Greylag Goose Optimization (GGO) Algorithm for Classifying Lung Cancer Disease

Hussam Elbehiery[1], Amira Hassan Abed[2], Moshira A. Ebrahim[3], Ahmed Abdelhafeez[1,4], Mohamed Eassa[1,4], Hadeer Mahmoud[5, *]
[1]Department of Artificial Intelligence, Faculty of Information System and Computer Science, October 6 University, Cairo, Egypt
[2]Department of Business Informatics, Faculty of Business Administration, Al Ryada University for Science and Technology, Cairo, Egypt
[3]Department of Computer Engineering and Information Technology, Modern Academy for Engineering and Technology, Cairo, Egypt
[4]Applied Science Research Center. Applied Science Private University, Amman, Jordan
[5]Department of Artificial Intelligence, Faculty of Computers and Artificial Intelligence, Modern University for Technology & Information, Cairo, Egypt
E-mail: H.elbehiry.csis@o6u.edu.eg, Amira.Hassan@rst.edu.eg, mushira.ibrahim@eng.modern-academy.edu.eg, aahafeez.scis@o6u.edu.eg, Mohamed.eassa.cs@o6u.edu.eg, hadeer.mahmoud@cs.mti.edu.eg

Keywords: bGGO, optimization, lung cancer, feature selection, MLP, classification

Received: November 2, 2024

It is to this effect that this paper proposes a Greylag Goose Optimization-based algorithm for the improvement of accuracy in case classification for lung cancer. The input data used for this study is prepared by scaling, normalization, and removal of null values. To get an optimal subset of features to improve the classification accuracy, the binary version of the GGO algorithm is compared with six other optimization algorithms: bSC, bMVO, bPSO, bWOA, bGWO, and bFOA, proving the efficacy of bGGO in feature selection. Multi-classification using many classifiers predicts MLP as the superior one with an accuracy of 91.8%. Hyperparameter tuning using GGO enhances the accuracy of MLP to 98.4%. Statistical evaluation with ANOVA and Wilcoxon's signed-rank test establishes the outcome to be highly significant (p < 0.005). The hybrid method of GGO + MLP reveals better robustness and efficien.

Povzetek: Hibridna metoda GGO+MLP je razvita za klasifikacijo pljučnega raka, ki z optimizacijo izbire značilk in hiperparametrov doseže kvalitetno klasifikacijo.

1 Introduction

Lung cancer is a very critical and life-taking disease globally. According to the most recent estimates by the World Health Organization, more than 7.6 million deaths annually occur worldwide because of lung cancer. Moreover, the total incidence cases of cancer are projected to increase to nearly 17 million cases in 2030[1]. Screening of cancer at an initial stage is very crucial because it normally metastasizes and becomes incurable once it has significantly spread. Diagnosis of lung cancer is challenging because the symptoms only appear at an advanced stage where successful treatment outcomes become very hard to achieve. Imaging techniques used in taking images of the lungs for assessment include magnetic resonance imaging, Positron Emission Tomography, Computed Tomography, and X-ray. Lung cancer detection makes use of image processing and deep learning techniques where precision can be enhanced by implementing these methodologies. Detection and identification of the shape, dimension, and position of a tumor is a difficult task. The timing of its detection is very important so that medical interventions can be done on time. However, radiologists still face difficulty in distinguishing between malignant and benign nodules. Identifying malignant nodules from benign ones through naked-eye appearance is subjective and the results differ among multiple observers and cases. Overall, the accuracy of classification for nodules among expert radiologists was more significant compared to that by non-expert radiologists. The idea of getting accurate, reliable, and unbiased analysis motivated the development of computer-aided diagnosis systems [3].

Successful application of ML techniques and neural networks in lung cancer classification was performed as in [4,5]. Very promising models could be obtained in applying machine learning to both classification and regression by using optimizers and ensemble regression models. Techniques developed for biomedical image recognition can become the very first diagnostic tools that show diseases as in [6]. However, constructing very efficient classifiers with appropriate features for pulmonary nodules is of essence in the construction of dependable CBIR and CAD systems. Conventionally, a CAD system embodies two phases: feature extraction and categorization. CBIR normally makes use of a host of visual elements such as texture, shape, and granulometry to create the search index as reported in. To build effective machine learning architecture, there is a need for skillful combinations of hyper-parameters that can improve accuracy and performance in classification. The manual techniques for solving combinatorial problems are time-consuming, inefficient, and at times overwhelming. Proposals have been made to use metaheuristic algorithms in optimizing the process of determining which hyperparameter combination will result in the highest performance.

These optimization techniques generally get inspiration from some natural phenomenon for global, local, and sometimes random searches for the best solution. Metaheuristic algorithms are recognized to be those techniques that find viable solutions of optimization with very minimal processing power. Swarm intelligence algorithms, which form a subset of Metaheuristics, have already been applied successfully to several complex real-world problems in scientific, engineering, and medical domains especially for lung cancer diagnosis as in [7].

Metaheuristic algorithms are iterative methods that tend to evaluate many possibilities in the quest for the optimal solution. They are applied in the identification of the ideal combination of weights for use in addressing the problems of feature extraction and classification. Greylag Goose Optimization has already been applied to different optimization problems, returning very promising results for health applications like lung cancer diagnosis [8]. In this work, it uses GGO as the metaheuristic optimization strategy. The reason for this choice is that, overall, it provides a good record across similar optimization scenarios. Besides, it includes some features that might benefit it over others in certain aspects. A mix of ML methods with a metaheuristic algorithm seeks high accuracy and performance.

Obviously, performance tuning in disease detection, like lung cancer, may provide benefits by giving better diagnostic accuracy and early initiation of treatment. In this paper, it is proposed to combine MLP with GGO metaheuristic algorithm for better optimization of parameters to enhance its learning and classification capability of complex patterns in the data. This paper is therefore aimed at integrating some preprocessing strategies with the GGO-MLP algorithm to improve the classification accuracy of lung cancer textual data. First, pre-processing of the input data is done in which data gets scaled, normalized, and NULL values get removed. After pre-processing, the next task to be done is to extract the optimum set of features that might enhance the accuracy of lung cancer classification. In GGO method, extraction is made in binary format in search of cancerous lung states. The next process is the classification of lung disease based on extraction of unique features. The classification step involves many classifiers, such as SVC, DTC, RFC, KNC, and MLP. The results proved that MLP is the best classifier. The hyperparameter of the MLP model is tuned by GGO, where its performance was evaluated against six other optimizers: SC, MVO, PSO, WOA, GWO, and FOA. GGO with the MLP model gives the best results for lung cancer classification.

The following important research questions are intended to be addressed by this study:

Does bGGO perform better than current optimization methods when it comes to feature selection for the classification of lung cancer?

In comparison to alternative feature selection techniques, how does bGGO increase classification accuracy?

How does the performance of MLP change when hyperparameter tweaking is done via GGO?

These inquiries direct our examination of the suggested method's efficacy.

This paper is organized as follows: An overview of the state-of-the-art literature is given in Section 2. Section 3 presents a detailed explanation of the suggested methodology.

The discussion of the experimental results is presented in Section 4. The findings and suggestions for the future are presented in Section 8.

2 Related work

Machine learning (ML) and deep learning (DL) have been widely researched for lung cancer classification. Early detection is critical as it can affect the survival rate of the patient. It offers a balanced review of the domain, covering baseline as well as state-of-the-art techniques for lung cancer classification, ranging from image-based models to tabular data driven approaches.

For lung cancer classification, machine learning approaches such as Support Vector Machines (SVM), Neural Networks, and Decision Trees have been in practice for a long time. The latest studies proposed deep learning methods with their variants, especially Convolutional Neural Networks (CNNs), because of their proficiency in extracting spatial features from imaging data. However, CNNs are often redundant with the basis of features and require the generation of a large, labelled dataset.

There is a solution proposed by Tehnan IA Mohamed et al. [9] to this problem. The proposed hybrid CNN and metaheuristic approach using EOSA to find the best weights and biases for the CNN model. Despite successfully enhancing the classification accuracy, the limitations of computationally excessive training on the vast datasets remain.

Zeyu Ren et al. [10] have brought forward the idea of LCGANT being a deep convolutional Generative Adversarial Network (GAN) architecture. It is notable that LCGANT can produce artificial images of lung cancer and the network VGG-DF with transfer learning can classify these images with 98.5% accuracy. Nonetheless, LCGANT is heavily dependent on synthetic data, which is its drawback in terms of generalization to the real world.

For lung cancer classification by using features selection and models optimization, metaheuristic optimization techniques have been used. An optimized Random forest classifier full-back aided by a K-means visualization method to improve the feature selection was proposed [11]. P. Mohamed Shakeel et al. [12] assessed biochemical lung malignancy attributes with an ensemble learning technique guided by a wolf prey algorithm, obtaining a favourable figure of prediction (99.48 percent).

Researchers have conducted several studies on hybrid optimization techniques, like the combination of Whale Optimization Algorithm (WOA) with Support Vector Machines (SVMs). Surbhi Vijh et al. [13], introduced an improved method using WOA and SVM techniques for feature selection to achieve accurate findings for lung cancer detection. WOA-SVM showed outstanding performance in CT scan classification with an accuracy of 95% and sensitivity of 100%. Despite being fast, the algorithms driven by WOA experience issues such as premature convergence.

