Received: July 12, 2025. Revised: October 22, 2025.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

XCOA-MLP: Extended Coyote Optimization Algorithm for Training Neural Networks in Medical Data Classification

Maher Talal Al-Asaady[1,2] **Teh Noranis Mohd Aris[2]*** **Nurfadhlina Mohd Sharef[1]**
Hazlina Hamdan[1]

[1]*Department of Computer Science, Faculty of Computer Sciences and Information Technology, Universiti Putra Malaysia, Malaysia*
[2]*Department of Network and Computer Software Techniques, Northern Technical University, Mosul 41001, Iraq*
* Corresponding author's Email: nuranis@upm.edu.my

Abstract: Artificial neural networks (ANNs) are widely applied in medical data classification due to their ability to model complex and nonlinear patterns. However, their performance is highly contingent upon the effectiveness of their training methods. Traditional training methods such as backpropagation often suffer from slow convergence and local minima. Meta-heuristic algorithms provide global search capability but may face efficiency limitations. This study proposes XCOA-MLP, a multilayer perceptron trained using an improved coyote optimization algorithm (XCOA). XCOA extends the standard COA by introducing a leader pack (LP) technique that accelerates convergence and enhances exploitation through inter-pack knowledge sharing. The model was evaluated on five UCI medical datasets and benchmarked against eight meta-heuristic algorithms. Results show that XCOA-MLP achieves superior accuracy and robustness, recording 97.8% on Breast cancer, 78.2% on Diabetes, and 87.7% on Parkinsons datasets. These findings demonstrate XCOA-MLP's effectiveness in improving neural network training for medical data classification.

Keywords: Coyote optimization algorithm (COA), Metaheuristics, Medical data classification, Multilayer perceptron (MLP), Neural network training, Optimization, Weights and biases.

1. Introduction

Medical data classification plays a vital role in supporting disease diagnosis and treatment planning, where accurate predictive models can improve early detection, guide therapies, and enhance patient outcomes [1]. Artificial neural networks (ANNs), particularly multilayer perceptrons (MLPs), are widely adopted for this task due to their ability to model nonlinear and complex relationships. However, their performance strongly depends on the optimization of weights and biases during training [2].

Traditional training approaches, such as backpropagation (BP), remain popular because of their deterministic formulation and ease of implementation. Nevertheless, they often suffer from slow convergence, sensitivity to initialization, and entrapment in local minima [3, 4], which are further exacerbated when dealing with noisy or high-dimensional medical datasets [5].

In order to overcome these difficulties, meta-heuristic algorithms (MHAs) have become more and more popular among researchers. These are the population-based optimization techniques that are informed by natural, social, or physical phenomena that provide the ability to search globally and are robust without gradient information [6]. Techniques such as genetic algorithms (GA) [2], mountain gazelle optimizer (MGO) [7], and whale optimization algorithm (WOA) [8] have been applied with varying success. Yet, many of these methods still encounter issues such as premature convergence and computational inefficiency in large-scale medical applications [4, 9].

Coyote optimization algorithm (COA) is a population-based meta-heuristic proposed by Pierezan and Coelho [10], which mimics the social

Nomenclature	
cult	Median behavior of coyotes in a pack
D	Dimension of the search space
f(X)	Fitness value of solution X
lb_j, ub_j	Lower and upper bounds of the j^{th} dimension
N	Population size (total coyotes)
n, m, r	#input, hidden, and output neurons
N_c	Number of coyotes per pack
N_p	Number of packs
O	The rank-based ordered social conditions
P_e	Coyotes migration probability
P_s, P_a	Scattering and association probabilities
pup	A new coyote's social condition
r_j	Random number in [0,1] for dimension *j*
R_j	Random factor for generating new pup
r_{LP}	Weight (strength of the LP influence)
rnd	A uniform random number
soc	Social condition of a coyote
T	Maximum iterations or stopping criterion
T_c	Cultural tendency of a pack
T_{LP}	Cultural tendency of Leader Pack
W, θ	Weights and biases of the neural network
X_i	Social condition of the i^{th} coyote
y_k	Actual target output for sample k
y'_k	Predicted output for sample k
α	Alpha coyote (best solution in a pack)
δ_1, δ_2	Influence of alpha and pack tendency
δ_{LP}	LP influence on the regular packs

organization and interactions of coyotes in packs, combining evolutionary and swarm intelligence strategies. COA includes birth, death, and migration operations to explore the solution space.

