.

- **Development of the SCNN-LFGOA Technique:** A novel classification model that integrates SCNN and LFGOA, specifically designed to address limitations in handling high-dimensional data and dataset variability.
- **Enhanced Prediction Accuracy:** Demonstrates superior performance with an average accuracy of 99% across four heart disease datasets, significantly outperforming traditional classifiers such as Regression Trees and Logistic Regression.
- **Comprehensive Multi-Dataset Evaluation:** Evaluates the model across diverse datasets, showcasing its adaptability and robustness, unlike prior approaches that often lack scalability or consistency.
- **In-Depth Metric Analysis:** Provides a detailed evaluation using metrics such as specificity, sensitivity, and F1-Score, ensuring a holistic assessment of the model's effectiveness.
- **Advancement in Healthcare Deep Learning:** Introduces a scalable and accurate classification framework for deep learning in healthcare, potentially influencing future research and clinical applications in heart disease diagnostics.

This manuscript is organized as follows: Section 2 reviews existing heart disease prediction models, emphasizing the limitations addressed by this study. Section 3 details the proposed hybrid DL and meta-heuristic algorithm. Section 4 describes the datasets and their characteristics. Section 5 presents and discusses the model's performance results.

2. Literature review

Predicting heart disease is challenging due to numerous risk factors, including high blood pressure, elevated cholesterol levels, and irregular heartbeats [5]. Machine learning (ML) and deep learning (DL) algorithms have proven effective in processing extensive healthcare data for cardiac disease prediction [5]. Various models employ combinations of parameters and classification algorithms, which are central to traditional approaches for forecasting heart disease. ML models extract insights through statistical and mathematical analyses, which can later generalize predictions based on available features [6].

Techniques like Density-Based Spatial Clustering of Applications with Noise (DBSCAN), XGBoost, and Synthetic Minority Over-Sampling Technique-Edited Nearest Neighbors (SMOTE-ENN) have been integrated into heart disease prediction models to manage data distribution, identify and remove outliers, and improve accuracy [7]. Recent comparative studies indicate that such hybrid models often outperform traditional models. For instance, optimized feature selection approaches enhance the prediction accuracy of ML techniques for cardiovascular disease by focusing on critical features, allowing for more precise results [8].

Several classification techniques, including decision trees, artificial neural networks (ANNs), K-nearest neighbors (KNN), fuzzy logic, logistic regression, and support vector machines (SVM), have been extensively applied to heart disease prediction [9]. As demonstrated in recent studies, these algorithms, when combined with optimized feature selection, improve prediction accuracy and support early disease detection. Consequently, these strategies, coupled with effective categorization techniques, improve accuracy in distinguishing between benign and abnormal heart conditions [10].

ML and DL models benefit from vast datasets, including digital

Table 1
Comparison Analysis of different ML/DL Techniques on Heart Datasets.

Ref.	Year	Dataset	Proposed Technique	Summary	Evaluation Metrics
[22]	2022	CVD (Dataset 1)	MLP-EMBDA	Employing attribute selection and classification, to precisely forecast heart illness. Thus, a novel MultiLayer Perceptron for Enhanced Brownian Motion based on Dragonfly Algorithm (MLP-EBMDA) and an optimized unsupervised feature selection strategy have been proposed in this study.	Accuracy = 0.97
[23]	2022	Framingham (Dataset 2)	Multilayered Perceptron	In this study, inputs for experimental analysis, entire and reduced feature subsets were used to examine several machine learning categorization models.	Accuracy = 0.71
[24]	2022	Heart UCI (Dataset 3)	Gradient Boosted Tree	In this work, several machine learning classifiers were used to identify crucial factors that improved the prediction of heart disease.	Accuracy = 0.94
[25]	2022	Heart Data (Dataset 4)	Cluster-based Bidirectional LSTM (C–Bi-LSTM) algorithm	In this work, cluster-based bi-directional long-short term memory (C–Bi-LSTM) has been presented to improve the predictive capability of conventional ML models.	Accuracy = 0.95

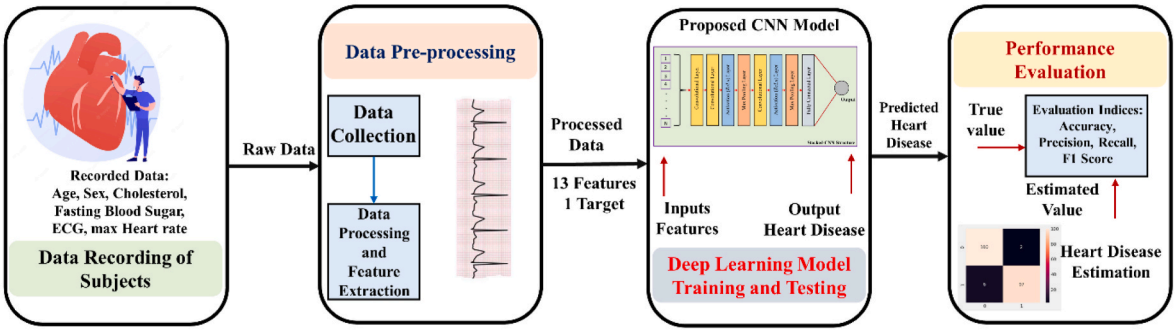


Fig. 1. Proposed structure for training and testing of deep learning model for heart rate prediction.

medical data collected from wearable devices, which supports efficient and accurate heart disease identification with minimal resources [11]. However, overfitting issues demand large datasets for ML model training [12]. To address the "curse of dimensionality"—where redundant or irrelevant features hinder model performance—optimized feature selection is crucial [13]. Feature selection methods, particularly in cardiovascular disease prediction, reduce model complexity, increase efficiency, and enhance predictive accuracy [14]. In heart disease research, feature selection algorithms are widely applied, particularly due to the advantages of handling high-dimensional data effectively [15].

CNN-based techniques have gained popularity for heart disease detection, as they allow for the representation of intermediate and advanced image features. Hybrid CNN models with multiple fully connected layers and convolutional layers demonstrate effectiveness in medical data analysis, particularly for disease prediction [16]. Recent studies have shown that CNN structures, optimized with backpropagation and augmented with feature selection algorithms, improve prediction performance in complex healthcare datasets [17].

Advanced DL techniques, such as deep belief networks (DBNs) and deep neural networks (DNNs), are increasingly used for precise cardiac disease prediction by employing multi-layer architectures [18]. The integration of DNNs with layered convolutional structures and database-driven approaches enhances accuracy in predicting heart disease, as seen in recent studies [19]. Comparative analyses of ML and DL techniques applied to heart disease datasets indicate that hybrid approaches integrating CNNs, DBNs, and optimization algorithms provide higher accuracy than conventional methods, particularly in clinical settings [20,21].

The comparative analysis of different papers on heart disease using different datasets and proposed technique is given in Table 1.

The accuracy of heart disease is tested in these literature evaluations using a variety of ML/DL models. However, it is strongly advised to obtain outcomes that are very close to perfect because precision is crucial, particularly in the medical profession. Prediction accuracy in conventional algorithms is a major issue. For datasets relating to heart illness, we have here suggested a hybrid model made up of stacked convolutional layers and a metaheuristic algorithm the Enhanced Grasshopper Optimization Algorithm (GOA) with Levy Flight.

Despite significant advancements in ML and DL for heart disease prediction, several limitations persist in existing literature. Traditional ML methods, such as logistic regression, decision trees, and support vector machines, often struggle to maintain consistent performance across diverse datasets due to their reliance on fixed assumptions and limited adaptability. Deep learning models, while effective in handling complex feature interactions, are prone to overfitting, particularly when working with high-dimensional data or imbalanced datasets. Furthermore, optimization algorithms commonly used in hybrid approaches lack the dynamic capability to balance exploration and exploitation, leading to suboptimal hyperparameter tuning and reduced generalizability in clinical applications. Another key challenge lies in ensuring scalability and robustness when integrating predictive models with diverse clinical indicators, such as age, sex, blood pressure, and cholesterol levels, while accounting for dataset-specific variations. These gaps highlight the need for a comprehensive approach that combines robust feature extraction with advanced optimization techniques to enhance both predictive accuracy and model adaptability.

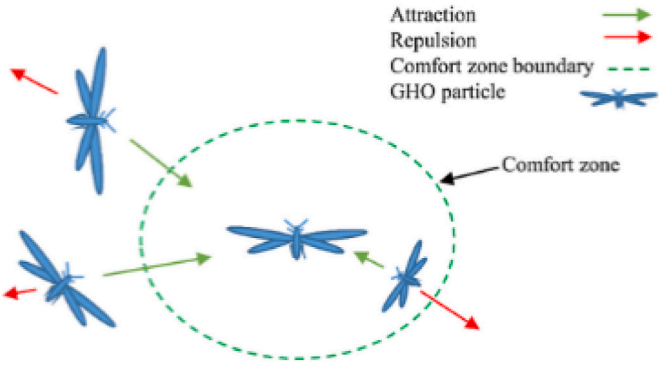


Fig. 2. Optimization procedure of Goa algorithm.

3. Proposed technique

In this section, first, the Enhanced Grasshopper Optimization with Levy Flight algorithm is presented highlighting the use of this approach for optimization problems. Successively, the architecture of the proposed S–CNN is explained. Finally, S–CNN training using LFGOA is outlined. The detailed structure if the proposed technique is shown in Fig. 1.

3.1. Grasshopper optimization algorithm (Goa)

The GOA method is motivated by the swarming and foraging behavior of grasshoppers in nature for tackling of the optimization in numerical issues [26]. Nymph and maturity are the two phases that the grasshopper experiences throughout its life cycle. The nymph phase is characterized by slow, tiny steps as opposed to the lengthy, swift movements that characterize the adult stage. Movements and maturation of nymphs serve as a metaphor for the times of intensity and heterogeneity of GOA. The search method for GOA is sliced into two stages: exploitation and exploration, as displayed in Fig. 2.

In the exploration phase, we calculate the fitness value for each grasshopper swarm and update all positional values (search for food sources). We identify the best option out of all options throughout the exploitation stage.

In the GOA method, each grasshopper denotes the number of participants' responses. The position X_i of each solution is calculated using a mathematical model of the behavior of grasshopper swarms as follows:

$$X_i = S_i + G_i + A_i \tag{1}$$

X_i characterizes the location of the *ith* candidate, S_i indicates the connection of the *ith* candidate with the other swarms, G_i symbolizes the force of gravity acting on the *ith* grasshopper, and A_i denotes the wind convection, and these variables may be represented by the equations below:

$$S_i = \sum_{j=1}^{N} s(d_{ij})\widehat{d_{ij}}, \text{where } i \neq j \tag{2}$$

$$s = fe^{-r} - e^{-r} \tag{3}$$

where N is the total number of candidates (grasshoppers), $d_{ij} = |\, x_j - x_i \,|$ represents the unit vector from the *ith* to the *jth* grasshopper swarm, and $\widehat{d_{ij}} = |x_j - x_i|$ is the Euclidean distance between the *ith* and the *jth* grasshopper swarm $\widehat{d_{ij}} = \left|\frac{x_j - x_i}{dij}\right|$ characterizes the unit vector from the *ith* to the *jth* candidate swarm.