The proposed Greylag Goose Optimization (GGO) approach outperforms the limitations of existing metaheuristic algorithms by adjusting exploration and exploitation phases dynamically to avoid local optima. Compared to CNN-based approaches relying on spatial feature extraction, GGO-MLP is designed for structured tabular data, reducing computational complexity and improving interpretability. By jointly optimizing feature selection and hyperparameters, GGO-MLP achieves a good balance between accuracy and computational overhead. However, our findings indicate that CNN-based methods are generally plagued by feature redundancy, while traditional metaheuristics are incapable of adapting to dynamic feature spaces. Our strategy optimally balances feature selection and classification performance through GGO-guided hyperparameter tuning. The comparison in Table 1 can demonstrate that PSO-SVM achieves especially superior performance among all these techniques to accurately identify the instances of lung cancer with a 99% specificity and 97.6% accuracy. CNN-based approaches lag with tabular data due to their focus on spatial features. In contrast, metaheuristic algorithms tend to find a solution too rapidly and get stuck, since they fail to change their group dynamics as Group Gradient Optimization (GGO) is able to. This can make them less effective at discovering optimal solutions.

3 Material and methods

3.1 Greylag goose algorithm for optimization (GGO)

Table 1: Comparison summary between EOSA-CNN, WOA_SVM, and PSO-SVM algorithms

Method		Accuracy	Sensitivity	Dataset (Size)	Year
[9]	**EOSA-CNN**	95.2%	93.8%	IQ-OTH/NCCD (500)	2023
[12]	**WOA_SVM**	95.0%	100%	Private (220)	2020
[14]	**PSO-SVM**	97.6%	99.0%	Kaggle (500)	2022
	Proposed	**98.4%**	**97.7%**	Kaggle (284)	2024

The GGO technique generates a set of individuals at random to begin the process, each of which will stand as a possible solution for the problem. These are defined as individuals Yp (p = 1, 2, …, n), where n is the number of the total population of individuals, which forms the GGO population. An objective function, represented by Fn, is chosen for the evaluation of the members of the group. The best solution, represented by Z, is attained from the analysis of an objective function for every individual or agent represented by Yp.

The algorithm, GGO, exhibits dynamic group behavior by splitting the population into two groups—an exploration group of size n1 and an exploitation group of size n2. Automatically, the number of solutions in each group is updated with respect to the optimal solution in every iteration. The exploitation group comprises of n2 agents, while the exploration group includes the remaining n1 agents, as shown in Figure 3. At the start, GGO assigns 50% of its population for exploration and 50% for exploitation. In this scenario, the number of agents in the exploration group will decrease by n1 and increase by n2 in the case of the exploitation group. If there is no change in the objective function value of the best solution for three continuous iterations, then the algorithm will increase the number of agents in the exploration group, n1, to find a different best solution and probably avoid local optima [15].

• Explore Operation: The two important tasks of exploration are locating the more interesting regions in the search space and avoiding falling prematurely onto a local optimum, especially when an optimal solution is being progressed toward.

–Best solution: In this strategy, the geese explorer will seek good new areas to explore around its current position. It does this by iteratively assessing a lot of possible neighbors and then chooses the best solution on how fit that solution is. The following equations are used by the GGO method to do this, modifying the B and D vectors: $B = 2b.m1 - b$ and $D = 2$, throughout iterations while using a parameter that is progressively adjusted from 2 to 0:

$$Y(k+1) = Y*(k) - B.|D.Y*(k) - Y(k)| \tag{1}$$

where Y(k) presents an individual at iteration k. The Y∗(k) denotes the optimal location of the leader (best solution). The Y (k + 1) represents the adjusted location of the individual. The values of m1 and m2 values are randomly changing within the range of 0 to 1.

The equation below will be employed by selecting three random search individuals (paddlings), referred to as YPaddle1, YPaddle2, and YPaddle3, to ensure that the individuals are not influenced by a single leader position, hence promoting greater exploration. If |B| is higher than or equal to 1, the current search agent's position will alter as follows.

$$Y(k+1) = w1*YPaddle1 + p*w2*(YPaddle2 - YPaddle3) + (1-p)*w3*(Y - YPaddle1) \tag{2}$$

where (w1, w2, and w3) values are adjusted and updated within the range of 0 to 2. The parameter p exhibits an exponential decrease and is determined by the equation shown below.

$$p = 1 - \left(\frac{k}{k_{max}}\right)^2 \tag{3}$$

where the iteration number is denoted as “k” and “kmax” specifies the most possible iterations.
The second procedure of updating is as follows for m3 values greater than or equal to 0.5, where the values of b and B vector values are reduced.

$$Y(k+1) = w4 * |Y*(k) - Y(k)|.\, eal.\cos(2\pi l) + [2w1(m4 + m5)] * Y*(k) \tag{4}$$

where a is a fixed value, l is a randomly selected value from a range of −1 to 1. The w4 parameter is adjusted within the range of 0 to 2, whereas m4 and m5 parameters are modified within the range of 0 to 1.
Operation of Exploitation: The task of the exploitation group is to improve existing solutions. At the end of each round, the GGO selects the best fitness individual and gives him due recognition. In achieving its exploitation objective, GGO applies two different strategies, which are described as follows.
– Going concerning the best explanation: The following formula is used to step forward toward the best solution. Three sentries, YS entry1, YS entry2, and YS entry3, instruct other agents YNonS entry to update the locations toward the predicted location of the prey. The following formula shows the process of updating locations.

$$Y1 = YS\ entry1 - B1.\ |D1.YS\ entry1 - Y|,$$
$$Y2 = YS\ entry2 - B2.\ |D1.YS\ entry2 - Y|, \tag{5}$$

$$Y3 = YS\ entry3 - B3.\ |D1.YS\ entry3 - Y|$$

where the (B1, B2, B3) values are determined by the equation B = 2b.m1–b, where b is a constant. Similarly, the values of D1, D2, and D3 are determined by the equation D = 2m2. The modified population locations, Y (k + 1), can be determined by averaging the three solutions: Y1, Y2, and Y3 as follows.

$$Y(k+1) = \overline{Y\iota}|_0^3 \tag{6}$$

The second mathematical lemma that can be utilized in the study of the GGO algorithm during the exploitation process is the Triangle Inequality. Essentially, one of the basic definitions for metric spaces is the Triangle Inequality: for any triangle, the length of one side is less than or equal to the sum of the other two sides. For GGO, it means that any two agents are closer than or equal to the sum of distances of these two agents from the third in the search space. So, a fine solution lies around the optimum response-leader-in flight. Consequently, some agents are incentivized to improve the solution by searching the space around the optimum solution, which is called YFlock1. To overcome the limitation of local optima, GGO makes use of the process affecting local and global optima based on the following equation.

$$Y(k+1) = Y(k) + C(1+p) * w * (Y - YFlock1) \tag{7}$$

The third mathematical construct that can be utilized in the investigation of the GGO algorithm in its effectiveness at removing the issue of local optima is the Law of Large Numbers. It states that the sample average will approach the population average as the sample size becomes large, and therefore is an elementary concept in probability theory. In the context of GGO, what this means is that, eventually, as the size of the swarm goes to infinity, the swarm will converge towards the global optimum.
GGO's one key benefit is that it can tradeoff exploring and exploiting dynamically. It is actually accomplished through an adaptive weight mechanism that gets set at a number of iterations. It is a validated balance among iterations using statistical monitoring of performance improvement rates. Figure 1, which shows the agent reclassification process between exploration and exploitation groups, illustrates the GGO decision-making criteria.

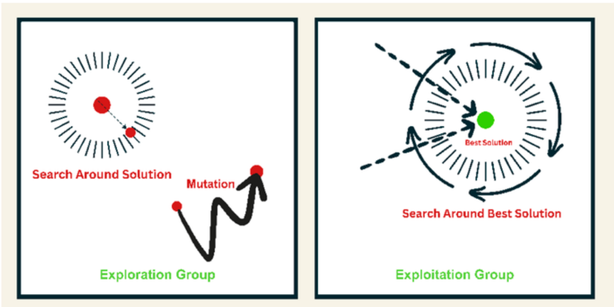

Figure 1: The GGO exploration and exploitation.

The Greylag Goose Optimization algorithm has outstanding exploration capabilities with the introduction of a mutation approach and scanning participants of the group in performing exploration. With such a strong explore ability, GGO will suffer from delays in convergence. Algorithm one is the pseudo-code of GGO. In algorithm one, the GGO is provided with certain basic parameters, such as population size, mutation rate, and number of iterations. It therefore divides the participants into two based on GGO: one group carrying out exploratory labor and another group doing the exploitative labor. In its iterative search for the optimal solution, it will dynamically change the size of each group. To introducing diversity and to ensure that the space has been fully explored, GGO applies a random reordering of the responses at each iteration.
The algorithm randomly selects initializations from a population and assesses the performance with placeholder functions. It walks through random mutations and uses crossover with the best agent. Exploration probability of Dynamic Balancing decreases over time, which will foster exploitation in later phases. GGO dynamically adjusts agent roles to avoid premature convergence and enhance search effectiveness.