This study introduces XCOA-MLP, a novel model that employs a variant of COA for training MLPs in medical data classification tasks. The proposed algorithm integrates a proposed leader pack (LP) technique into the COA algorithm to enhance convergence and improve classification accuracy. The effectiveness of the model is validated using standard benchmark functions [11] and five UCI medical datasets [12], and its performance is compared with eight meta-heuristic training methods.

The remainder of this paper is as follows: Section 2 reviews related work; Section 3 details the methodology (COA, XCOA, proposed XCOA-MLP model, and experimental configuration); Section 4 presents and discusses the results; and Section 5 concludes the paper.

2. Related work

Traditional training methods, particularly BP, are widely used to train MLP neural networks due to their simplicity; however, they suffer from several issues, such as slow convergence, sensitivity to initial parameters, and a tendency to be trapped in local minima [3, 13], especially when applied to noisy or complex datasets [5]. Various MHAs have been explored to overcome these limitations [9]. These are population based optimization algorithms that are based on natural, social, or physical phenomenon and have the ability to search globally and perform well even in the absence of gradient data.

Evolutionary approaches such as cellular GA [2], greedy GA [14], and differential evolution variants [15] achieved promising results, though often limited by scalability or parameter sensitivity. Swarm-based methods, including MGO [7], FOX [16], AMO [17], IMP-GWO [18], FMFO [19], and WOA [8], improved classification accuracy and search efficiency, yet some struggled with premature convergence or scalability. Physics-inspired algorithms like GSA [20], SCA [21], and MVO [22] offered reliable convergence but less adaptability compared to swarm-based methods. Table 1 shows a list of recent studies that utilized MHAs to train ANNs with metrics, key findings, and interpretations.

MHAs have been effective in the training of ANNs but current procedures still face major issues such as premature convergence, failure to seek out plausible solutions, and computation impracticability under complex problem spaces [4].

COA algorithm has been demonstrated to possess the ability to address these constraints by engaging evolutionary and swarm intelligence process [10]. Nevertheless, problems of local optima convergence, along with delays in search over high-dimensional spaces, remain [23]. Several enhancements, including the Sobol series [24], ICOA [25], MvCOA [26], and chaotic tent maps [27], have been proposed to improve diversity and convergence speed. These advances highlight COA's adaptability and motivate its further development for robust neural network optimization.

The proposed XCOA-MLP modified COA by introducing the LP technique, which facilitates inter-pack knowledge transfer. This innovation enhances convergence speed, improves exploitation while maintaining diversity, and addresses the limitations of conventional BP and previously applied MHAs.

3. Methodology

3.1 Coyote optimization algorithm (COA)

COA is an MHA that utilizes the social behavior of coyotes and social dynamics of packs to find optimal solutions [10]. The algorithm divides its

Table 1. Overview of studies utilizing meta-heuristic algorithms for training neural networks

Ref.	Year	MHA	Metrics	Key Findings	Interpretation
[2]	2022	CGA	Acc, Sn, Sp	CGA-DX exhibited higher accuracy and lower MSE than other MHAs.	Practical crossover and mutation mechanism.
[14]	2022	GGA	Acc, Sn, Sp	The GGA outperformed the traditional MLP on several medical datasets.	Enhanced population selection ensures better convergence.
[28]	2025	DE	Acc	The DDE-OP achieved second-best classification accuracy in all datasets.	Uses clustering and quasi-opposition strategy.
[7]	2025	MGO	Acc	Superior accuracy and convergence on ten UCI datasets.	Integrates four key behavioral strategies.
[16]	2024	FOX	Acc, Sn, Sp, RMSE	Improved training, avoided local optima.	Probabilistic jumps mimic red fox hunting.
[17]	2022	AMO	Acc, Sn, Sp	Enhanced search, avoided local optima.	The adaptive movement mimics ant behavior, ensuring robust optimization.
[18]	2023	IGWO	Acc, Precision, F1	Outperformed the original GWO in performance and convergence.	Improved accuracy; higher computational cost.
[19]	2023	FMFO	MSE, Acc, Speed, Time	Enhanced search speed and accuracy.	Combines flame-based search with adaptive behavior.
[8]	2022	WOA	Acc, AUC, Sp, Sn	Effective in avoiding local minima, particularly effective for medical datasets.	Modified WOA improves robustness for medical data.
[20]	2023	GSA	MSE	Strong MLP performance via chaos & Lévy flight.	Chaos theory enhances search, improving performance in complex spaces.
[29]	2020	SCA	Acc, MSE	High performance in complex datasets.	Sine-cosine mechanics navigate non-linear datasets effectively.
[22]	2016	MVO	Acc, MSE	Improved convergence and local optima handling.	Multiverse-inspired exploration excels at avoiding local optima.