Additionally, s signifies the intensity of two social elements (grasshopper swarms' mutual attraction and repulsiveness) and l represents the appealing length scale and f represents the degree of closeness. Repulsion happens when the distance between two grasshopper swarms is between *[0, 2.079],* and comfort zone is created when neither attraction nor repulsion happens at exactly 2.079. The attraction force increases as the distance exceeds *2.079*, then gradually declines until it approaches to the value four. When the distance between grasshopper swarms is greater than *10*, the function s is unable to impose forces between them. To address this issue, we map the separation between grasshopper swarms in the range [1,4]. Gravitational force G_i may be calculated using the equation shown below:

$$G_i = -ge_g \tag{4}$$

where g stands for gravitational constant and e_g is a unit vector pointing towards the earth's center. The formula for calculating A_i is given below:

$$A_i = ue_w \tag{5}$$

where e_w is the unit vector for the wind direction and u stands for the drift constant. Equation (1) may be recreated by changing the S_i, G_i, and A_i values in Equations (2)–(5).

$$\begin{aligned} X_i &= \sum_{j=1}^{N} s(d_{ij})\widehat{d_{ij}} - g\widehat{e_g} + u\widehat{e_w} \\ &= \sum_{j=1}^{N} s(|x_j - x_i|)\frac{|\widehat{x_j - x_i}|}{d_{ij}} - g\widehat{e_g} + u\widehat{e_w} \text{ where } i \neq j \end{aligned} \tag{6}$$

The mathematical equation (6) cannot be utilized directly to address the optimization issues since the grasshopper swarms soon reach their global optimum (comfort zone) and do not converge to a specific target or a designated point. The author suggests that the equation below be accurately actuarially performed in order to address optimization difficulties and prevent grasshopper swarms from swiftly reaching the comfortable zone:

$$X_i^d = c\left(\sum_{j=1}^{N} c\frac{UB_d - LB_d}{2} s\left(\left|x_j^d - x_i^d\right|\right)\frac{|x_j - x_i|}{d_{ij}}\right) + \widehat{T_d} \tag{7}$$

$\widehat{T_d}$ stands for the top solution thus far in the d_{th} dimension space, where UB_d and LB_d are the upper and lower limits in the d_{th} dimension, respectively. There is no G_i component in Equation (7) since the gravitational force is not considered and assume that the target T_d is always in the direction of the wind (A_i component). The second word, T_d, mimics the propensity of grasshoppers to migrate in the direction of their source of food.

a. Key Parameter "c" in GOA:

The grasshopper swarm approach relies on the significance of c in the (7) Equation for the global as well as local search. Based on the number of repeats, the inner c in (7) Eq. is utilized to modify the repulsion, attraction, and familiarity zone across bugs. Additionally, it is in charge of lessening the pressures that attract and repel grasshopper swarms. As the number of iterations increases, the outer c in Equation (7) restricts the grasshopper's motions about the target and aids in lowering the search coverage around the target. Here is how the coefficient c is suggested to work:

$$c = c_{max} - t\frac{c_{max} - c_{min}}{t_{max}} \tag{8}$$

where $c_{\max}$ and $c_{\min}$ are the maximum and minimum values of c, respectively, and t is the present recurrence and $t_{\max}$ is the highest possible value for iteration. You can choose to set $c_{\max}$ and $c_{\min}$ to 1 and 0.00001, respectively.

b. GOA with Levy Flight:

A grasshopper's position is updated using its current location, the world's best location, and the locations of other grasshoppers in the swarm.

$$L(x) \sim |x|^{-1-\theta} \tag{9}$$

where x characterizes the random step size of Levy's flying nature and a power-law index with a range of *[0, 2]* is set to *1.5* for the Levy distribution graph's peak sharpness control. The parameter has varied values that lead to distinct distributions; for lower values, it creates longer leaps, while for larger values, it makes shorter jumps. But the precise form of the Levy distribution is challenging to replicate in computer code. In Algorithm 1, the Levy Flight procedure is shown.

The Mantegna strategy, which generates a random variable with a likelihood distribution that is extremely comparable to the specified Levy stable dispersion, is one of the rapid and accurate ways. The Mantegna algorithm consists of three phases. The following is how Mantegna's approach calculates the step length S for random walks:

$$S = \frac{U}{|V|^{\frac{1}{\theta}}} \tag{10}$$

V and U should be obtained based on standardized distributions, where S is the random step length variable and U and V are two normal stochastic variables with standard deviations of σ_v and σ_u:

$$U \sim N\left(0, \sigma_U^2\right), V \sim \left(0, \sigma_V^2\right) \tag{11}$$

The symbol ~ in Eq. (11) indicates that samples should be taken from the distribution and that the random element complies with the distribution on the right. Since the standard deviations σ_u and σ_v cannot be selected separately for any value, we often set:

$$\sigma_v = 1 \tag{12}$$

Following this configuration, the standard deviation σ_u may be found by:

$$\sigma_U = \left\{\frac{\Gamma(1+\theta) \times sin\,(0.5\pi\theta)}{\Gamma[0.5(1+\theta)] \times \theta \times 2^{0.5(\theta-1)}}\right\}^{\frac{1}{\theta}} \tag{13}$$

Equations (9)–(13), which replicate the search of small walking distance and occasionally longer walking distance, were used to achieve the step size of the Levy flight. The step size is then determined using Equation (13).

$$step\ size = f * S \tag{14}$$

Levy flights help the new solution leave the search space, unless they

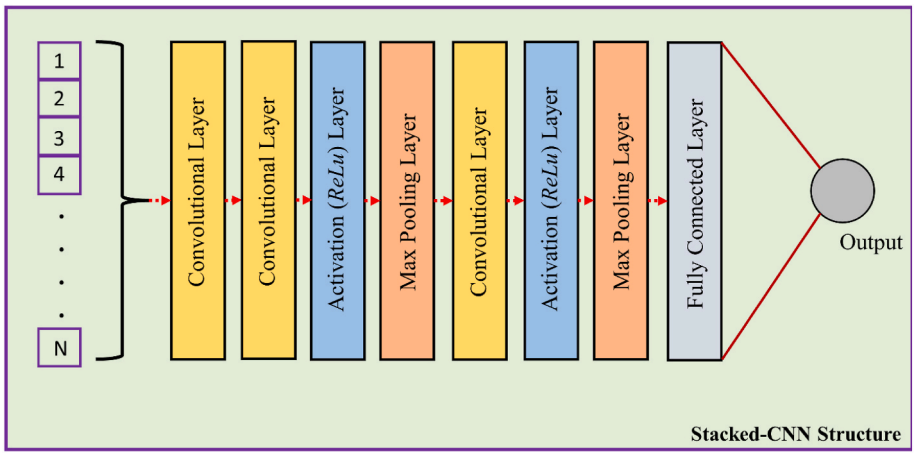


Fig. 3. Stacked 1DCNN structure for heart disease prediction.

become very aggressive. Levy walks are defined by the factor value *f (f = 0.01)* that is produced from *L/100*; the factor depends on the dimension of the desired problem. Algorithm 1 can demonstrate how the Levy flying method works. In Algorithm #1, the levy flying procedure is presented.

Algorithm # 1: Pseudo code of Levy flight function

1. Begin
2. In order to determine the step length S using Mantegna's algorithm.
3. $S = \frac{U}{|V|^{\frac{1}{\theta}}}$
4. where *θ parameter is a random value inside the [0,2] interval.*
5. U and V based on normal distributions with standard deviation σ_u and σ_v and gained by:
6. $U \sim N(0, \sigma_U^2), V \sim (0, \sigma_V^2)$
7. for *simplicity, the standard deviation σ_v usually be set $\sigma_v = 1$*
8. get standard deviation σ_u by Eq. (13):
9. $\sigma_U = \left\{ \frac{\Gamma(1+\theta) \times sin\,(0.5\pi\theta)}{\Gamma[0.5(1+\theta)] \times \theta \times 2^{0.5(\theta-1)}} \right\}^{\frac{1}{\theta}}$
10. *Step size is evaluated by Eq. (14): step size = f * S*
11. Execute actual random walk or flight with:
*12. New Position = current Position * LFGOA_Levy (dim)^f*
13. return *new position*
14. End

Algorithm # 2: Pseudo code for Levy Flight-based Grasshopper Optimization Algorithm

1. *Initialize parameters of c_{max} and c_{min}, maximum number of iterations L*
2. *Swarm Initialization*
3. X_i^d = *rand (N, dim). *(up - down) + down*
4. *Since determining the grasshopper's efficiency was the focus of the initial repetition*
5. ***for*** *j = 2 to N* ***do***
6. X_i^d = *LFGOA_Levy(dim)^f. *(high - low) + low*
7. *Calculate the fitness value for each grasshopper (search agent)*
8. *Find the best grasshopper in the first population.*
9. *Target_Position = sorted_grasshopper (1, :)*
10. *Target_Fitness = sorted_fitness (1)*
11. *l = 2*
12. *Start from the second iteration.*
13. *Since determining the grasshopper's efficiency was the focus of the initial repetition*
14. ***while*** *l < L + 1(Each grasshopper; agent l E (1, 2 ... N))* ***do***
15. $c = c_{max} - l\frac{c_{max} - c_{min}}{Max\_iter}$%
16. ***for*** *i = 1 to N* ***do***
17. Map *the distance of grasshoppers in the interval* [1,4]
$X_i^d = c\left(\sum_{j=1}^{N} c\frac{UB_d - LB_d}{2} s\left(\left|x_j^d - x_i^d\right|\right)\frac{\left|x_j - x_i\right|}{d_{ij}}\right) + \widehat{T}_d$
18. *Grasshopper_Pos (i, :) = Grasshopper_Pos (i, :). *LFGOA_levy(dim)*
19. ***end for***
20. $X_i^{new} = X_i^d + Td$
21. *Update the best solutions if there's a better one.*
22. ***if*** *Grasshopper_Pos (1, i) < TargetFitness* ***then***
23. *Target_Position = Grasshopper_Pos (i, :)*
24. *TargetFitness = Grasshopper Fitness (i, :)*
25. ***end if***
26. ***end while***
27. ***Return*** *so far, the optimal solution is obtained as the global optimal solution*

c. Adding Levy Flying to the GOA.

The GOA algorithm's placement of Levy Flight will immediately result in completely different outcomes; in some other cases it gives even worse results. Based on the aforementioned information, we effectively included Levy flight into the GOA algorithm using the straightforward, yet effective procedures shown below [27]. First, except for the first grasshopper (whose fitness was assessed using the first iteration), the remaining grasshoppers received Levy flying distribution values rather than rand values. Because of the high diversity in the initialization stage, most of the grasshoppers got off to a stronger start as a result of this.

Second, the Levy flying mechanism, which corrects the flaws and may be freed from an optimal local location and continued in another area of the search domain for the LFGOA, enables the objective to be achieved within the execution iteration. Algorithm 2 presents the LFGOA algorithm's pseudo-code.

Contrarily, despite the fact that current processes have significantly enhanced the GOA algorithms, there is still a high likelihood of entering a local optimum due to underdeveloped convergence. In addition, the GOA algorithm's usage of diversity-based truth justification remains in its infancy.

At the initialization step, Levy flight values rather than random integers between *[0, 1]* from the uniform distribution were allocated to each grasshopper. This immediately boosted the LFGOA algorithm's vast variety. The Levy flight function is represented by the aforementioned formula, LFGOA $Levy_{(dim)}$, where dim is the problem's dimension size. Second, randomization is more likely to increase the accuracy and capability of LFGOA's global search since any enormous step is theoretically possible and heavy-tailed random redistribution has a more effective step length.

3.2. Stacked-convolutional neural network architecture

In recent times, 1D Convolutional Neural Networks (1D-CNNs), an improved version of 2D-CNNs, were created [28]. The 1D-CNN model is typically an artificial feed-forward neural network made up of convolution layers and pooling layers as illustrated in Fig. 3.

The Rectified Linear Unit (ReLU) is commonly used to process the input through layers, neurons, and activation processes. This kind of activation is frequently used to introduce nonlinearity. Using dropout layers and normalization (scaling data) techniques, overfitting is frequently prevented. Under the same conditions (same architecture, system, and hyperparameters), a 1D-CNN is significantly less computationally challenging than a 2D-CNN. More recent studies show that the majority of 1D-CNN applications, which usually utilize small topologies, frequently employ networks with 50 neurons or less (with 1–2 hidden CNN layers). The tiny version of 1D-CNNs is particularly well suited for low-cost and real-time applications, especially on portable or mobile systems, due to its reduced progressing needs. Compact 1D-CNNs performed better in the specialized research for applications with poor labeling and strong signal fluctuations acquired from various sources (i.e., EKG/ECG, civil, time-series forecasting, high-power circuitry, power engines or motors, etc.). The main difference between 1D-CNN and 2D-CNN is that the latter model employs 1D arrays instead of the matrices that are frequently utilized in 2D-CNNs as input vectors. The kernel and filter sizes have also been converted to 1D values. As a result, 1D-CNN models utilize the number of layers and neurons present in the architecture to analyze the raw input and extract the relevant information from it as given by Eq. (15), Eq. (16), and Eq. (17).

$$\boldsymbol{a}_{o,fl}^{l} = \boldsymbol{f}\left(\sum_{\text{im}} \mathbf{a}_{\text{i}}^{1-1} * \mathbf{k}_{\text{io,fl}}^{1} + \mathbf{b}^{1}\right) \tag{15}$$

$$\boldsymbol{a}_{o}^{l} = \boldsymbol{f}\left[\boldsymbol{max}\left(\sum_{\boldsymbol{im}} \boldsymbol{a}_{i}^{l-1}\right) + \boldsymbol{b}^{l}\right] \tag{16}$$

$$\boldsymbol{a}_{o}^{l} = \boldsymbol{f}\left(\boldsymbol{a}_{i}^{l-1} * \boldsymbol{z}_{io}^{l} + \boldsymbol{b}^{l}\right) \tag{17}$$

where, a is a one-dimensional input matrix $(n\times1)$, f (.) is an activation function, $\mathrm{k}_{\mathrm{io,fl}}^{1}$ is a kernel filter (k_1), lth is the number of filters F, and a_o^l is the output of the lth convolutional layer. B and z are learnable parameters. The procedures of feature extraction, regression, and classification are combined into a single process to enhance the accuracy of the categorization or regression issue. The fundamental advantage of 1D CNNs is that they only require a series of 1D convolutions, which are just the linear weighted summing of two 1D arrays. This can also result in a cheap computing cost. Such a linear action can be carried out concurrently throughout the Forward-Propagation and Back-Propagation processes.