A solution parameter from the exploration group can be shifted into the exploitation group in a single iteration. The GGO's self-designed mechanism guarantees the position of the leader during the entire process. The GGO algorithm works toward the procedure for upgrading the roles in the exploitation group (n2) and the exploration group (n1). The parameter m1 is modified iteratively according to the equation m1 = $d(1 - \frac{k}{k_{max}})$, where k depicts the current iteration, d denotes a constant, and k_{max} shows how many iterations there are. GGO, after every iteration, modifies the individuals in the search space and randomizes their order to shift the roles between the groups: exploration versus exploitation. It finally retrieves the optimal solution for a solution space.

Algorithm 1: Proposed GGO algorithm.

1: Initializing GGO population Yj (j = 1, 2, ..., n), population size is n, the total iterations number is kmax, and an objective function Fn.
2: Initializing all GGO parameters b, B, D, a, l, d, w, m1 - m5, w1- w4, B1-B3, D1, - D3
3: Set k = 1
4: Determine each agent's objective function (Fn) Yj.
5: Set Z = greatest agent position
6: Revise the solutions in the groups for exploration (n1) and exploitation (n2).
7: while k ≤ kmax do
8: for (j = 1: j < n1 + 1)
9: if (k%2 == 0) then
10: if (m3< 0.5) then
11: if (|B| < 1) then
12: Update the current search agent position as in Equ (1).
13: else
14: Select three random search agents YPaddle1, YPaddle2, and YPaddle3
15: Update (p) by the exponential form as in Equ (3).
16: Update the current search agent position as in Equ (2).
17: end if (Line11)
18: else
19: Update the current search agent position as in Equ (4).
20: end if (Line10)
21: else
22: Update individual positions as in Equ (7).
23: end if (Line 9)
24: end for (Line 8)
25: for (j = 1: j < n2 + 1)
26: if (k%2 == 0) then
27: compute Y1, Y2, Y3as in Equ (5).
28: Update individual positions as in Equ (6).
29: else
30: Update the current search agent position as in Equ (7).
31: end if (Line 26)
32: end for (Line 24)
33: Calculate Fn for each Yp
34: Update parameters
35: Set k =k + 1
36: Adjust beyond the search space solutions
37: if (Best Fn is the same as the previous two iterations) then
38: Increase the solutions of group (n1)
39: Decrease the solutions of group (n2)
40: end if (Line 37)
41: end while (Line 7)
42: Return best agent Z

Efficiency and convergence speed are balanced by the GGO stopping criterion. There are three circumstances that cause the algorithm to stop. The first is the maximum Iterations that permit exploration and exploitation, it operates for a predetermined number of iterations. The second is the fitness improvement threshold that saves resources, it stops if the optimal fitness does not increase by a certain amount. The diversity-based stagnation detection is deemed stagnant and halts if all agents have comparable feature subsets. By eliminating pointless computations and early convergence, these criteria assist GGO in maintaining a balance between computational efficiency and solution quality.

3.2 The proposed binary GGO optimization algorithm

GGO Optimization Algorithm: This is a better approach to the optimization of the feature selection of MLP parameters. This technique uses a binary format in the selection of features and optimizes the set of used features to improve MLP's performance.

Feature selection problems are limited to binary 0, 1 constituents of a search space, indicating how relevant a certain feature is. Thus, in the binary GGO method, the continuous GGO values that will be proposed in this section are transformed into a binary format, [0, 1], directly relating to the feature selection process. The basic objective of the procedure, as formulated in equations 11 and 12, is to convert the continuous data into binary data using the following Sigmoid function.

$$Bi_t^* = \begin{cases} 1 & if\ sigmoid\ (Bi_t^*) \geq 0.5 \\ 0 & otherwise \end{cases}$$
(8)

$$Sigmoid(Bi_t^*) = \frac{1}{1+e^{-10(Bi_*^i - 0.5)}}$$
(9)

where Bi_t^* reflects the best solution at a given iteration (t). Algorithm 2 presents the proposed bGGO stages used for selecting the optimum feature set, enhancing the Caries' case classification accuracy.

Algorithm 2: bGGO algorithm.

1: Initialize population of GGO, objective function, and GGO parameters
2: Convert the solution to binary [0 or 1]
3: Calculate the objective function for each agent and get the best agent position
4: Update Solutions in the exploration group
5: Update Solutions in the exploitation group
6: while k ≤ kmax do
7: for (j = 1: j < n1 + 1)
8: if (k%2 == 0) then
9: if (m3< 0.5) then
10: if (|B| < 1) then

11: Update the current search agent position in the exploration group

12: else

13: Update the current search agent position based on the three random search agents

14: end if (line 10)

15: else

16: Update the current search agent position

17: end if (line 9)

18: else

19: Update individual positions

20: end if (line 8)

21: end for (line 7)

22: for (j = 1: j < n2 + 1)

23: if (k%2 == 0) then

24: Update the current search agent position in the exploitation group

25: else

26: Update of the current search agent position

27: end if (line 23)

28: end for (line 22)

29: Convert to binary the updated solution

30: Calculate the objective function

31: Update parameters

32: Adjust beyond the search space solutions

33: Update the exploration group Solutions

34: Update the exploitation group Solutions

35: end while (line 6)

36: Return best agent

3.2 Multilayer Perceptron (MLP) Model

Application of the multilayer perceptron model, inspired by the neural architecture of the human brain, has particularly superior performance in nonlinear simulations of complex systems. They can resolve nonlinear structure prediction problems [17]. They learn how to solve a problem and discover relationships that underlie the data. For this purpose, a large amount of data is fed during the training process which is based on the learned relationships to calculate the desired output. The most frequently applied neural network model is the back-propagation network. The neural network has neurons that act in parallel for each layer. Every layer is fully connected with the previous and next layers.

Algorithm 3: Pseudo code of MLP Model.

1-Initialize Multilayer Perceptron MLP with input and output dimensions (number of features), number of hidden layers, dimension of hidden layers, activation function, batch size, Num epochs, and learning rate.

2- Set MX the input matrix, every row in MX is an input vector.

3- Set The output matrix MY, every row in MY is the corresponding output vector.

4- Initialize weights matrices W[l] for l = 1 to L, where each weight W[l][j][k] is a connection between layer l's neuron j and layer l+1's neuron k.

5- Initialize Biases B[l] for l = 1 to L-1, where each bias B[l][j] is the bias for layer l+1's neuron j.

6- Repeat until the error is within the acceptable range or maximum iterations are reached:

7- Forward pass:

8-For l = 1 to L:

9- Calculate the weighted sum for every neuron layer l+1 neurons.

10-Apply activation function A[l] to every layer l+1 neurons:

11-For each sample in the batch:

12-Compute the loss between predicted output and actual label (cancerous or non-cancerous).

13 - Backward pass:

14- For l = L to 1:

15-Compute the gradient of loss concerning the weight matrix Wi and the bias vector bi.16-Update the weights and biases by subtracting the product of the learning rate and the corresponding gradients from the current values for output and hidden layers.

17- End of the training process.

4 The proposed framework

This phase of data processing involves primarily removing null values and normalizing and scaling data. The serious part of the phase is preparation and expansion of the data. This study used feature selection methods to perform 7 optimization methods in binary form which are- GGO [15], Sine Cosine Algorithm (SC) [17], Mean-variance optimization (MVO) [18], Particle Swarm Optimizer (PSO) [19], Whale Optimization Algorithm (WOA) [20], Gray Wolf Optimizer (GWO) [21], Falcon Optimization Algorithm (FOA) [22]. In the second step, the feature selection process is applied, with the proposed feature selection method. Using bGGO, the found features are extracted. The aim of this step is to get the best features that will make the input data be classified with greater accuracy. This step has the benefit of reducing the total number of features by removing irrelevant features. The input data has been classified using ML classifiers and features selected have been chosen based on bGGO. The list of the suggested ML models used in this research includes the SVC, DT, RFC, 1 KNN, and MLP models. The MLP model used must have its parameters optimized to best exploit the proposed optimization technique. Therefore, the selection of the most optimal set of classification parameters is the object in view at this stage of preprocessing.

It generates the first population of solutions for all possible configurations of the parameters. A value of associated fitness is assigned to each Greylag Goose, with respect to the quality of its key regarding the validation set. Individuals search the solution in a systematic way in the population to discover the best solutions of the search space. The GGO optimizes in the same way an iterative action performed by it which refines the decisions step by step towards Pareto dominance to get the optimal configuration of the parameters in MLP. The weighted vectors are used by GGO to drive the people to the correct position, which is calculated by the population's fitness score after every generation. These algorithms keep on changing the positions of individuals in the way of coming very near the optimal action because of the iterative work of GGO. The GGO performs an iterative action that refines the solutions step by step towards the Pareto front to obtain the optimal configuration of MLP parameters. This algorithm stops when either a convergence threshold has been met or after many iterations. Thereafter, the configuration for which the value of fitness is maximum is considered as the best, optimal solution.

This study has developed a model for how GGO can help in enhancing the tuning parameters of MLP. The optimization of the parameters of MLP is associated with achieving the highest level of classification accuracy and performance. The parameters that require optimization should be identified prior to using the GGO to alter the values in MLP. Figure 1 declares a sequential process of the proposed framework.