population into packs of coyotes, each one represents a candidate solution in the D solution space, with a social condition defined by Eq. (1).

$$soc_c^{p,t} = \vec{x} = (x_1, x_2, \dots, x_D) \quad (1)$$

The adaptation of coyote to the environment is measured by the fitness, denoted by $fit_c^{p,t} \in R$.

1) Coyote initialization: Initially, each coyote's social condition is randomly determined by Eq. (2).

$$soc_{c,j}^{p,t} = lb_j + r_j . (ub_j - lb_j) \quad (2)$$

where ub_j and lb_j are the lower and upper bounds of the dimension j, respectively, r_j is a random number in [0, 1], and the fitness of each coyote is then evaluated using Eq. (3).

$$fit_c^{p,t} = f\left(soc_c^{p,t}\right) \quad (3)$$

2) Alpha selection: Each pack has a single 'alpha' coyote representing the best solution within the pack. The alpha of the p^{th} pack at the t^{th} iteration is defined by Eq. (4).

$$\alpha^{p,t} = \{soc_c^{p,t} | arg_{c=\{1,2,\dots,N_c\}} + minf(soc_c^{p,t})\} \quad (4)$$

3) Cultural tendency: Because coyotes demonstrate group intelligence, they share cultural knowledge, influencing their collective behavior. The cultural tendency is determined by Eq. (5).

$$cult_j^{p,t} = \begin{cases} O^{p,t}_{\frac{(N_c+1)}{2},j}, & N_c \text{ is odd} \\ \frac{O^{p,t}_{\frac{N_c}{2},j} + O^{p,t}_{(\frac{N_c}{2}+1),j}}{2}, & otherwise \end{cases} \quad (5)$$

where $O^{p,t}$ denotes the rank-based ordered social conditions of coyotes.

4) Coyote birth and death: Coyotes' ages, defined as $age^{p,t}{}_c$ influence birth and death events. A new coyote's (pup) is formulated as Eq. (6).

$$pup_j^{p,t} = \begin{cases} soc_{r1,j}^{p,t}, & if\ rnd_j < P_s\ or\ j = j_1 \\ soc_{r2,j}^{p,t}, & if\ rnd_j \geq P_s + P_a\ or\ j = j_2 \\ R_j, & otherwise \end{cases} \quad (6)$$

where j_1 and j_2 are random dimensions, r^1 and r^2 are parent coyotes, R_j is a random number, and rnd_j is a uniform random number. The scattering and association probabilities, P_s and P_a, are given as:

$$P_s = 1/D \quad (7)$$

$$P_a = (1 - P_s)/2 \quad (8)$$

5) Updating social conditions: Social conditions update based on alpha influence $\delta 1$ and pack influence $\delta 2$, computed as:

$$\delta_1 = \alpha^{p,t} - soc_{cr1}^{p,t} \quad (9)$$

$$\delta_2 = cult^{p,t} - soc_{cr2}^{p,t} \quad (10)$$

Thus, the coyote's new social conditions are updated based on the influence of the alpha and the pack.

$$new\_soc_c^{p,t} = soc_c^{p,t} + r_1 . \delta_1 + r_2 . \delta_2 \quad (11)$$

where r_1 and r_2 represent the weights of the alpha and pack effects, respectively, randomly selected from [0, 1]. An assessment of the new social condition is presented in Eq. (12).

$$new\_fit_c^{p,t} = f(new\_soc_c^{p,t}) \quad (12)$$

If the new social condition yields a better fitness value, it is adopted as in Eq. (13).

$$soc_c^{p,t+1} = \begin{cases} new\_soc_c^{p,t}, & if\ new\_fit_c^{p,t} < fit_c^{p,t} \\ soc_c^{p,t}, & otherwise \end{cases} \quad (13)$$

6) Coyote migration: Coyotes may migrate between packs, influenced by the probability of pack size:

$$\mathrm{P_e} = 0.005 . \mathrm{N_c^2} \quad (14)$$

Finally, the algorithm selects the optimal solution as the best-adapted coyote.