3.3. Synergistic contributions of SCNN and LFGOA

This subsection explains how the SCNN and the LFGOA work together to enhance predictive accuracy and optimize model performance in heart disease prediction.

1. **Feature Extraction with SCNN**:
   - The SCNN component is designed to handle complex patterns in heart disease data by learning high-level features across multiple layers. Its convolutional architecture captures essential relationships between clinical indicators (e.g., age, cholesterol levels, blood pressure) by applying multiple filters, which progressively learn from lower-to higher-level representations.
   - Through its stacked structure, SCNN effectively captures both local and global dependencies within the data, making it especially suited for heart disease prediction, where combinations of multiple risk factors influence the outcome. This comprehensive feature extraction lays a strong foundation for accurate classification.
2. **Parameter Optimization with LFGOA**:
   - LFGOA enhances the model by tuning SCNN's hyperparameters (such as learning rate, number of filters, and layer depth) to achieve optimal accuracy. Unlike traditional methods like grid or random search, LFGOA uses a metaheuristic approach inspired by grasshopper swarm behavior, which improves the efficiency of the search process.
   - The addition of Levy Flight in LFGOA introduces randomness into the search steps, allowing the algorithm to explore the parameter space more thoroughly. This randomness helps the optimization process avoid getting trapped in local optima, which could otherwise limit model performance. As a result, LFGOA finds a more globally optimal solution for SCNN's parameters, directly improving classification accuracy.
3. **Combined Effect for Enhanced Performance**:
   - Together, SCNN and LFGOA create a hybrid model where SCNN's powerful feature extraction is complemented by LFGOA's precision in hyperparameter tuning. SCNN builds a reliable classification framework by extracting relevant features, while LFGOA optimizes these features' impact by refining the parameters.
   - This combined approach not only boosts predictive accuracy but also improves model robustness across diverse heart disease datasets, outperforming traditional methods. By leveraging SCNN for in-depth data representation and LFGOA for optimal configuration, the SCNN-LFGOA model delivers a comprehensive solution for heart disease prediction, addressing the common challenges of overfitting, local optima, and generalization.

3.4. Hyperparameters of stacked-convolutional neural network

The performance of the DNN, particularly the stacked CNN structure described here, heavily depends on the careful tuning of its hyperparameters. Key hyperparameters in this model include the number of filters in each convolutional layer, filter size, stride, padding, batch size, dropout rate, and learning rate. Effective tuning of these parameters is essential to balance model accuracy and generalization [29].

Description of the hyperparameters are as follows.

- **Number of Filters and Filter Size**: These control the ability of the SCNN to capture intricate data patterns. Insufficient filters or overly small filter sizes may limit the model's capacity to learn complex features. Conversely, excessive filter numbers or overly large sizes can cause overfitting, reducing performance on new data.
- **Stride and Padding**: Stride dictates the step size for the convolution, affecting how much of the input is covered by each filter movement. Padding determines how the input is bordered, impacting the spatial dimensions of the output. Both are critical in preserving essential data features.
- **Learning Rate**: This sets the step size for each training iteration. A high learning rate may lead to divergence, while a low learning rate could slow down convergence. Finding an optimal learning rate is crucial for effective model training.
- **Batch Size**: This defines the number of samples processed before the model's internal parameters are updated. A larger batch size generally provides a more stable gradient estimation, whereas a smaller batch size may allow faster updates but with greater variance.
- **Dropout Rate**: Regularization is achieved through dropout, which helps prevent overfitting by randomly dropping units from the network during training. Adjusting the dropout rate helps ensure robust performance on new data.

Hyperparameter tuning can be performed using grid search, random search, or Bayesian optimization; however, in this study, the LFGOA is employed for efficient tuning. LFGOA enables the exploration of a wide

Table 2
Optimized values of SCNN hyperparameters achieved through LFGOA.

Hyperparameter	Range	Optimized Value
Number of Filters	[32, 64, 128]	64
Filter Size	[3x3, 5x5, 7x7]	5x5
Stride	[1,2]	1
Padding	[valid, same]	same
Learning Rate	[0.001, 0.01, 0.1]	0.01
Batch Size	[16, 32, 64, 128]	64

parameter space, finding optimal settings that improve both accuracy and efficiency. Table 2 summarizes the ranges of key hyperparameters and their optimized values as determined through LFGOA.

By tuning these hyperparameters, the SCNN achieves improved performance and generalization on unseen data, ensuring high predictive accuracy and stability.

3.5. Stacked-covolutional neural network training using levy flight-based grasshopper optimization algorithm

The **Levy Flight-based Grasshopper Optimization Algorithm (LFGOA)** is a metaheuristic optimization technique inspired by the natural swarming behavior of grasshoppers, with enhanced exploration capabilities introduced by Levy Flight. Its ability to balance exploration and exploitation has proven effective in addressing complex optimization problems, such as hyperparameter tuning for deep learning models.

For hyperparameter tuning of the stacked CNN (S–CNN), LFGOA systematically explores the hyperparameter space to identify the optimal configuration for maximizing classification performance. The **detailed flowchart** showing the hyperparameter tuning process using LFGOA is shown in Fig. 4, while the pseudocode of the algorithm is presented in **Algorithm 3**.

3.5.1. Cost functions for tuning

The choice of the cost function in LFGOA depends on the classification task and dataset characteristics. Common cost functions include.

1. **Accuracy:** The proportion of correctly identified cases, widely used for balanced datasets.
2. **F1 Score:** A metric combining precision and recall, suitable for imbalanced datasets.
3. **Cross-Entropy Loss:** A cost function that quantifies the dissimilarity between actual and predicted class probabilities, commonly used in deep learning.

For this study, **accuracy** is employed as the primary cost function due to the balanced nature of the datasets, ensuring that the selected hyperparameters directly improve model classification performance. In scenarios where datasets are imbalanced, the F1 score may be used to provide a more comprehensive evaluation of the model's effectiveness.

3.5.2. Algorithm description

The LFGOA begins by initializing a population of candidate solutions (grasshoppers), each representing a unique combination of hyperparameters. The algorithm iteratively updates the population based on a fitness function, which evaluates the classification performance of the S–CNN with the given hyperparameter configuration. Levy Flight introduces randomness to the updates, enabling the algorithm to avoid local optima and thoroughly explore the parameter space.

Algorithm 3: Hyperparameter Tuning of S–CNN using LFGOA

Input: Hyperparameter search space, Max_iterations, Population_size
Output: Optimal hyperparameter set H*
1. Initialize population of grasshoppers (H) randomly within the search space.
2. Evaluate the fitness of each solution using the selected cost function (e.g., accuracy).
3. Set the best solution H* as the current global best.
4. For iteration = 1 to Max_iterations do:

(*continued on next page*)

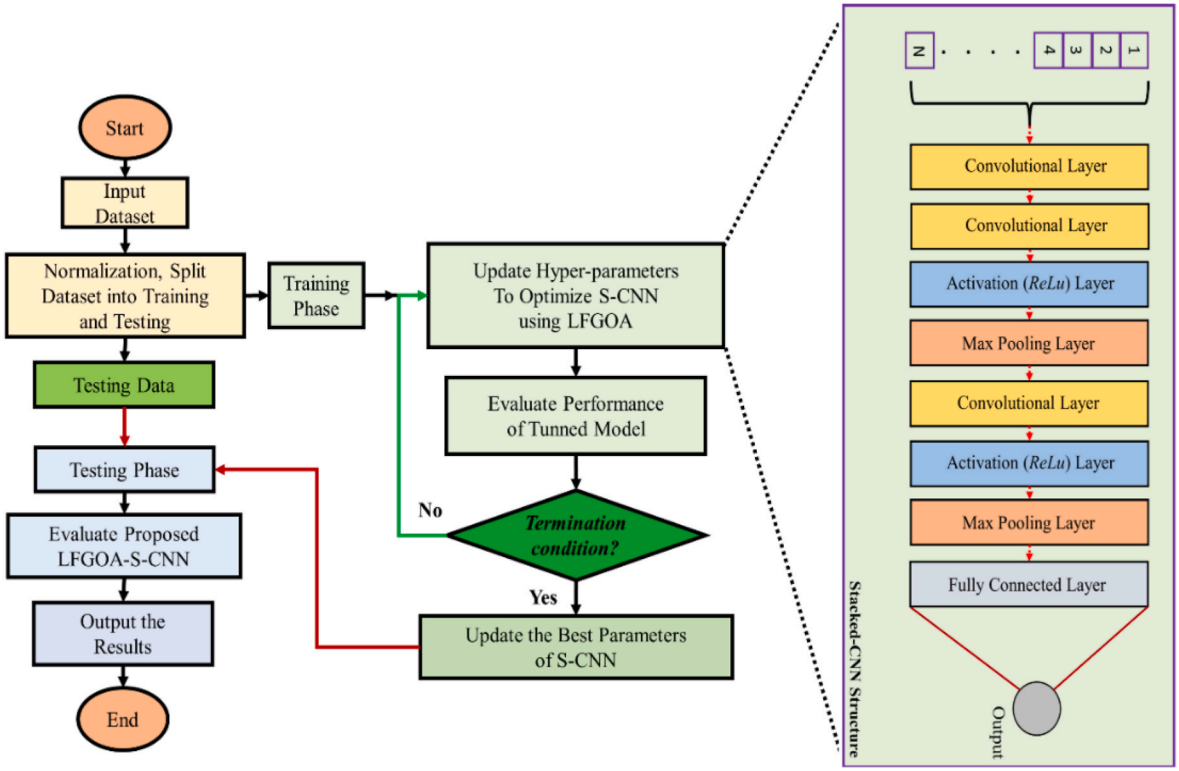


Fig. 4. Flow chart of hyperparameter tuning of S–CNN using LFGOA.

Table 3
Data Description of different available features for Prediction.

Sr. No	Dataset Attribute	Dataset Description	Dataset Value
1	Age	Age is an important factor in prediction.	Integer value.
2	Sex	Gender	Male = 1, Female = 0
3	Chest Pain (CP)	a person experiencing different types of chest discomfort	Asymptomatic = 4, Typical angina = 1, A typical angina = 2, Non-angina = 3
4	Resting Blood-Pressure (BPS)	High blood pressure under resting condition	Integer or float value
5	Cholesterol	Serum-Cholesterol	Integer or float value
6	Fasting blood sugar (FBS)	Fasting-Blood-Sugar is higher than expected	False = 0, True = 1
7	Resting ECG (RES)	EKG/ECG during rest	ST-T Wave-Abnormality = 2, Normal = 0, Left-ventricular hypertrophy = 1
8	Max-Heart-Rate-Achieved (Thal-ch)	Maximum rate you have ever achieved	Integer or float values
9	Exercise-induced angina (Exa)	Exercise-related-angina state	No = 0, yes = 1
10	Old peak	Compared to rest, exercise tempted ST depression.	Integer or float value
11	Slope	Peak workout ST segment slope	Flat = 1, down slopping = 0, upsloping = 2
12	Coronary Artery (ca)	Numerous arteries have been colored using fluoroscopy.	Int or float numbers
13	Thalassemia (Thal)	Normal, fixable, reversible defects	
14	Target value	Heart disease diagnostic	Indicating the dimension of diameter

(*continued*)

a. For each grasshopper in the population:
i. Update the position using grasshopper swarm dynamics:
Position_i = Current_Position + c * (Social_Interaction + Environmental_Factors)
Where c decreases linearly over iterations to balance exploration and exploitation.
ii. Introduce randomness using Levy Flight to enhance search diversity:
New_Position = Current_Position + Levy_Step(Random_Vector)
iii. Ensure updated position remains within the hyperparameter search space.
iv. Evaluate fitness of the updated position.
v. Update H* if the fitness of the new position is better than the current best.
b. Adjust parameters to reduce exploration and increase exploitation as iterations progress.
5. End For
6. Return H* as the optimal hyperparameter set.

3.5.3. Combined impact of LFGOA and S–CNN

By Using LFGOA, the hyperparameter tuning process ensures.

- **Efficient exploration** of the hyperparameter space through swarm dynamics and Levy Flight.
- **Avoidance of local optima**, enabling globally optimal solutions for the S–CNN configuration.
- **Improved classification performance**, with hyperparameters tuned specifically for accuracy and robustness.