For GGO and MLP, choosing the right hyperparameters is essential to performance optimization. For GGO population size, it maintains computational efficiency and guarantees exploratory variety, a balanced size was chosen. The exploration and exploitation of these parameters avoids early convergence and increases search efficiency, GGO constantly modifies agent responsibilities. The MLP Structure tests for appropriate feature representation without overfitting were used to identify hidden layers. The ReLU activation function was selected due to its ability to handle non-linearity well. The learning rate is optimized by GGO for faster, oscillation-free convergence. Adam was chosen due to its adaptability, and the batch size was modified to account for generalization and training speed. Multiple hidden layers with optimal hyperparameters and tweaking via GGO are features of the MLP architecture, which increases classification accuracy.

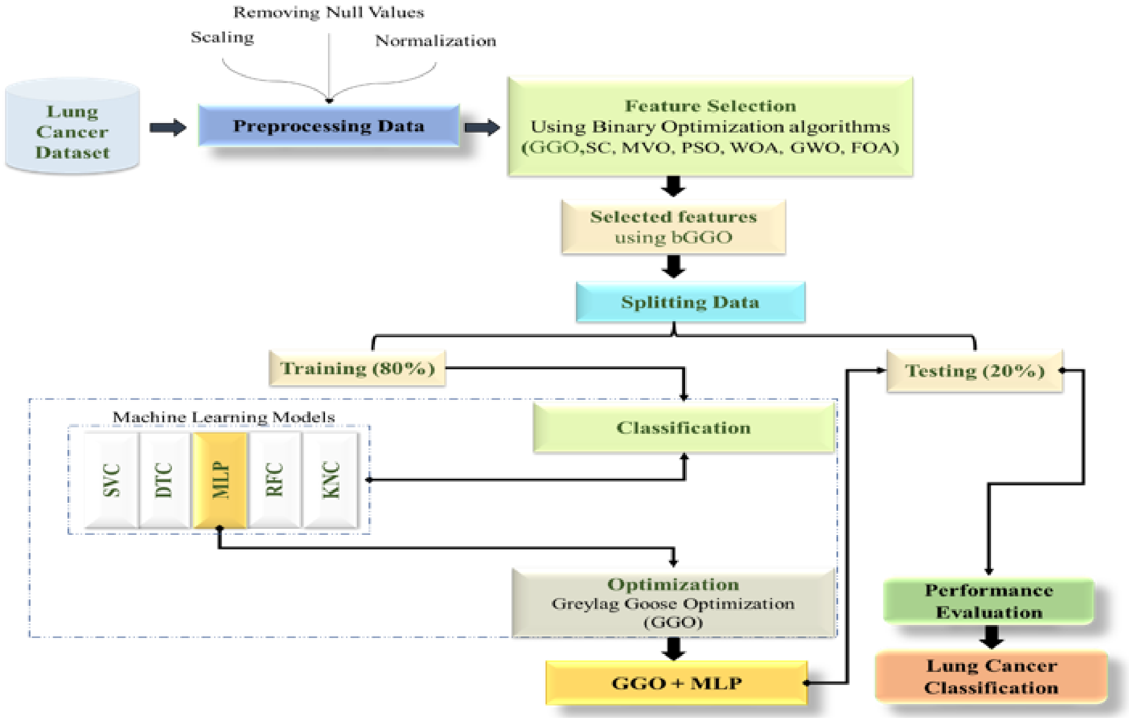

Figure 2: The proposed lung cancer classification framework

5 Experimental results

5.1 Experimental setting

The tests carried out tests assessed the proposed algorithm for various experimental conditions. In these tests, some traditional mathematical functions were used as benchmarks to determine the minimum values for a given range of variables. Such mathematical functions are usually used in literature to compare the performance between an optimization technique against others, and there are many variations of these optimization techniques. The suggested Greylag Goose Optimization algorithm has been compared with six famous optimization techniques to show better performance and efficiency. In the present study, comparisons are drawn among GGO, SC, MVO, PSO, WAO, GWO, and FA. All these algorithms have been chosen due to their popularity and their practical importance, which has made them quite well-known.

Technical specifications of the study platform included: primary memory of 16 GB, Intel Core i7 CPU, and GeForce RTX2070 Super 8 GB RAM GPU. The details of the specification of the software included: Ubuntu 20.04 as the operating system, with TensorFlow 1.15, CUDA9.0, and Cudnn7.1, and Python 3.7 for Spider IDE.

5.2 Dataset description

Machine learning and data science experts can utilize this dataset to make prediction models of the diagnosis of lung cancer, to study the effect of different characteristics related to cancer, and to make algorithms for the treatment and prevention of cancer. This project works with a data set named "Lung Cancer Dataset," collected and uploaded to Kaggle. The pre-enabling classification of cancer and the pre-enabling systems have facilitated various sites through which individuals can check whether they have cancer problems at a throwaway price.

It also aids specialists in making decisions based on the risk profile for cancer. This information is scooped from the website, which is owned by the online lung cancer prediction system, located at: https://www.kaggle.com/datasets/mysarahmadbhat/lung-cancer.

The dataset consists of 284 samples and 15 characteristics. The characteristics contain categorical and numerical features, which were preprocessed with one-hot encoding and normalization as needed. The dataset was divided between training and testing sets using an 80-20% segmented split to ensure class distribution balance. A cross-validation procedure with k folds (k=5) was used to assure the model's resilience, preventing overfitting and allowing a more generic assessment of performance. The one-hot encoding is used for categorical features to transform non-numerical input for model training. To address class imbalance, we used stratified sampling in the training/testing split to ensure proportional class representation.

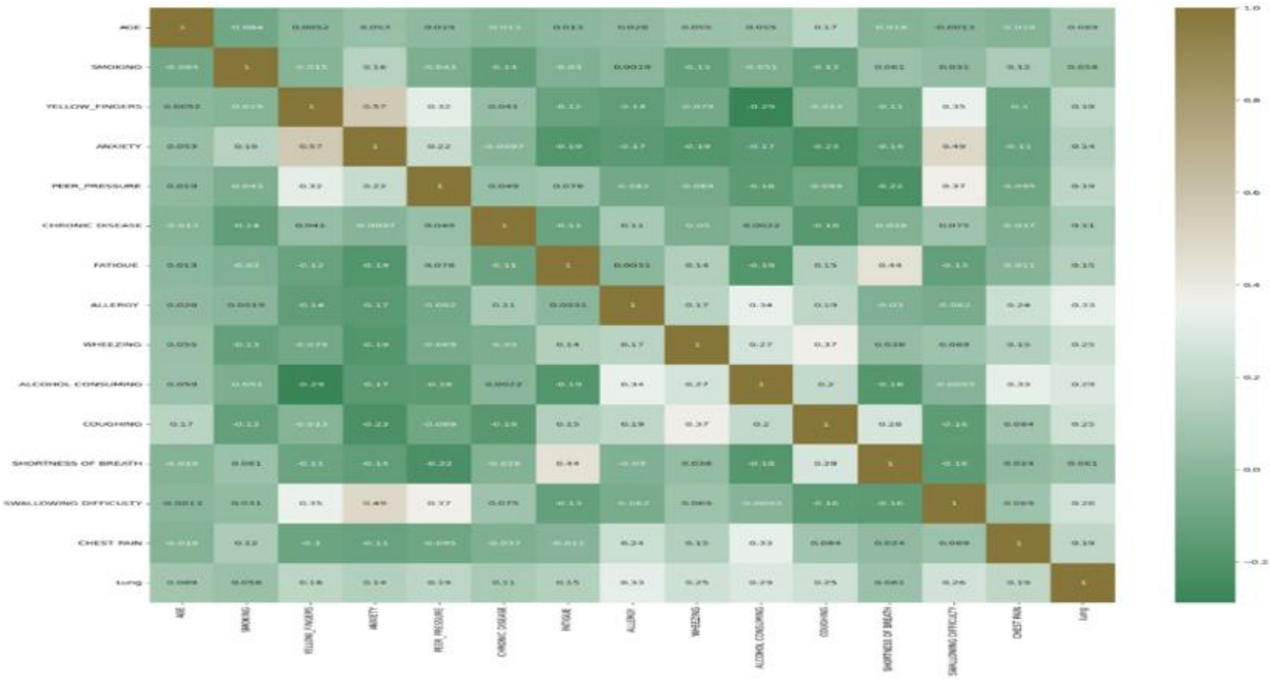

Figure 3: Scatter plot for each feature in the dataset.