Together, the S–CNN and LFGOA framework achieves enhanced predictive accuracy, as evidenced by consistent improvements across datasets. The combination of robust feature extraction and efficient hyperparameter optimization positions the SCNN-LFGOA as a highly effective solution for heart disease prediction.

4. Results and discussion

The SCNN-LFGOA prediction models are provided in this section. The initial stages of data preparation and collecting are outlined. The goal function is then described. The evaluation indicators that are utilized to compare performance are thought about one after another. The results and predictions model are then explored for several datasets linked to heart disease.

4.1. Data description

Using datasets linked to cardiac disorders that were gathered by several health sectors, we tested the SCNN-LFGOA model that was built in this study. There is a description of the data collecting, processing, and potential applications processes. The utilization of this information appears to have promise for the creation of data-driven algorithms for forecasting heart disease values. Additionally, Table 4, Table 5, Table 6, and Table 7 provide the statistical descriptions of the four heart-related datasets (i.e., mean, standard deviation, maximum, and minimum). By calculating the correlation coefficients, we were able to obtain the correlation values between the data characteristics. The correlation matrix with respect to different datasets is given below. As there are many different attributes used to evaluate the targeted results. So, a little insight about the attributes that are available in all of the four datasets is given in Table 3.

In order to evaluate the impact of various factors on the development of heart disease and to build a DL-based system (SCNN-LFGOA) for cardiac disease detection, four datasets—CVD (Dataset 1), Framingham (Dataset 2), Heart_UCI (Dataset 3), and heart-data (Dataset 4)—were employed in this study. To perform a thorough assessment of medical traits and clinical pathways used in the diagnosis of heart disease, the study makes use of four datasets. The datasets came from several sources. The datasets included some key medical characteristics such as "Age," "Hyper-tension," "Glucose Levels," "Blood Pressure," "Cholesterol," "Sex," Max Heart Rate Achieved ("Thalach"), "Exercise-Induced Angina ("Ex Ang"), "Old Peak," "Slope," "Coronary Artery ("CA")," "Thalassemia ("Thal")," Chest Pain ("CP")," Resting Blood Pressure ("Trest BP")". Since it makes data analysis easier and increases the precision and speed of the ML or DL algorithms, data preprocessing is a crucial part of the ML/DL life cycle. Because the collected datasets had difficulties with missing values and class imbalance, we applied a variety of pre-processing approaches.

There are 11 features in total in the CVD dataset which also contain the target variable (stroke), some categorical features include "gender," "ever married," "job type," "residence type," and "smoking status" which are also available in the dataset. There are 5110 patient records in total, and the BMI characteristic had 201 missing or null values. Feature correlation may be beneficial in a variety of ways, including evaluating the interdependence of data characteristics and how each feature influences the output feature [30]. Feature correlation among different attributes for dataset 1 can be seen in Fig. 5 below. All of the available

Table 4
Data description (Statistical metrics) for Dataset 1.

Statistics	Gender	Age	Hyper-Tension	Heart Disease State	Ever-Married	Work -Type	Residence- Type	Avg. Glucose level	B.M.I	Stroke
Mean	0.41	43.22	0.097	0.054	0.34	0.83	0.49	106.14	28.893	0.048
Minimum	0.0	14	0.0	0.0	0.0	0.0	0.0	55.12	10.30	0.0
Maximum	2.0	82	1.0	1.0	1.0	4.0	1.0	271.74	97.60	1.0
SD	0.49	22.61	0.296	0.226	0.47	1.1	0.49	45.28	7.854	0.21

Table 5
Data description (Statistical metrics) for Dataset 2.

Statistics	Age	Edu.	Current Smoker	Cigs/Day	BMI	Sex	BP Meds	Prevalent Stroke	Hypertension	Diab.	Chol.	Ten Year CHD
Mean	49.58	1.95	0.49	9.0	25.8	0.42	0.03	0.005	0.31	0.02	236.7	0.15
Minimum	32	1.0	0.0	0.0	15.54	0.0	0.0	0.0	0.0	0.0	107.0	0.0
Maximum	70	4.0	1.0	70	56.8	1.0	1.0	1.0	1.0	1.0	696.0	1.0
Standard Deviation	8.57	1.01	0.5	11.9	4.07	0.49	0.17	0.076	0.46	0.16	44.33	0.36

Table 6
Data description (Statistical metrics) for Dataset 3.

Statistics	Gender	Age	Chest Pain	Resting BP	Cholesterol	Thalach	Old Peak	Slope	Thalassemia	Number
Mean	0.78	53.51	0.75	132.13	199.14	137.55	0.85	0.35	0.3	0.55
Minimum	0.0	28	0.0	0.0	0.0	60.0	−2.60	0.0	0.0	0.0
Maximum	1.0	77	3.0	200	603	202.0	6.2	2.0	2.0	1.0
Standard Deviation	0.40	9.42	0.93	18.44	108.9	25.14	1.05	0.60	0.56	0.5

Table 7
Data description (Statistical metrics) for Dataset 4.

Statistics	Age	Gender	Pain	Resting BP	Cholesterol	Fast. Blood Sugar	Resting ECG	Max Heart Rate	Exang	Old peak	No. of vessels	Target
Mean	54.43	0.7	0.94	131.61	246.0	0.15	0.53	149.11	0.34	1.07	0.75	0.51
Minimum	29	0.0	0.0	94.0	126.0	0.0	0.0	71.0	0.0	0.0	0.0	0.0
Maximum	77	1	3	200.0	564.0	1.0	2.0	202.0	1.0	6.2	4.0	1.0
Standard Deviation	9.07	0.46	1.03	17.52	51.6	0.36	0.52	23.0	0.47	1.17	1.03	0.50

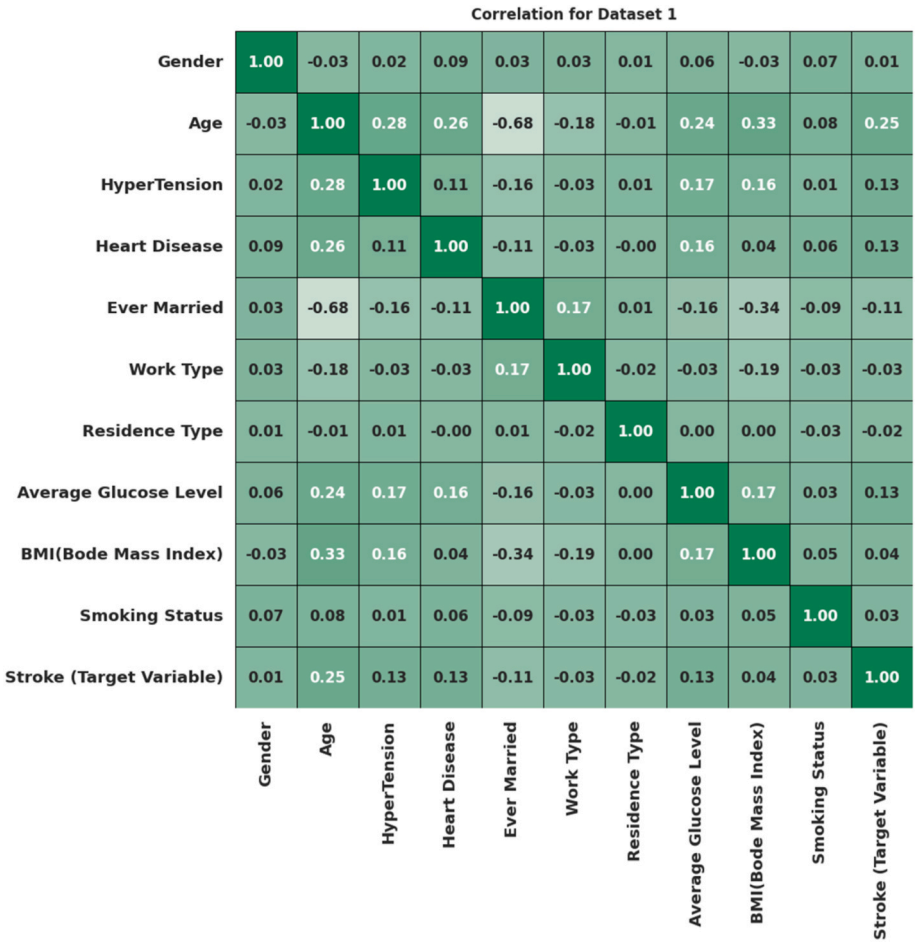


Fig. 5. Featured Data correlation in dataset 1.

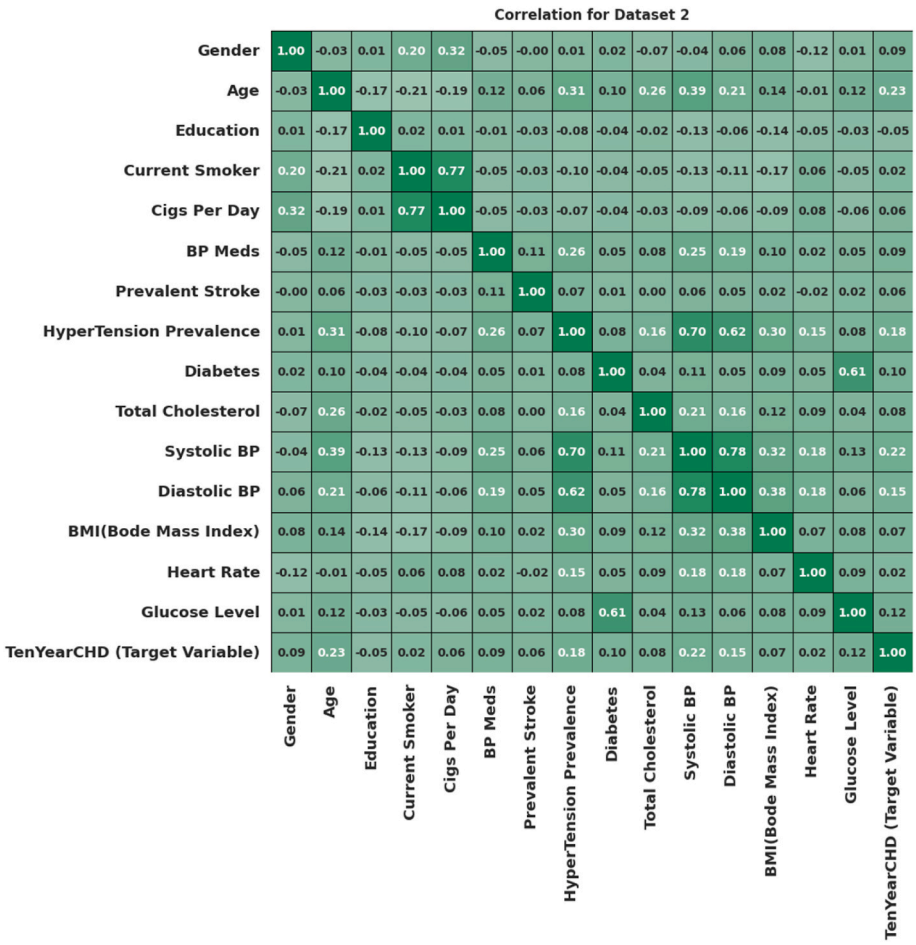


Fig. 6. Featured Data correlation in dataset 2.

features in dataset 1 are exhibiting a favorable connection with the choice and show a positive correlation with respect to the target variable except "Ever married," "Work Type," and "Residence Type" which show a negative relationship with the target variable. Statistics on gender, age, if you've ever been married, work type, housing type, hypertension, heart disease risk, typical blood sugar level, BMI (body mass index), and aim variable stroke are shown in Table 4.

Framingham (dataset 2) has 4240 patient records with a total of 645 null values for the various characteristics. This dataset lacks categorical features; instead, all characteristics have already undergone a numerical transformation that is crucial for deep learning algorithms. Similarly, the features' correlation among different attributes are presented here in Fig. 6. The characteristic "education" in dataset 2 showed negative value, whereas all other features showed positive correlations with respect to the target variable "TenYearCHD." The statistical metrics about the features available in Dataset 2 such as Age, Education, Smoker activity, BMI (Body Mass Index), Sex, Blood Pressure (BP) stroke analysis, Diabetes, Cholesterol, and target variable Ten Year CHD are given in Table 5 below.

There are 920 total patient records in Heart-UCI (dataset 3), and several characteristics had more than 50% of their values missing. Categorical characteristics including "Gender," "CP (Chest Pain)", " FBS (Fasting Blood Sugar)," "RestECG," " ExAng ", "Slope" and " Thal (Thalassemia)" can also be found in this dataset. Also the feature correlation among different attributes are given in Fig. 7. Thalach, CP (Chest Pain), Cholesterol and Slope have the negative connections with target and all other features show positive relationships. The statistical descriptions regarding the dataset for numerical features are given in form of Table 6 below.

In contrast, 1025 entries in Heart-data (dataset 4) had no missing values. A null value can instead indicate that the value is unknown rather than that it does not exist. Moreover, all category properties had previously been transformed to numbers. Similarly, relationships among different attributes can be seen in Fig. 8. It is clear that CP, Rest ECG, Thalach, and Slope showed the positive relationships with target value and all other attributes gave negative correlations with respect to target values. The statistical metrics for numerical attributes such as Age, Sex, CP (Chest Pain), Col (Cholesterol), FBS, ExAng, Old Peak, Slope, Rest-ECG of dataset 4 are given in Table 7.

4.2. *Data preprocessing*

The majority of missing values in healthcare data samples are the result of the inadequate gathering of information, or the doctor may decide to disregard the findings since the patient's medical test is assumed to have a low yield. Although data imputation techniques are effective for coping with missing data, their usage in the medical field is limited, and it is uncertain how useful they are in particular for identifying illnesses [31]. Since standard data imputation techniques fall short of capturing the complexity of missing data in health care applications, the majority of research purposely skip the incomplete cases and do not examine observations with missing value [32].

Moreover, the class distribution in dataset 1 was rather lopsided. In the sample, there were 5110 individuals, however only 249 of them had diseases associated to stroke. There are many categorical attributes available in dataset 1 and after converting them back to numerical values using Label Encoder which is feasible to process through deep neural network. A box plot or whisker plot displays a five-number

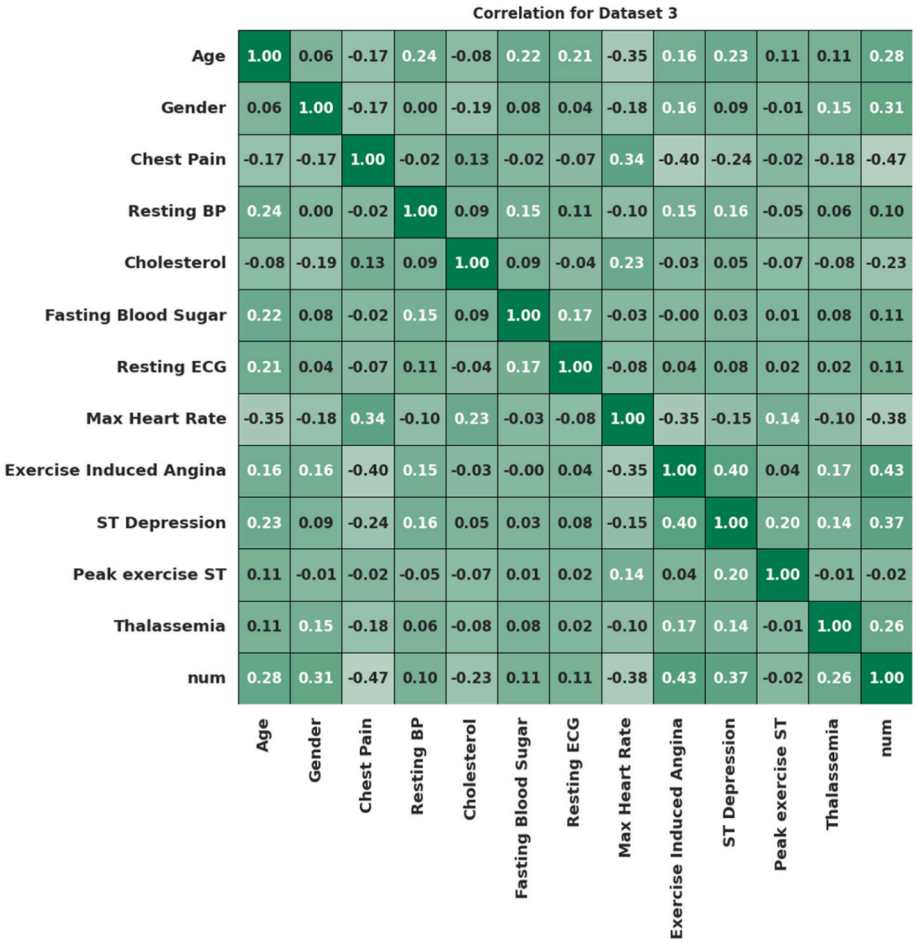


Fig. 7. Featured Data correlation in dataset 3.

overview of a data collection. The five-number summary includes the minimum, first quartile, median, third quartile, and maximum. Fig. 9 displays the box plot for Dataset 1 after being converted into numerical values.

As it can be seen in Fig. 10, there are many outliers present in attributes namely BMI and Average Glucose Level in dataset.

The 644 records out of the 4202 patient records in dataset 2 had a CHD risk. Furthermore, there are categorical features that need to be transformed into numerical data. Fig. 10 shows data distribution using box plot in dataset 2 after numerical transformation.

The attributes, Total Cholesterol, Systolic BP, Diastolic BP, BMI, Heart Rate, Glucose level, have number of outliers presented in these features. But dataset 3 had five unique categories that predicted various characteristics of heart disease and 920 different records were there in dataset. But we changed it into binary classification problem, as all other datasets are also following the binary classification. So, here we use 0 and 1 classes that is used to predict the heart disease status. This dataset 4 has a lot of categorical characteristics, therefore after converting them back to numerical values, display Fig. 11's box plot below to see how the attributes are distributed.

There were 1025 different records and balanced target attributes in dataset 4, also the boxplot is presented in Fig. 12 after transformation of attributes into numerical form. During the training of ML/DL models, classification errors are caused by the dataset's unbalanced composition [33]. To counteract the detrimental effects of unequal data, we used a method known as the "Synthetic Minority Over Sampling Technique (SMOTE)". There were formed two classes, one for minorities and one for majorities. Those who had cardiovascular disease were classified as the majority, while those who had no symptoms were classified as the minority. The remaining 4861 observations in the dataset were classified as the majority, while 1249 observations were classified as the minority. In dataset 2, a similar procedure was utilized to generate 644 random observations from 4202 majority cases while designating the other data records as majority candidates. As the target distribution was uniform and the data records weren't out of balance, no strategy was required for additional datasets.

Few of the features are really important to get the accurate results about heart disease. Ageing results in considerable changes in the heart and blood vessels, according to scientific research. Your heart rate, for instance, is lower when you exercise than it was when you were younger. The National Heart, Lung, and Blood Institute Trusted Source claimed in Ref. [34], a person's likelihood of acquiring heart disease may rise as they age due to physiological changes. Ischemic heart disease, stroke, and renal failure are all recognized to be at risk due to hypertension [35]. The arteries become less elastic as a result of high blood pressure, which limits the quantity of blood and oxygen that can reach the heart and may eventually cause heart disease. Heart disease is more likely to strike young diabetics. Diabetics are more prone to develop heart disease early in life. Diabetes causes the blood arteries that regulate your heart and blood vessels to contract more forcefully, which increases your risk of heart disease [36]. This procedure can eventually lead to a heart attack. The primary goal of this study is to identify medical variables that can increase the accuracy of heart disease prediction.

4.3. *Objective function and evaluation parameters*

To evaluate the accuracy and effectiveness of the proposed approach,

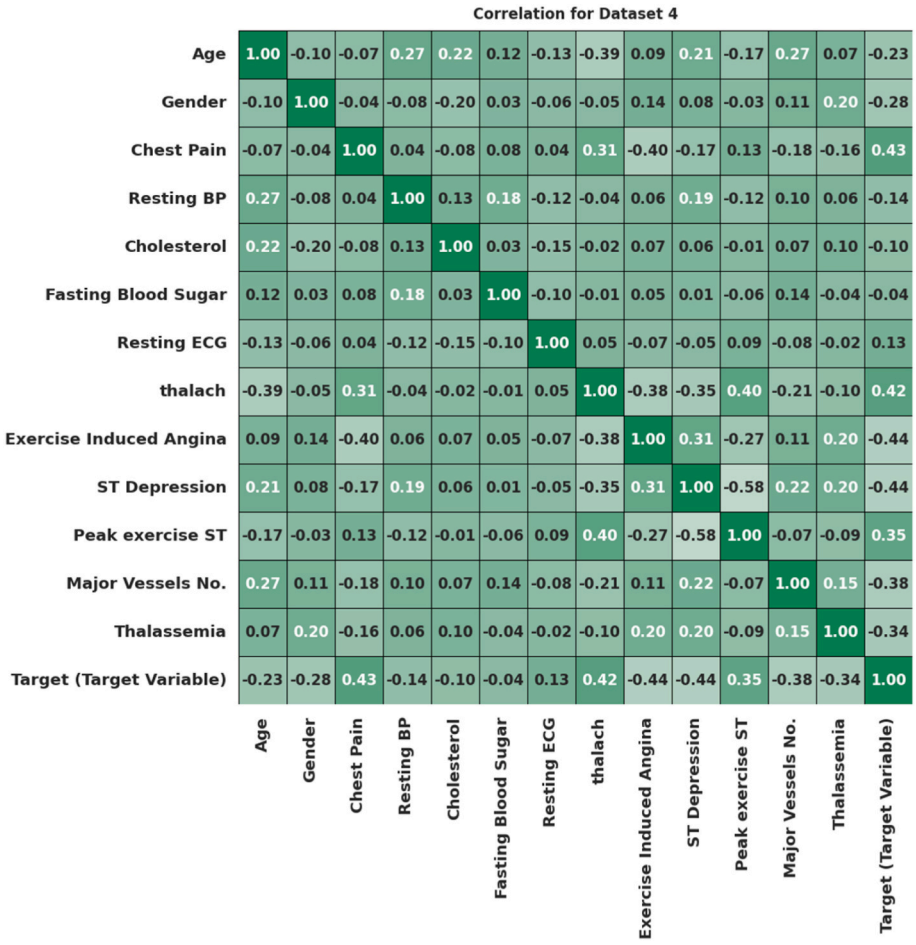


Fig. 8. Featured Data correlation in dataset 4.

we calculate key classification metrics: Precision (also known as Specificity), Recall (Sensitivity), F1-Score (the harmonic mean of precision and recall), and overall Accuracy. These metrics were chosen for their ability to provide a well-rounded assessment of model performance, particularly in the context of heart disease prediction, where both false positives and false negatives can have significant implications for patient outcomes. While metrics like AUC and Matthews Correlation Coefficient (MCC) are also valuable, Precision, Recall, F1-Score, and Accuracy were prioritized to focus on classifying heart disease presence accurately and balancing the trade-off between precision and recall [37].

In machine learning and deep learning, these metrics are derived using the rates of True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN), calculated as follows.

1. **True Positive (TP)**: Correctly classifies patients with heart disease.
2. **True Negative (TN)**: Correctly identifies patients without heart disease.
3. **False Negative (FN)**: Incorrectly classifies patients with heart disease as negative.
4. **False Positive (FP)**: Incorrectly classifies patients without heart disease as positive.

These metrics allow a nuanced understanding of the classifier's performance. Precision and Recall are particularly crucial, as they respectively assess the model's ability to avoid false positives and false negatives. The F1-Score combines these two aspects, offering a balanced measure of the model's reliability. The confusion matrix is also used to visualize these outcomes, enabling a quick assessment of how accurately the model classifies each case. The mathematical definitions of these metrics for the given datasets are presented in Equations (18)–(21).

$$Accuracy = (TP + TN)/(TP + FP + FN + TN) \tag{18}$$

Accuracy represents the overall proportion of correct predictions (both positive and negative) out of all cases. While a high accuracy indicates good overall performance, it does not fully capture the importance of identifying true cases of heart disease, especially when dealing with imbalanced datasets.

$$Precision = (TP)/(TP + FP) \tag{19}$$

Precision focuses on the model's ability to avoid false positives by measuring the proportion of correctly identified positive cases (heart disease) out of all predicted positives. In heart disease prediction, a high precision means that when the model predicts heart disease, it is more likely to be correct, reducing unnecessary stress and further tests for healthy individuals.

$$Recall = (TP)/(TP + FN) \tag{20}$$

Recall measures the model's capacity to correctly identify actual heart disease cases, emphasizing its effectiveness in minimizing false negatives. In medical diagnostics, false negatives are particularly critical, as they represent cases where patients with heart disease are incorrectly classified as healthy, potentially leading to missed or delayed treatments. High recall is therefore essential for patient safety.

$$F1 - Score = 2^{*}(Precision^{*}Recall)/(Precision^{*}Recall) \tag{21}$$

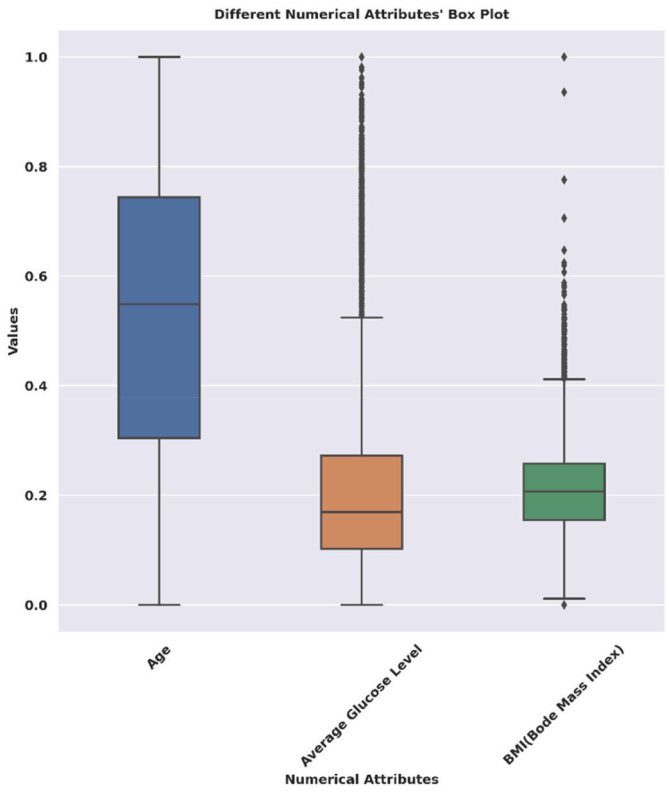


Fig. 9. Box plot of features available in Dataset 1.