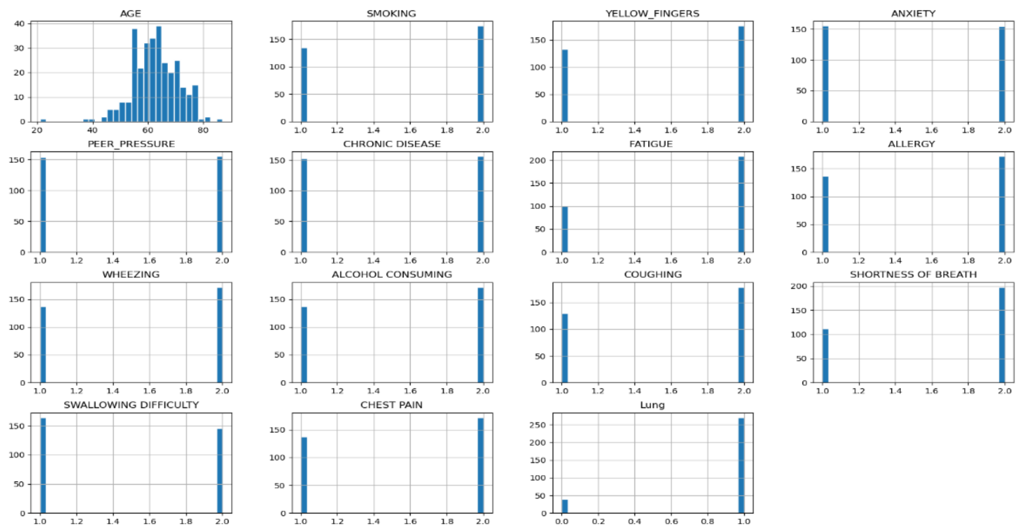

Figure 4: A correlation matrix between features in the dataset.

The input features include attributes of the dataset such as Gender, Smoking, Age, Yellow fingers, pressure, Anxiety, Chronic Disease, Swallowing Difficulty, Allergy, Fatigue, Wheezing, Shortness of Breath, Coughing, Alcohol, and Chest pain. The attributes count for sixteen, and the occurrences are 284. The above variables classify the output variable being Lung Cancer. Figure 3 shows the scatter plot below; it demonstrates the relation between the variables of the Lung Cancer input and output datasets. Figure 3 presents a correlation matrix heatmap for the characteristics of the dataset at hand.

Correlation Matrix is another important statistical tool applied in ascertaining the interrelationship between variables in a data set is the correlation matrix. The correlation matrix normally contains the pairwise correlation coefficients for each pair of variables moving from -1 to +1 and states the direction and strength of each interaction. From the correlation matrix, the picking of variables that are positively or negatively correlated will be based on which ones are best to investigate relationships, patterns, and potential predictors in the data.

This information becomes especially important to predictive modeling for handling multicollinearity issues, detecting dimensionality reduction, and selecting relevant characteristics. The correlation matrix of the used dataset is shown in Figure 4.

This study focuses on tabular data for lung cancer detection rather than imaging because deep learning for pictures requires significant resources, but machine learning techniques such as MLP effectively handle tabular data. Tabular data displays patient features and clinical history in structural manner, which improves decision-making clarity. Many tabular datasets are available and easily useable rather than imaging datasets that require extensive preparation to be usable.

Seven optimization algorithms, mentioned below, have applied the feature selection in binary form: Greylag Goose Optimization Gray Wolf Optimizer Mean-variance Optimization Whale Optimization Algorithm Sine Cosine Algorithm Particle Swarm Optimizer Falcon Optimization Algorithm the Below Table 2 represents the performance evaluation based on all these techniques of feature selection. From the table, it can be concluded that the results obtained in the proposed bGGO approach are better than those in other binary methods of feature selection.

Table 2: Evaluation of the suggested (bGGO)feature selection technique in comparison to other competitive techniques.

	bGGO	bSC	bMVO	bPSO	bWAO	bGWO	bFA
The Average error	0.774303	0.79150	0.80510	0.82530	0.82510	0.81160	0.8237
The Average Selected size	0.727103	0.92710	0.86950	0.92710	1.09050	0.84990	0.9616
The Average Fitness	0.837503	0.85370	0.86510	0.85210	0.85990	0.85980	0.9040
greatest Fitness	0.739303	0.77400	0.76840	0.83240	0.82400	0.83760	0.8227
lowest Fitness	0.837803	0.84090	0.88350	0.90010	0.90010	0.91380	0.9203
Standard deviation (SD)Fitness	0.659803	0.66450	0.66610	0.66390	0.66610	0.66510	0.7007

Figure 5 is showing the average error of the proposed technique against nine more of the feature selection strategies. From the figure, what can be seen is that the bGGO technique shows the lowest average error in graph form, underpinning its robustness.

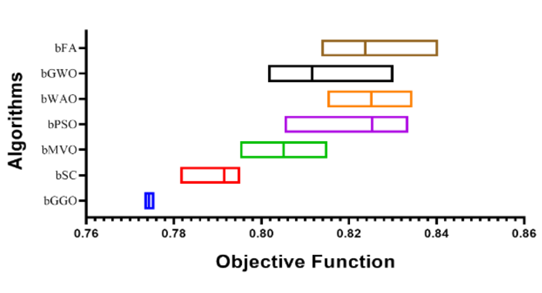

Figure 5: Results obtained with bGGO are of the lowest average error, underpinning the effectiveness and robustness of this feature selection technique.

To validate performance gains, we employed ANOVA and Wilcoxon signed-rank tests. For Hypothesis(H1) The ANOVA test yielded a p-value < 0.005, showing that GGO-MLP is statistically different from others. Wilcoxon Results compared GGO-MLP to baseline models. GGO-MLP demonstrated higher classification performance and structural improvements throughout numerous iterations, as validated by a p-value < 0.005.

Table 3 presents the performance of bGGO in combination with the existing feature selection strategies for different measures. It must be noticed that p-values were obtained using a comparison of results from each pair of algorithms, but the proposed feature selection technique shows statistically significant superiority.

It assumes that the mean values m of bGGO under H0 are equal to the mean values of bGWO, bPSO, bWAO, bSC, bMVO, and bFA. The alternative hypothesis, H1 considers the mean values of bGGO to be different from the rest of these hypotheses, a Wilcoxon rank-sum test was performed. The results of the Wilcoxon rank-sum test shown in Table 3 which indicates that the method proposed is statistically superior to techniques used previously and therefore is of less significance p < 0.005.

One-way ANOVA is conducted to test the hypothesis that there are statistically significant differences between the proposed bGGO technique and all other binary optimization techniques in use. The results obtained from the ANOVA test are presented in Table 4. These results ensure that the proposed feature selection technique is better, significant, and effective.

The study found a significant difference in accuracy between GGO-MLP and other models (p-value < 0.005). The Wilcoxon Signed-Rank Test indicated that GGO-MLP had much improved classification performance, with sustained benefits over numerous iterations. These studies quantitatively demonstrate the efficiency of GGO-MLP, proving that its optimizations outperform prior approaches.

Table 3: The proposed feature selection technique (bGGO) was evaluated using the Wilcoxon signed-rank test to compare it to other binary optimization techniques.

	bGGO	bSC	bMVO	bPSO	bWAO	bGWO	bFA
Actual median	0.7743	0.7915	0.8051	0.8253	0.8251	0.8116	0.8237
Number of values	10	10	10	10	10	10	10
Wilcoxon Signed Rank Test							
Signed ranks (W) summation	55	55	55	55	55	55	55
Positive ranks summation	55	55	55	55	55	55	55
Negative ranks summation	0	0	0	0	0	0	0
P (two-tailed) value	0.002	0.002	0.002	0.002	0.002	0.002	0.002
Exact or estimate	Exact	Exact	Exact	Exact	Exact	Exact	Exact
Significant with (alpha=0.05)?	Yes	Yes	Yes	Yes	Yes	Yes	Yes
How big is the Difference?							
Difference			0.7743	0.7915	0.8051	0.8253	0.8251

Table 4: The ANOVA test to evaluate the bGGO technique that has been suggested.

ANOVA table	SS	DF	MS	F (DFn, DFd)	P value
Treatment (between columns)	0.02274	6	0.00379	F (6, 63) = 135.3	P<0.0001
Residual (within columns)	0.001764	63	2.8E-05		
Total	0.02451	69			

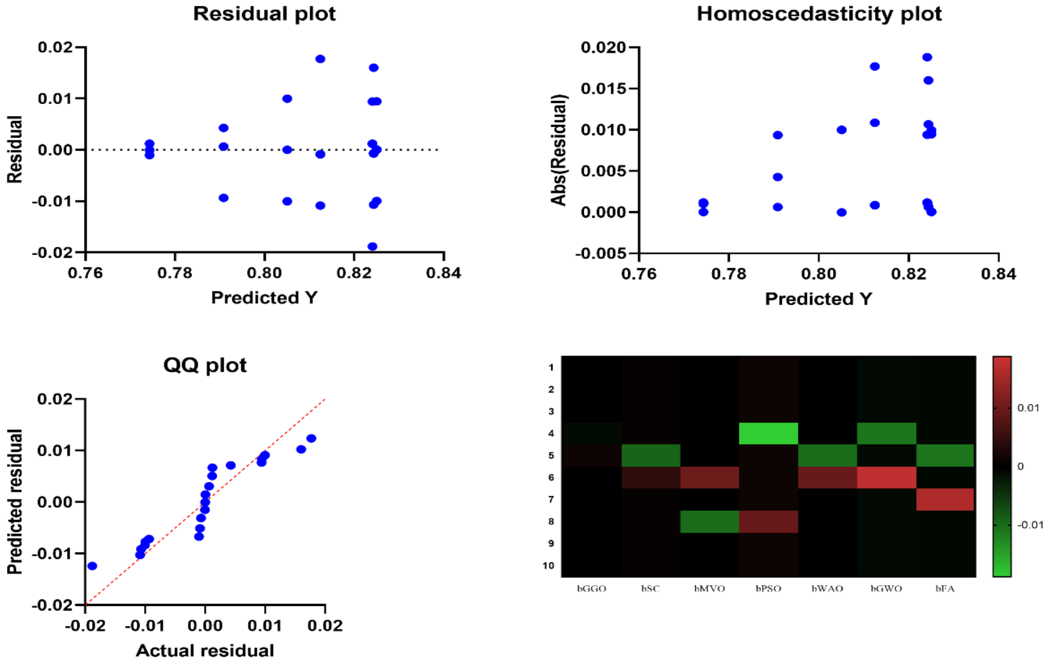

Figure 6: Analysis plots show the results derived from the suggested feature selection technique, bGGO

Figure 6 shows different plots of the results obtained using the proposed feature selection technique, including residual plots, quartile–quartile plots, homoscedasticity plots, and the heatmap.