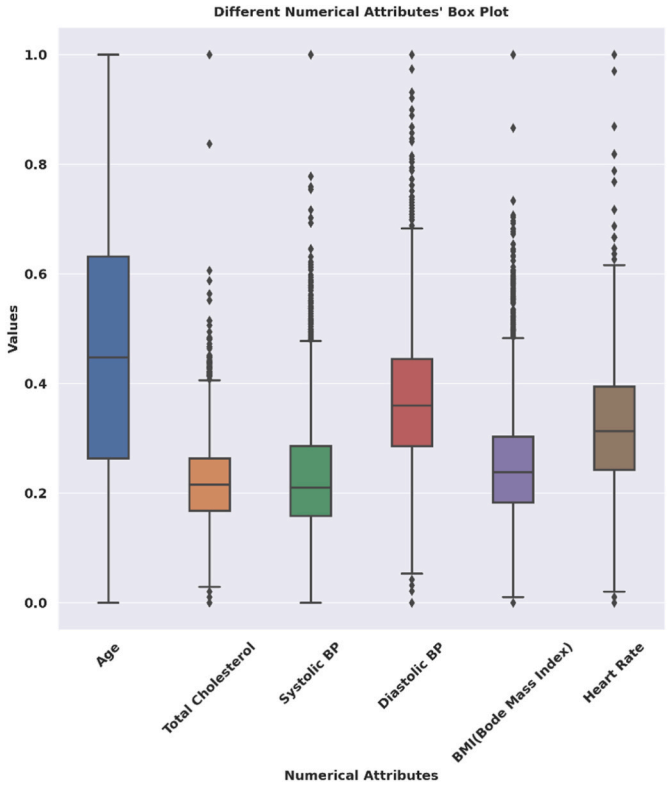


Fig. 10. Box plot of features available in Dataset 2.

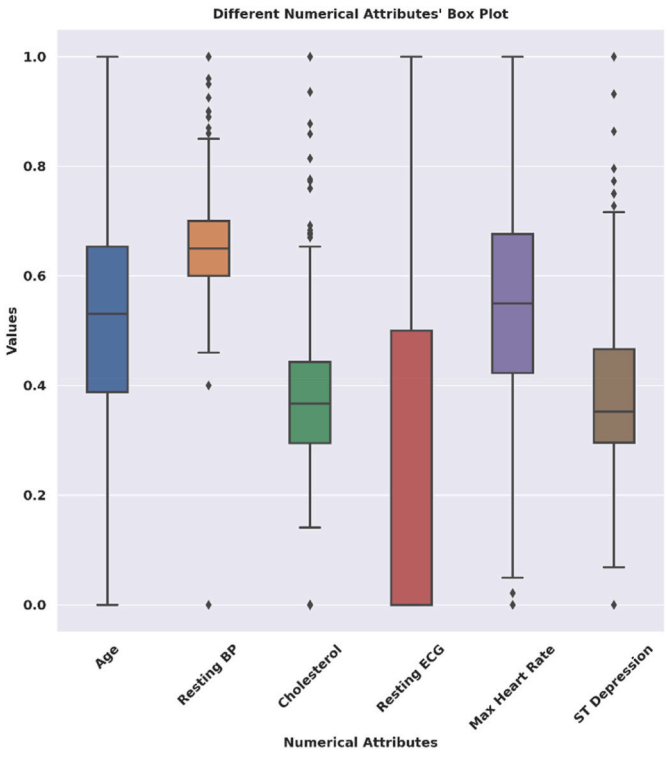


Fig. 11. Box plot of features available in Dataset 3.

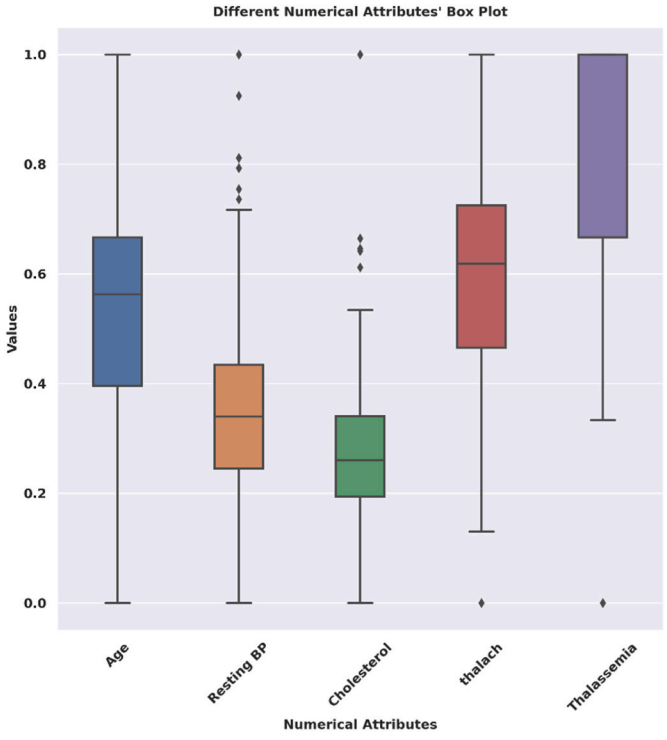


Fig. 12. Box plot of features available in Dataset 4.

The F1-Score is the harmonic mean of precision and recall, balancing the trade-off between false positives and false negatives. This metric is especially valuable in medical contexts where both high precision and high recall are crucial, ensuring that the model not only identifies true cases of heart disease but also minimizes misclassifications.

These metrics collectively offer a comprehensive assessment of the model's performance in medical predictions, with a particular focus on recall to ensure that critical cases of heart disease are accurately identified.

Table 8
Evaluation Results generated on dataset 1.

Metrics	Proposed Model (SCNN-LFGOA)	KNN	RF	SVM	NN	Logistic Regression
Accuracy	0.98	0.94	0.96	0.88	0.92	0.78
F1-Score	0.98	0.93	0.96	0.89	0.91	0.77
Precision	0.99	0.94	0.95	0.88	0.90	0.79
Recall	0.98	0.93	0.96	0.87	0.92	0.80

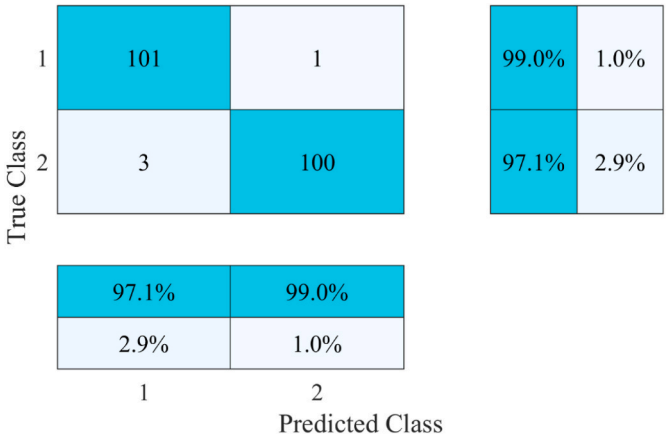


Fig. 13. Confusion matrix for proposed model regarding dataset 1.

5. Evaluation of proposed model on datasets

In this study, all ML/DL models were evaluated to predict binary disease outcomes using four heart disease datasets, with each dataset split into 80% training and 20% testing. To further assess model robustness and ensure it generalizes well across different datasets, a 5-fold cross-validation was employed. This technique reduces the likelihood of overfitting, particularly given the high consistency in accuracy across datasets, and strengthens the reliability of the SCNN-LFGOA model. Cross-validation also provides a comprehensive evaluation by training the model multiple times on different data partitions, thus confirming the model's adaptability and accuracy across varied heart disease datasets.

Looking at the categorization results, in Table 8, proposed model (SCNN-LFGOA) for dataset 1 achieved the highest accuracy of 0.98%, F1-Score with 0.98%, Precision with 0.99%, and Recall with 0.98%. Other classifiers, such as KNN, SVM (Support Vector Machine), and RF (Random Forest), NN (Neural Network), LR (Logistic Regression) performed well and delivered good prediction accuracy. It can be seen clearly that the proposed model performed well with given dataset.

In contrast, 97 out of 99 properly identified the class 1 number, and 100 out of 106 correctly forecasted the class 0 value. Confusion matrix for predicted results regarding dataset 1 is shown in Fig. 13.

Also, the results shown for different evaluating metrics for dataset 1 in form of bar chart is illustrated in Fig. 14:

In Table 9, The metrics used to check the accuracy of proposed model for dataset 2, proposed model (SCNN-LFGOA) for dataset 2 achieved the highest accuracy of 0.99%, F1-Score with 0.98%, Precision with 0.99%, and Recall with 0.99%. Other classifiers namely; KNN, RF and NN performed relatively well on dataset 2 but LR and SVM, performed worse and delivered relatively poor accuracy. It can be deduced that proposed model performed really well on given dataset.

The confusion matrix shown in Fig. 15, demonstrates how accurately our model anticipated the value. About class 0, 99 out of 102 right predictions were given, and for class 1, 100 out of 103 correct predictions. The bar chart for evaluation metrics is shown in Fig. 16.

In Table 10, which lists the metrics that were used to assess the

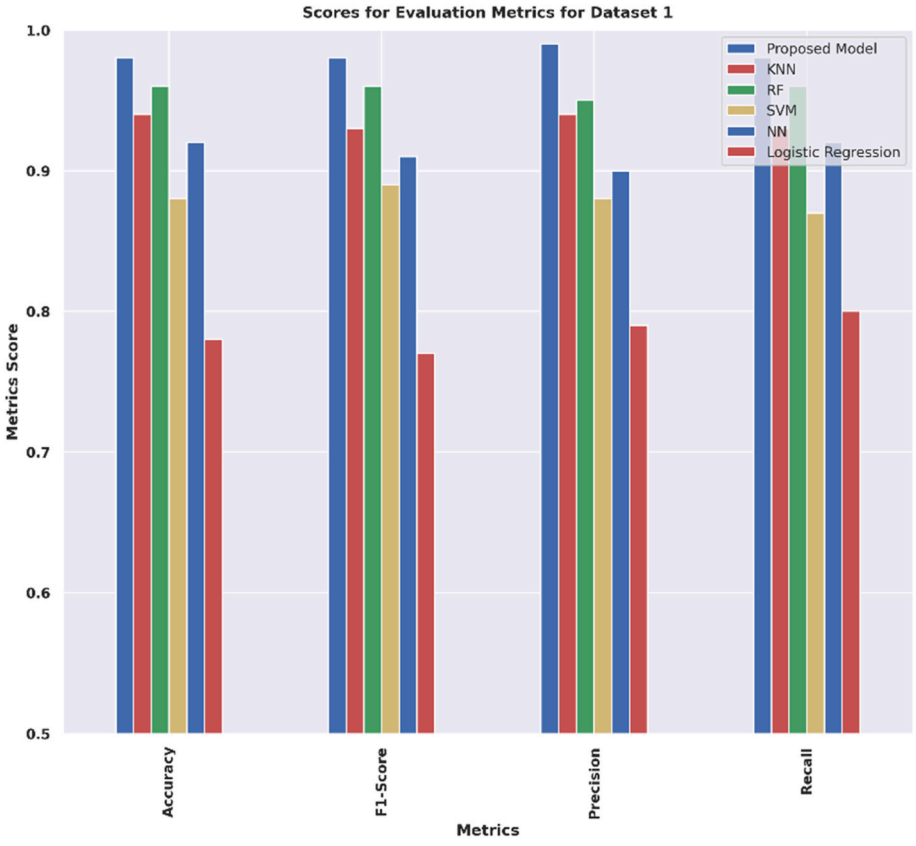


Fig. 14. Bar chart of all evaluated metrics on dataset 1.

Table 9
Evaluation Results generated on dataset 2.

Metrics	Proposed Model (SCNN-LFGOA)	KNN	RF	SVM	NN	Logistic Regression
Accuracy	0.99	0.80	0.95	0.64	0.94	0.60
F1-Score	0.98	0.81	0.95	0.63	0.93	0.61
Precision	0.99	0.80	0.94	0.64	0.94	0.62
Recall	0.99	0.82	0.95	0.66	0.92	0.61

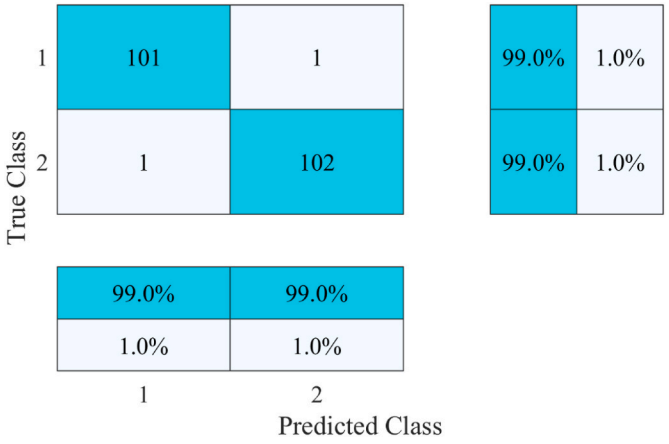


Fig. 15. Confusion matrix for proposed model regarding dataset 2.

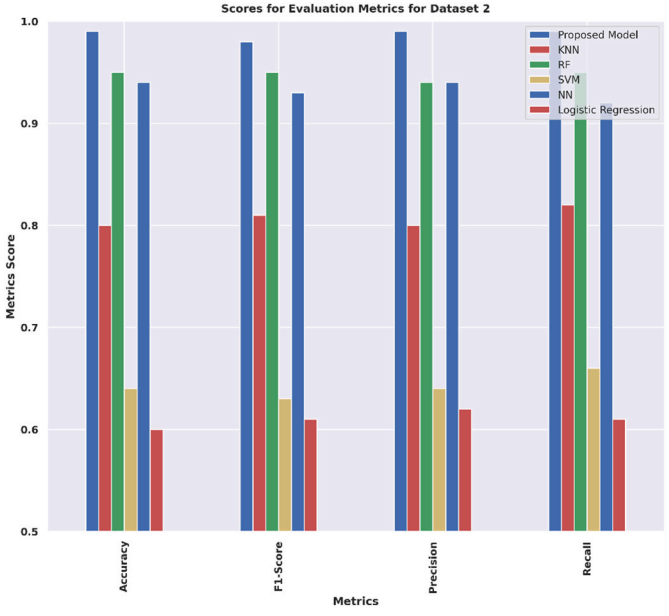


Fig. 16. Bar chart of all evaluated metrics on dataset 2.

Table 10
Evaluation Results generated on dataset 3.

Metrics	Proposed Model (SCNN-LFGOA)	KNN	RF	SVM	NN	Logistic Regression
Accuracy	0.99	0.79	0.97	0.72	0.92	0.64
F1-Score	0.99	0.78	0.96	0.71	0.91	0.63
Precision	0.98	0.79	0.96	0.73	0.90	0.62
Recall	0.99	0.79	0.97	0.72	0.92	0.64

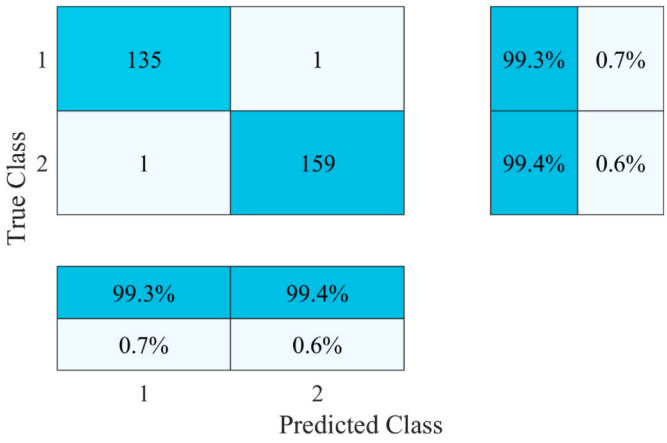


Fig. 17. Confusion matrix for proposed model regarding dataset 3.

proposed model's accuracy for dataset 3, the proposed model (SCNN-LFGOA) for dataset 3 received the best marks for accuracy, F1-Score, Precision, and Recall, all of which were 0.99%. For dataset 3, RF and NN did very well, although LR, SVM, and KNN did poorly and provided relatively low accuracy. It is evident that the suggested model worked quite well on the provided dataset.

The confusion matrix shown in Fig. 17 shows how well our model predicts the value. 136 out of 136 predictions for class 0 were right, while 160 out of 160 predictions for class 1 were accurate.

The assessment metrics bar chart is shown in Fig. 18.

The suggested model (SCNN-LFGOA) for dataset 4 earned the highest scores for accuracy which is 0.98%, F1-Score with 0.98%, Precision with 0.97, and recall as shown in Table 11, which provides the metrics that were used to assess the proposed model's correctness. In contrast to LR, SVM, KNN, RF and NN which performed relatively poor and had low accuracy for dataset 4. From the presented dataset, the recommended model performed fairly well.

The confusion matrix shown in Fig. 19 shows how well our model predicts the value. 102 out of 105 predictions for class 0 were right, while 100 out of 100 predictions for class 1 were accurate. Fig. 20 illustrates the bar chart of the evaluated metrics for dataset 4.

5.1. Comparative analysis

Four reputable research are contrasted with the suggested method to ascertain its efficacy. The analysis of the research has thus been looked at. The overall comparative analysis is given in Table 12 where the comparative models, paper reference code along with evaluation results on different datasets using the suggested models and our proposed models are presented.

In the case of CVD dataset, the results of accuracy, precision and F1-score are seen to perform better than the MLP-EMBDA model presented in Ref. [22] while the recall metric is the same i.e. 98%. In the case of Framingham dataset, the study in Ref. [23] utilized a multilayered perceptron to evaluate the performance metrics. The proposed algorithm in this paper was able to greatly outperform the metric results by up to 28% increase. In the case of Heart_UCI dataset, authors in Ref. [24] predicted early heart disease by employing a gradient boosted tree model to produce metric results of up to 94% but the current model proposed was able to outperform these results by an increase of 5%. In the case of heart data dataset, the paper presented in Ref. [25] proposed a cluster-based bidirectional LSTM model. Although the model was able to produce high accuracy and precision results, the proposed SCNN-LFGOA was able to demonstrate far better results in terms of all four of the evaluation metrics.

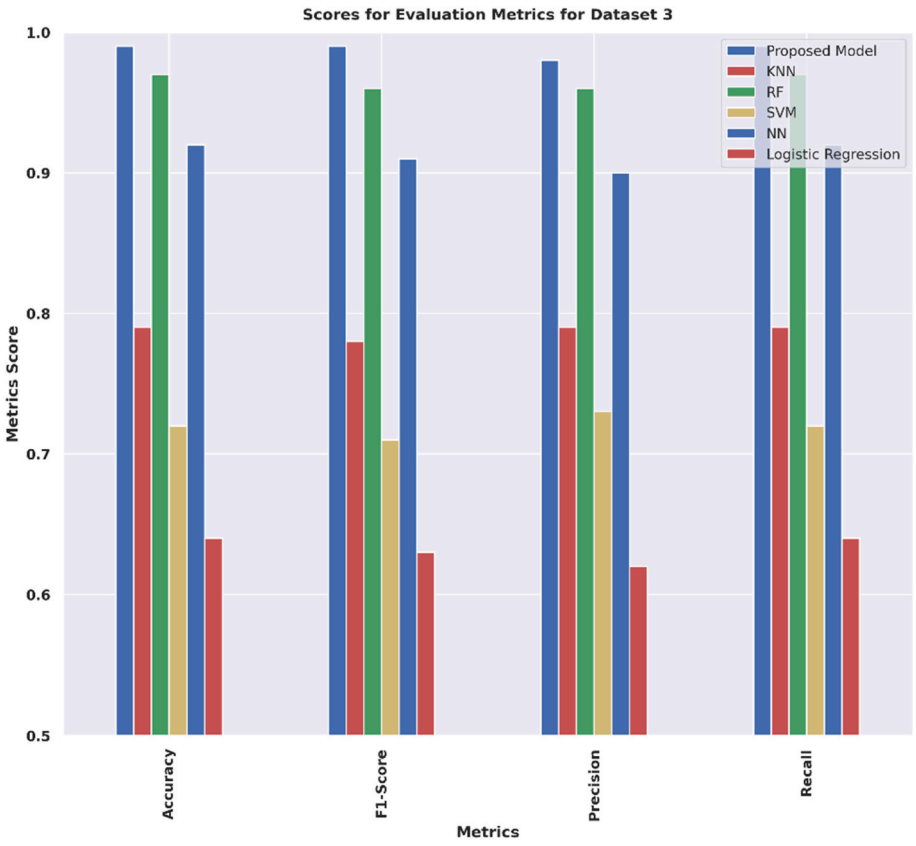


Fig. 18. Bar chart of all evaluated metrics on dataset 3.

Table 11
Evaluation Results generated on dataset 4.

Metrics	Proposed Model (SCNN-LFGOA)	KNN	RF	SVM	NN	Logistic Regression
Accuracy	0.98	0.88	0.95	0.78	0.90	0.80
F1-Score	0.98	0.87	0.95	0.78	0.91	0.82
Precision	0.97	0.84	0.94	0.72	0.92	0.81
Recall	0.98	0.82	0.95	0.70	0.90	0.79

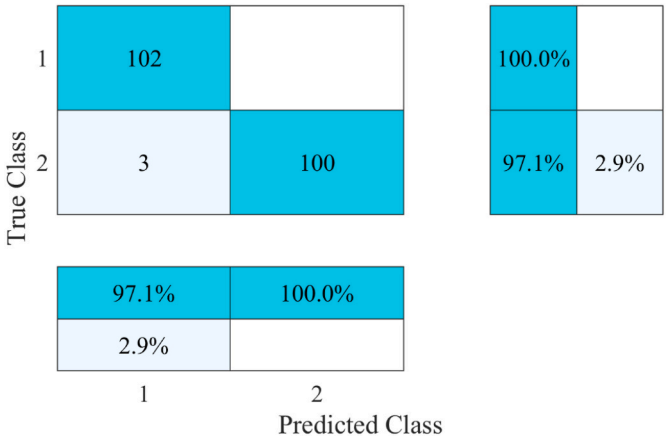


Fig. 19. Confusion matrix for proposed model regarding dataset 4.

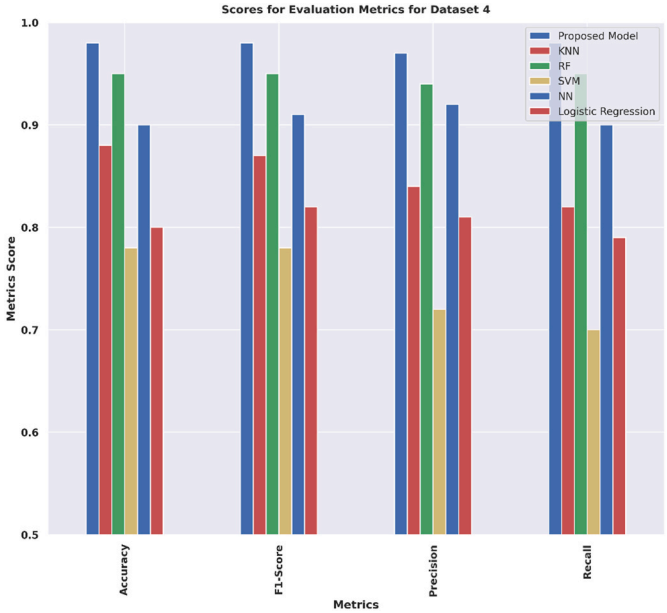


Fig. 20. Bar chart of all evaluated metrics on dataset 4.