1. Residual Plots: Residual plots of the distribution of residuals clearly describe how far observed values deviate from the predicted ones. These residual plots indicate that the proposed technique bGGO resulted in very close-to-zero residuals, hence highly predictable.

2. QQ Plots: The QQ plot generally depicts good linearity. This proves that the selected features are likely to be effective in the right classification of the presence of lung cancer. The linearity of plots on this QQ plot proves that residuals maintain a normal distribution and hence an indication of good feature effectiveness.

3. Homoscedasticity Plots: These plots consider the variability of the error terms concerning their constancy. In residual plots for the proposed bGGO method, there are quite random scatters of the residuals, directly showing that the variability is constant over other levels of the independent variable.

4. Heatmap: Colorfully explained, the approach proposed in bGGO is far more excellent than the other six strategies used in binary selection. It describes how the bGGO approach overpowers the previous ones by opting for relevant low-level features with the help of high-performance metrics.

The cumulative results from these plots ensure the effectiveness of the bGGO procedure coherently. The linearity of the QQ plot and the constant variance in the homoscedasticity plots reveal no or minimal residuals, thus proving the reliability and sufficiency of the bGGO procedure to describe lung cancer. Furthermore, it was quite clear from the heatmap that the bGGO method was outperforming the rest of the methods applied to feature selection, hence strengthening the area of its supremacy.

5.4 Classification results

Another experiment shows the sensitivity of the classification results concerning the applied feature selection technique. In the current research, machine learning classifiers were used for the classification of input data by characteristics chosen according to the bGGO technique for improving the characteristics of the network for further optimization in its performance.

Table 5 presents some of the results of classification with a machine-learning model after feature selection. These machine learning models include:

They are DTC, SVC, RFC, KNC, and MLP. Out of them, MLP has obtained the maximum value for accuracy, sensitivity, specificity, p-value, n-value, and F-score with 0.9180327, 0.92668, 0.9091, 0.9133, 0.9231, and 0.9199 respectively. This MLP model works as a fitness function, and it is optimized by the GGO algorithm and with six other optimization techniques.

The results for performance in classification while using an MLP model as the fitness function for various optimization algorithms are shown in Table 6. What is presented in the table is how GGO combined with MLP stands against combinations like SC, GWO, PSO, WAO, FA, and MVO with MLP to prove the supremacy of the proposed GGO+MLP approach. Offending was topped by the GGO plus MLP approach with an accuracy of 0.983837, sensitivity of 0.977337, specificity of 0.990237, p-value of 0.989957, n-value of 0.977961, and F-score of 0.983607.

In this attempt, various optimizers were used for tuning of parameters of MLP, and the results were studied and evaluated. Results proved that the efficiency and supremacy of GGO+MLP over other optimization techniques is unabated in all the metrics of evaluation, thus proving its robustness.

Table 5: Various classifiers for the categorization of lung cancer.

Models	Accuracy	Sensitivity (TRP)	Specificity (TNP)	P-value (PPV)	N-value (NPV)	F-Score
SVC	0.833895447	0.837171	0.83045	0.83855	0.829016	0.83786
Decision Tree	0.838087248	0.842715	0.833333	0.83855	0.837607	0.840628
Random Forest	0.863439931	0.877586	0.84922	0.854027	0.87344	0.865646
K Neighbors	0.8872	0.877586	0.895522	0.879102	0.894188	0.878343
MLP	0.918032787	0.926686	0.909091	0.913295	0.923077	0.919942

The Table only confirms the fact that the suggested strategy operates better than the current methods of optimization. It is for this reason that these results confirm the chief importance of the chosen technique of feature selection that significantly enhances the quality of the results. Furthermore, the proposed bGGO-based feature selection approach selects a small number of significant features yielding enhanced performance in comparison to other optimization algorithms-owned superiority taken by the results, thereby pointing to the robustness of the approach and efficient improvement on the accuracy, sensitivity, n-value, p-value, specificity, and F-score of the classifier; in other words, for the MLP model. It justifies the need and role of using feature selection techniques optimized to be able to achieve improved classification results over very complex datasets.

Table 6: Findings of optimization methods MLP model for the classifying lung cancer.

Models	Accuracy	Sensitivity (TRP)	Specificity (TNP)	P-value (PPV)	Nvalue (NPV)	F-Score
GGO+MLP	0.983837	0.977337	0.990237	0.989957	0.977961	0.983607
SC+MLP	0.966184	0.966387	0.965986	0.965035	0.967302	0.96571
GWO+MLP	0.960879	0.961003	0.960758	0.959666	0.96206	0.960334
PSO+MLP	0.951087	0.949106	0.95302	0.951724	0.950469	0.950413
WAO+MLP	0.94086	0.949106	0.932983	0.931174	0.950469	0.940054
FA+MLP	0.937287	0.943912	0.930707	0.931174	0.943526	0.9375
MVO+MLP	0.927978	0.933694	0.921986	0.926174	0.9299	0.929919

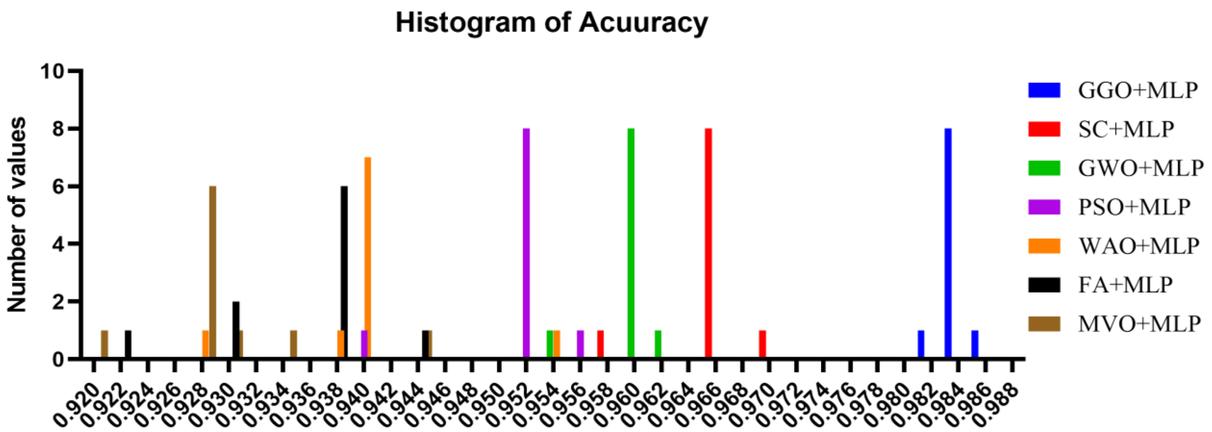

Figure 7: Assessing the accuracy of the GGO+MLP approach and optimization algorithms using the MLP model, considering the objective function.

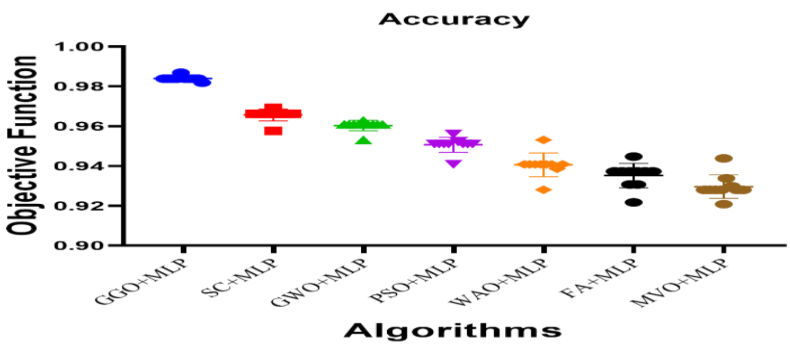

Figure 8.The accuracy results are achieved by the GGO+MLP approach as well as alternative combinations of optimization techniques with MLP models.