6. Conclusions

Heart disease prediction has become a critical focus in healthcare, driven by the need for timely and accurate diagnostic tools. This study proposes a novel hybrid model, the Stacked Convolutional Neural Network with Levy Flight-based Grasshopper Optimization Algorithm (SCNN-LFGOA), designed to achieve high prediction accuracy with low

Table 12
Comparative Analysis of different techniques with proposed model.

Paper Ref ID	Dataset	Proposed Technique	Suggested Model evaluated Metrics	Proposed Model evaluation metrics
[22]	CVD (Dataset 1)	MLP-EMBDA	Accuracy = 0.97 F1-Score = 0.96 Precision = 0.95 Recall = 0.98	Accuracy = 0.98 F1-Score = 0.98 Precision = 0.99 Recall = 0.98
[23]	Framingham (Dataset 2)	Multilayered Perceptron	Accuracy = 0.71 F1-Score = 0.71 Precision = 0.70 Recall = 0.71	Accuracy = 0.99 F1-Score = 0.98 Precision = 0.99 Recall = 0.99
[24]	Heart_UCI (Dataset 3)	Gradient Boosted Tree	Accuracy = 0.94 F1-Score = 0.95 Precision = 0.95 Recall = 0.94	Accuracy = 0.99 F1-Score = 0.99 Precision = 0.98 Recall = 0.99
[25]	Heart_Data (Dataset 4)	Cluster-based Bidirectional LSTM (C–Bi-LSTM) algorithm	Accuracy = 0.95 F1-Score = 0.93 Precision = 0.96 Recall = 0.94	Accuracy = 0.98 F1-Score = 0.98 Precision = 0.97 Recall = 0.98

computational complexity. The SCNN-LFGOA model combines advanced feature extraction capabilities with an optimization method that effectively fine-tunes hyperparameters, reducing the risk of overfitting and ensuring robust performance across datasets. By leveraging four datasets—Cardiovascular Disease, Framingham, Heart-UCI, and Heart-Data—our model demonstrates substantial improvements over traditional methods like Regression Trees, SVM, Logistic Regression, KNN, and Neural Networks. The SCNN-LFGOA achieved an average accuracy of 98.6%, precision of 0.97, recall of 0.98, and F1-score of 0.98, underscoring its superiority in heart disease classification. In addition to its high accuracy, the SCNN-LFGOA model is an advancement in heart disease prediction due to its adaptability across diverse datasets, as evidenced by its classification accuracy rates of 98.49%, 99%, 99%, and 98% across the four datasets. This versatility makes the model particularly valuable for clinical applications where data variability is a common challenge.

6.1. Limitations and future directions

While the SCNN-LFGOA model demonstrates strong performance, there are challenges that warrant further exploration. One limitation is the computational cost associated with optimization, which could impact scalability in real-world settings. Future work will focus on optimizing the model for larger, high-dimensional datasets and exploring its effectiveness in predicting early-stage heart disease. Additionally, efforts will be made to refine the model's generalizability, aiming to develop a tool that can proactively aid in preventing heart disease across diverse patient populations.

Declaration of competing interest

None.

Acknowledgments

This research is supported by the Artificial Intelligence, Biomechatronics and Collaborative Robotics research group at the Top Research Center Mechatronics (TRCM), University of Agder (UiA), Norway.

Data availability

We have shared links of datasets used in this work.

References

[1] S. Nalluri, R. Vijaya Saraswathi, S. Ramasubbareddy, K. Govinda, E. Swetha, Chronic heart disease prediction using data mining techniques, in: Data Engineering and Communication Technology: Proceedings of 3rd ICDECT-2K19, Springer, 2020, pp. 903–912, https://doi.org/10.1007/978-981-15-1097-7_76.
[2] M. Lu, et al., Health care costs of cardiovascular disease in China: a machine learning-based cross-sectional study, Front. Public Health 11 (2023) 1301276, https://doi.org/10.3389/fpubh.2023.1301276.
[3] G.S. Bhavekar, A. Das Goswami, C.P. Vasantrao, A.K. Gaikwad, A.V. Zade, H. Vyawahare, Heart disease prediction using machine learning, deep Learning and optimization techniques-A semantic review, Multimed. Tool. Appl. (2024) 1–28, https://doi.org/10.1007/s11042-024-19680-0.
[4] M. Badawy, N. Ramadan, H.A. Hefny, Healthcare predictive analytics using machine learning and deep learning techniques: a survey, Journal of Electrical Systems and Information Technology 10 (1) (2023) 40, https://doi.org/10.1186/s43067-023-00108-y.
[5] C. Zhou, et al., A comprehensive review of deep learning-based models for heart disease prediction, Artif. Intell. Rev. 57 (10) (2024) 263, https://doi.org/10.1007/s10462-024-10899-9.
[6] C.M. Bhatt, P. Patel, T. Ghetia, P.L. Mazzeo, Effective heart disease prediction using machine learning techniques, Algorithms 16 (2) (2023) 88, https://doi.org/10.3390/a16020088.
[7] R.C. Ripan, et al., A data-driven heart disease prediction model through K-means clustering-based anomaly detection, SN Computer Science 2 (2) (2021) 112, https://doi.org/10.1007/s42979-021-00518-7.
[8] M. Chu, P. Wu, G. Li, W. Yang, J.L. Gutiérrez-Chico, S. Tu, Advances in diagnosis, therapy, and prognosis of coronary artery disease powered by deep learning algorithms, JACC (J. Am. Coll. Cardiol.): Asia 3 (1) (2023) 1–14, https://doi.org/10.1016/j.jacasi.2022.12.005.
[9] J. Miah, D.M. Ca, M.A. Sayed, E.R. Lipu, F. Mahmud, S.Y. Arafat, Improving cardiovascular disease prediction through comparative analysis of machine learning models: a case study on myocardial infarction, in: 2023 15th International Conference on Innovations in Information Technology (IIT), IEEE, 2023, pp. 49–54, https://doi.org/10.1109/IIT59782.2023.10366476.
[10] B.S. Shukur, M.M. Mijwil, Involving machine learning techniques in heart disease diagnosis: a performance analysis, Int. J. Electr. Comput. Eng. 13 (2) (2023) 2177, https://doi.org/10.11591/ijece.v13i2.pp2177-2185.
[11] E. Dritsas, M. Trigka, Efficient data-driven machine learning models for cardiovascular diseases risk prediction, Sensors 23 (3) (2023) 1161, https://doi.org/10.3390/s23031161.
[12] M.A. Bansal, D.R. Sharma, D.M. Kathuria, A systematic review on data scarcity problem in deep learning: solution and applications, ACM Comput. Surv. 54 (10s) (2022) 1–29, https://doi.org/10.1145/3502287.
[13] A. Moslemi, A tutorial-based survey on feature selection: recent advancements on feature selection, Eng. Appl. Artif. Intell. 126 (2023) 107136, https://doi.org/10.1016/j.engappai.2023.107136.
[14] T.P. Gupte, et al., A plasma proteomic signature for atherosclerotic cardiovascular disease risk prediction in the UK Biobank cohort, medRxiv (2024), https://doi.org/10.1101/2024.09.13.24313652.
[15] S. Kabiri Rad, V. Afshin, S. Zahiri, An effective heart disease prediction model using deep learning-based dimensionality reduction on imbalanced data, J. Electr. Comput. Eng. Innovat. (2024), https://doi.org/10.22061/jecei.2024.10847.742.
[16] V. Chithra, H. Shashank, D.S. Patel, Heart disease detection using ensemble machine learning technique based on various risk factors, in: 2024 Second International Conference on Networks, Multimedia and Information Technology (NMITCON), IEEE, 2024, pp. 1–8, https://doi.org/10.1109/NMITCON62075.2024.10698891.
[17] M. Ashrafuzzaman, S. Saha, K. Nur, Prediction of stroke disease using deep CNN based approach, J. Adv. Inf. Technol. 13 (6) (2022), https://doi.org/10.12720/jait.13.6.604-613.
[18] H. Khan, N. Javaid, T. Bashir, M. Akbar, N. Alrajeh, S. Aslam, Heart disease prediction using novel ensemble and blending based cardiovascular disease detection networks: EnsCVDD-net and BlCVDD-net, IEEE Access (2024), https://doi.org/10.1109/ACCESS.2024.3421241.
[19] T.J. Saleem, M.A. Chishti, Deep learning for the internet of things: potential benefits and use-cases, Digital Communications and Networks 7 (4) (2021) 526–542, https://doi.org/10.1016/j.dcan.2020.12.002.
[20] H. Sahu, et al., Analysis of deep learning methods for healthcare sector-medical imaging disease detection, Contemp. Math. (2023) 830–852.

[21] P. Bizopoulos, D. Koutsouris, Deep learning in cardiology, IEEE reviews in biomedical engineering 12 (2018) 168–193, https://doi.org/10.1109/RBME.2018.2885714.
[22] D. Deepika, N. Balaji, Effective heart disease prediction using novel MLP-EBMDA approach, Biomed. Signal Process Control 72 (2022) 103318, https://doi.org/10.1016/j.bspc.2021.103318.
[23] M.S. Pathan, A. Nag, M.M. Pathan, S. Dev, Analyzing the impact of feature selection on the accuracy of heart disease prediction, Healthcare Analytics 2 (2022) 100060, https://doi.org/10.1016/j.health.2022.100060.
[24] C.A.u. Hassan, et al., Effectively predicting the presence of coronary heart disease using machine learning classifiers, Sensors 22 (19) (2022) 7227, https://doi.org/10.3390/s22197227.
[25] P. Dileep, et al., An automatic heart disease prediction using cluster-based bi-directional LSTM (C-BiLSTM) algorithm, Neural Comput. Appl. 35 (10) (2023) 7253–7266, https://doi.org/10.1007/s00521-022-07064-0.
[26] Y. Meraihi, A.B. Gabis, S. Mirjalili, A. Ramdane-Cherif, Grasshopper optimization algorithm: theory, variants, and applications, IEEE Access 9 (2021) 50001–50024, https://doi.org/10.1109/ACCESS.2021.3067597.
[27] L. Wu, J. Wu, T. Wang, Enhancing grasshopper optimization algorithm (Goa) with levy flight for engineering applications, Sci. Rep. 13 (1) (2023) 124, https://doi.org/10.1038/s41598-022-27144-4.
[28] F. Panjaitan, S. Nurmaini, and R. U. Partan, "Features Combination of Heart Rate Variability and Grid Search Optimization for Accurate Cardiac Disease Classification," *Available at: SSRN 4707096*.
[29] S.M. Hashemipoor, Optimized Deep CNN Model for Enhanced COVID-19 Detection, Trent University, Canada, 2024.
[30] F.M. Talaat, Revolutionizing cardiovascular health: integrating deep learning techniques for predictive analysis of personal key indicators in heart disease, Neural Comput. Appl. (2024) 1–24, https://doi.org/10.1007/s00521-024-10453-2.
[31] R. Avula, Applications of bayesian statistics in healthcare for improving predictive modeling, decision-making, and adaptive personalized medicine, International Journal of Applied Health Care Analytics 7 (11) (2022) 29–43.
[32] S. Nayak, P.M. Khilar, Data imputation in healthcare applications, in: *AI Healthcare Applications And Security, Ethical, and Legal Considerations*: IGI Global, 2024, pp. 49–67, https://doi.org/10.4018/979-8-3693-7452-8.ch004.
[33] M. Buda, A. Maki, M.A. Mazurowski, A systematic study of the class imbalance problem in convolutional neural networks, Neural Network. 106 (2018) 249–259, https://doi.org/10.1016/j.neunet.2018.07.011.
[34] C. Álvarez-Martín, F.F. Caballero, R. de la Iglesia, E. Alonso-Aperte, Association of MC4R rs17782313 genotype with energy intake and appetite: a systematic review and meta-analysis, Nutr. Rev. 00 (0) (2024) 1–16, https://doi.org/10.1093/nutrit/nuae075.
[35] M. Van Den Bogaart, M. Snaterse, T. Van Trier, F. Martens, S. Boekholdt, H. Jorstad, Associations of smoking status with long-term cardiovascular events in different arterial territories: the EPIC Norfolk prospective population cohort, Eur. Heart J. 45 (Supplement_1) (2024), https://doi.org/10.1093/eurheartj/ehae666.2878.
[36] T. Zhou, Vascular Risk Factor Treatment and Control for Stroke Prevention, 2024.
[37] P.M. Awua, Y.M. Malgwi, H.A. Saidu, Predictive modeling for early detection of cardiovascular diseases using machine learning, International Journal of Computer Science and Mathematical Theory (IJCSMT) 10 (No.4) (2024).