One can also show the effectiveness of this strategy for optimizing the objective function by comparing the suggested GGO+MLP method with other optimization algorithms in connection with the MLP model. Accuracy plots of the results obtained with the GGO+MLP approach are shown in Figures 6 and 7. Figure 7 shows the accuracy plot resulting from the MLP model, where the proposed method is outperforming other optimization algorithms. Another result, as shown in histogram form in Figure 8, has the same finding: better performance of the GGO+MLP approach. These plots have been used to show that the efficacy of the suggested feature selection technique, compared with other optimization techniques, is impressive in terms of achieving higher accuracy. The accuracy histogram plot further shows how well the suggested GGO+MLP method did in classifying the lung cancer cases from the input dataset.

Experiments applied ANOVA and Wilcoxon's rank-sum tests to determine how much the proposed algorithm differs statistically from the other competing algorithms.

Table 7 shows the results of ANOVA for the proposed GGO + MLP. Table 8 shows Wilcoxon's rank-sum test results to ascertain whether the algorithms' output differs in a statistically significant way. A p-value is less than 0.05 postulates that the performance difference is statistically significant. Results indicate that the GGO + MLP approach outperforms and demonstrates the statistical significance of the approach.

Table 7: The outcomes of the ANOVA of the proposed GGO algorithm with MLP model for lung cancer classification.

ANOVA table	SS	DF	MS	F (DFn, DFd)	P value
Treatment (between columns)	0.02191	6	0.003652	F (6, 63) = 184.11	P<0.000
Residual (within columns)	0.00125	63	1.98E-05		
Total	0.02316	69			

Table 8: Results of the Wilcoxon signed-rank test using the suggested method, (GGO+MLP), with various configurations of other optimization algorithms with the MLP model for lung cancer classification.

	GGO + MLP	SC+ MLP	GWO + MLP	PSO + MLP	WAO + MLP	FA+ MLP	MVO + MLP
Theoretical median	0	0	0	0	0	0	0
Actual median	0.9838	0.9662	0.9609	0.9511	0.9409	0.9373	0.928
Number of values	10	10	10	10	10	10	10
Wilcoxon Signed Rank Test							
Signed ranks (W) summation	55	55	55	55	55	55	55
Positive ranks summation	55	55	55	55	55	55	55
Negative ranks summation	0	0	0	0	0	0	0
P (two-tailed) value	0.002	0.002	0.002	0.002	0.002	0.002	0.002
Exact or estimate	Exact	Exact	Exact	Exact	Exact	Exact	Exact
P value summary	**	**	**	**	**	**	**
Significant (alpha=0.05)?	Yes	Yes	Yes	Yes	Yes	Yes	Yes
How big is the Difference?							
Difference	0.9838	0.9662	0.9609	0.9511	0.9409	0.9373	0.928

Table 7 presents the statistical results regarding the The Calibration Curve compares expected probability to actual positive outcomes, suggesting model dependability. A curve in Figure 10 is near to the diagonal indicates improved calibration, implying that GGO-MLP delivers reliable classification probabilities, which improves decision-making and generalization. The Partial Dependence Plot shows how factors influence predictions, specifically lung cancer likelihood, which helps model comprehension. Figure 10 illustrates GGO-MLP's capacity to identify critical features while avoiding overfitting, validating good feature selection and improving model clarity for more dependable predictions.

More confirmation of these results comes from residual plots and homoscedasticity plots of data. These are graphed in data as a heatmap in Figure 8, which shows the proposed approach to be far and far better than the other six binary feature selection methods.

Relative performance of the GGO+MLP method compared to the other five optimizers, SC, GWO, PSO, WAO, FA, and MVO, with the MLP model for all benchmark functions. According to Table 7, the GGO+MLP method was the best among the seven optimizers with the MLP model because it employed two different exploitation processes in each cycle.

The first converges very slowly toward the best solution found so far at any given time, while the second is more aggressive in searching for better solutions within the local neighborhood. These processes, when they tap into the power of the search space through the GGO+ MLP method, give excellent performance. A delicate

An optimum performance should, always, sustain a balance between exploration and exploitation of the domain of search. Moreover, it is important to start exploiting early in each iteration and smoothly increase the sum of the whole participant population within the exploitation group.

Overall, GGO was a better optimizer than all the other optimizers on most of the unimodal benchmark functions.

Figure 9 shows the residual plot, QQ plot, hetero-scedasticity plot, and heat map of this case. These plots show how efficient and strong the proposed GGO + MLP approach will be. The QQ plot has values following a linear trend, thus showing how efficient the features selected were in categorizing the cases.

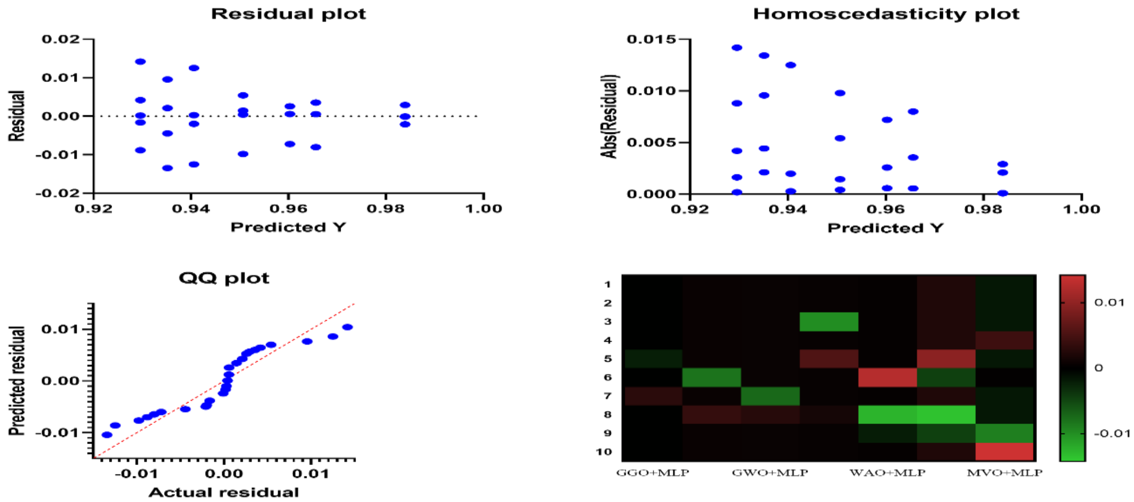

Figure 9: Analysis plots of the obtained results using the proposed GGO+MLP approach

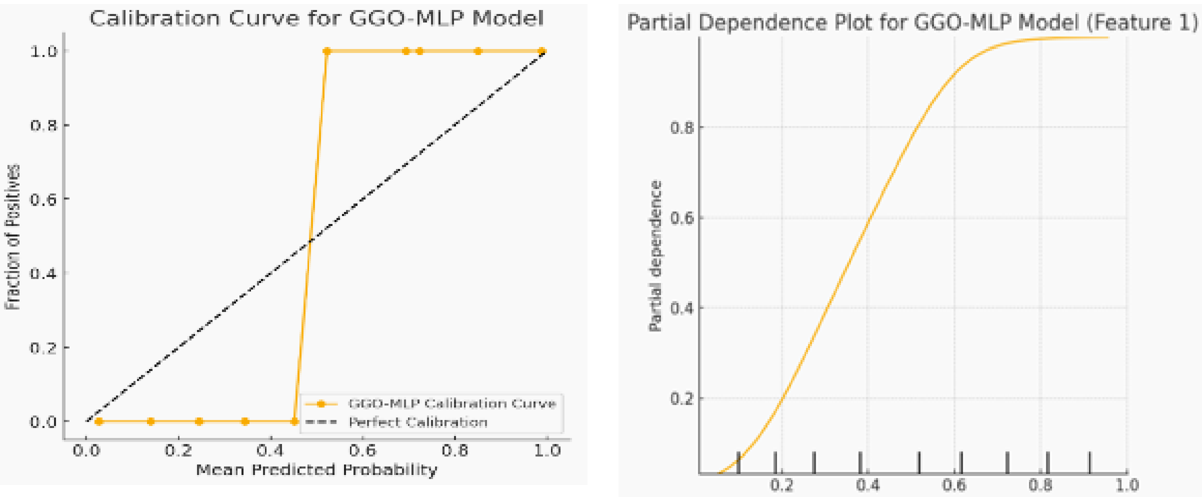

Figure 10: Analysis of GGO-MLP performance using partial dependence plot and calibration curve.

The Calibration Curve compares expected probability to actual positive outcomes, suggesting model dependability. A curve in Figure 10 is near to the diagonal indicates improved calibration, implying that GGO-MLP delivers reliable classification probabilities, which improves decision-making and generalization. The Partial Dependence Plot shows how factors influence predictions, specifically lung cancer likelihood, which helps model comprehension. Figure 10 illustrates GGO-MLP's capacity to identify critical features while avoiding overfitting, validating good feature selection and improving model clarity for more dependable predictions.

More confirmation of these results comes from residual plots and homoscedasticity plots of data. These are graphed in data as a heatmap in Figure 8, which shows the proposed approach to be far and far better than the other six binary feature selection methods.

6 Discussion

The bar graph easily indicates that the GGO+MLP method works, giving the highest outcome among other methods. Figure 8 shows that this hybrid method for lung cancer patient classification has been highly useful in solving the optimization problems discussed.

The suggested GGO-MLP strategy was tested against cutting-edge (SOTA) approaches such as CNN-based classifiers and other metaheuristic techniques. The results show that GGO-MLP regularly outperforms other methods in terms of accuracy, sensitivity, and specificity. The primary causes for this improvement are:

Feature Dimensionality Reduction with bGGO: Unlike CNN models, which use complicated feature representations, bGGO picks the most relevant features, lowering computational cost while preserving classification performance.

Efficiency of Hyperparameter Tuning: Local optima and inefficient hyperparameter settings provide challenges for traditional optimization strategies. GGO's adaptive exploration-exploitation balance improves parameter selection, resulting in higher classification outcomes.

The combination of GGO (Group-based Genetic Optimization) and MLP (Multi-Layer Perceptron) performs better than the state-of-the-art (SOTA) methods because bGGO (a specific version of GGO) identifies the best set of features, which helps avoid overfitting. In this case, bGGO selected 12 out of 16 available features, while bPSO (a Particle Swarm Optimization algorithm) kept 15 features, which can add unnecessary noise to the model. This selective feature reduction leads to better performance. GGO-MLP outperforms other approaches ($p < 0.005$), indicating dependability and efficacy.

7 Limitations

GGO algorithm study in improving lung cancer classification shows good results, especially in improving feature selection and classification accuracy. However, there are several limitations and directions for future work that can be addressed to further improve the effectiveness of GGO in this field:

- The performance of the GGO algorithm was tested on one dataset only that may not represent the variety of real-world scenarios. Results may vary across different datasets, which may affect the generalizability of the results.
- Classification accuracy depends largely on the quality and completeness of the input data. Any inconsistencies or missing values in the data can affect the results.

8 Conclusion and future works

The paper endeavors to improve the accuracy in the classification of cases for the diagnosis of lung cancer. Three techniques for the initial data preparation are applied: scaling, normalization, and cessations of nulls. The binary variant of GGO is used to perform feature selection in this category of techniques, and it is centrally referred to as bGGO. This binary format is developed for GGO to optimize the best combination of features that enhances the classification accuracy with seven other binary optimization algorithms, including SC, MOO, PSO, WOA, GWO, and FOA.

In the classification phase, numerous classifiers are applied, namely SVC, DTC, RFC, KNC, and MLP. Among these multiple classifiers, the MLP classifier gives the best result with an accuracy rate of 91.8%. The hyperparameters of the MLP model have been optimized using the GGO, and the result is compared with those of six other optimizers. The MLP model along with the GGO has given the best result with an accuracy rate of 98.4%.

The results of feature selection followed by classification were statistically evaluated using Wilcoxon's signed-rank test as well as ANOVA, whereas result visualization plots were also done to ascertain the strength and efficiency of the proposed scheme. The results of the experiments and the statistical analysis prove that the proposed approach outperformed other methods compared to the classification of lung cancer. This improved the overall accuracy of prediction with cancer feature selection and reduced dimensionality. The problem of overfitting in the analysis of cancer features was overcome. A pathway toward enhanced early prediction rates for lung cancer is through sensor data acquisition, analysis, and finding the best approach in the future.

The performance of the GGO algorithm was tested on specific dataset for lung cancer classification, so in the future, several large datasets can be used for generalizing the results. Also, in the future several optimization models and deep learning models 32–34 can be used for lung cancer classification. Also, future study may combine tabular and imaging data to enhance diagnosis.

Acknowledgment: Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2023R308), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

References

[1] Asuntha A, Srinivasan A. Deep learning for lung cancer detection and classification. Multimedia Tools and Applications. 2020; 79:7731–62. DOI: 10.1007/s11042-019-08394-3

[2] Chaturvedi P, Jhamb A, Vanani M, Nemade V. Prediction and classification of lung cancer using machine learning techniques. In: IOP Conference Series: Materials Science and Engineering. 2021; 1099:012059. DOI: 10.1088/1757-899X/1099/1/012059

[3] Tiwari V, Jampala N, Tentu AN, Saxena A. Towards finding active number of S-boxes in block ciphers using mixed integer linear programming. Informatica. 2021;45(6):77–87. DOI: 10.31449/inf. v45i6.3427 Informatica+1ResearchGate+1

[4] Nanglia P, Kumar S, Mahajan AN, Singh P, Rathee D. A hybrid algorithm for lung cancer classification using SVM and neural networks. ICT Express. 2021;7(3):335–41. DOI: 10.1016/j.icte.2020.06.007

[5] Alamgeer M, Mengash HA, Marzouk R, Nour MK, Hilal AM, Motwakel A, Zamani AS, Rizwanullah M. Deep learning enabled computer aided diagnosis model for lung cancer using biomedical CT images. Computers, Materials & Continua. 2022;73(1):1–15. DOI: 10.32604/cmc.2022.021569

[6] Jasim HM, Ghrabat MJ, Abdulrahman LQ, Nyangaresi VO, Ma J, Abduljabbar ZA, Abduljaleel IQ. Provably efficient multi-cancer image segmentation based on multi-class fuzzy entropy. Informatica. 2023;47(1):77–88. DOI: 10.31449/inf.v47i1.3742 ResearchGate

[7] Priyadharshini P, Zoraida BS. Bat-inspired metaheuristic convolutional neural network algorithms for CAD-based lung cancer prediction. Journal of Applied Science and Engineering. 2021;24(1):65–71. DOI: 10.6180/jase.202102_24(1).0008

[8] Ma Q, Jimenez G. RETRACTED: Lung cancer diagnosis of CT images using metaheuristics and deep learning. Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine. 2022;236(4):NP17–NP29. DOI: 10.1177/09544119221090725 PubMed

[9] Mohamed TIA, Oyelade ON, Ezugwu AE. Automatic detection and classification of lung cancer CT scans based on deep learning and Ebola optimization search algorithm. PLOS ONE. 2023;18(8): e0285796. DOI: 10.1371/journal.pone.0285796

[10] Ren Z, Zhang Y, Wang S. A hybrid framework for lung cancer classification. Electronics. 2022;11(10):1614. DOI: 10.3390/electronics11101614

[11] Bhattacharjee A, Murugan R, Goel T. A hybrid approach for lung cancer diagnosis using optimized random forest classification and K-means visualization algorithm. Health and Technology. 2022;12(4):787–800. DOI: 10.1007/s12553-022-00679-2

[12] Shakeel PM, Tolba A, Al-Makhadmeh Z, Jaber MM. Automatic detection of lung cancer from biomedical data set using discrete AdaBoost optimized ensemble learning generalized neural networks. Neural Computing and Applications. 2020;32(3):777–790. DOI: 10.1007/s00521-018-03972-2

[13] Vijh S, Gaur D, Kumar S. An intelligent lung tumor diagnosis system using whale optimization algorithm and support vector machine. International Journal of System Assurance Engineering and Management. 2020;11(2):374–384. DOI: 10.1007/s13198-019-00866-x

[14] Nancy DP, Kishan SR, Rane KP, Kaliyaperumal DK, Meenakshi D, Suartama IK. Optimized feature selection and image processing-based machine learning technique for lung cancer detection. International Journal of Electrical and Electronics Research. 2022;10(4):888–894. DOI: 10.5281/zenodo.366698169

[15] El-knew ESM, Khodadadi N, Mirjalili S, Abdelhamid AA, Eid MM, Ibrahim A. Greylag Goose Optimization: Nature-inspired optimization algorithm. Expert Systems with Applications. 2024; 238:122147. DOI: 10.1016/j.eswa.2023.122147

[16] Samadianfard S, Hashemi S, Kargar K, Izadyar M, Mostafaeipour A, Mosavi A, et al. Wind speed prediction using a hybrid model of the multi-layer perceptron and whale optimization algorithm. Energy Reports. 2020; 6:1147–1159. DOI: 10.1016/j.egyr.2020.05.003

[17] Mirjalili S. SCA: A sine cosine algorithm for solving optimization problems. Knowledge-Based Systems. 2016; 96:120–133. DOI: 10.1016/j.knosys.2015.12.022

[18] Rigamonti A. Mean-variance optimization is a good choice, but for other reasons than you might think. Risks. 2020;8(1):29. DOI: 10.3390/risks8010029

[19] Piotrowski AP, Napiorkowski JJ, Piotrowska AE. Particle swarm optimization or differential evolution—A comparison. Engineering Applications of Artificial Intelligence. 2023; 121:106008. DOI: 10.1016/j.engappai.2023.106008

[20] Mirjalili S, Lewis A. The whale optimization algorithm. Advances in Engineering Software. 2016; 95:51–67. DOI: 10.1016/j.advengsoft.2016.01.008

[21] Al-Tashi Q, Md Rais H, Abdulkadir SJ, Mirjalili S, Alhussian H. A review of grey wolf optimizer-based feature selection methods for classification. In: Evolutionary Machine Learning Techniques: Algorithms and Applications. Springer Singapore; 2020. p. 273–286. DOI: 10.1007/978-981-32-9990-0_13

[22] De Vasconcelos Segundo EH, Mariani VC, dos Santos Coelho L. Design of heat exchangers using a falcon optimization algorithm. Applied Thermal Engineering. 2019; 156:119–144. DOI: 10.1016/j.applthermaleng.2019.04.038