3.2 Extended COA (XCOA)

The solution updates in the original COA were limited to packs, with only two random chosen coyotes having an effect on each other. This limited communication tended to reduce the convergence speed and lead to stasis upon getting stuck in local optima by packs. The extended COA (XCOA) resolves these limitations by providing a leader pack (LP) technique, whereby coyotes in different packs will be brought together to exchange knowledge, improve exploitation, and speed up convergence. The mechanism of selection, cultural tendency and influence of the LP is formally stated in the subsequent subsections.

3.2.1. Leader pack (LP)

The LP is a special pack in COA, containing the top-performing coyotes from all packs, selected based on their fitness values. This elite subset introduces hierarchical control over all packs, facilitating the exchange of superior solutions and helping escape local optima. This ensures that strong solutions propagate across packs, guiding the population toward the global optimum. The LP updates periodically at fixed intervals, such as every T_{LP} iteration. At each T_{LP} interval, LP is formed by selecting the top N_{LP} coyotes from all packs based on the fitness values:

$$LP = argmin_{N_{LP}} \{f(soc_c^p) | p = 1,2,\ldots, N_p; c = 1,2,\ldots,N_c\} \quad (15)$$

where f represents the selection of social conditions associated with the minimum (best) fitness values across all packs, N_{LP} is the number of coyotes in LP.

3.2.2. LP cultural tendency

Once the LP is formed, it calculates its cultural tendency, representing the dominant social behavior of the best-performing coyotes. The cultural tendency of the LP, *cult*, is defined as the median of social conditions in LP, as in Eq. (16).

$$cult_j^{LP,t} = \begin{cases} O_{\frac{(N_{LP}+1)}{2},j}^{LP,t}, & if\ N_{LP}\ is\ odd \\ \frac{O_{\frac{N_{LP}}{2},j}^{LP,t} + O_{\left(\frac{N_{LP}}{2}+1\right),j}^{LP,t}}{2}, & otherwise \end{cases} \quad (16)$$

here, the [1, D] interval, $O^{LP,t}$ stands for the rank-based social condition of all coyotes in the LP at T_{LP} iteration. The LP's cultural tendency reflects high-performance solutions that influence the population.

Table 2. Benchmark functions used for XCOA validation

Function	Dim	Initial Range	F_{min}
F1 (Sphere)	30	[-100, 100]	0
F2 (Schwefel 2.22)	30	[-10, 10]	0
F3 (Schwefel 1.2)	30	[-100, 100]	0
F4 (Schwefel 2.21)	30	[-100, 100]	0
F5 (Rosenbrock)	30	[-30, 30]	0
F6 (Step)	30	[-100, 100]	0
F7 (Quartic + noise)	30	[-1.28, 1.28]	0
F8 (Schwefel)	30	[-500, 500]	-418.98
F9 (Rastrigin)	30	[-5.12, 5.12]	0
F10 (Ackley)	30	[-32, 32]	0
F11 (Griewank)	30	[-600, 600]	0
F12 (Penalized #1)	30	[-50, 50]	0
F13 (Six-hump Camel)	2	[-5, 5]	-1.0316
F14 (Shekel's Foxholes)	2	[0, 14]	0

3.2.3. LP influence mechanism

At each T_{LP} interval iteration, the LP temporarily merges with each pack to inject its cultural trend and refine solutions. This ensures that regular packs inherit successful strategies, thereby improving search efficiency. The updated social conditions were computed using Eqs. (17) and (18).

$$\delta_{LP} = cult^{LP,t} - soc_c^{p,t} \quad (17)$$

$$new\_soc_c^{p,t} = soc_c^{p,t} + r_{LP} \cdot \delta_{LP} \quad (18)$$

where δ_{LP} is the LP influence on the regular packs, r_{LP} is an added random weight factor determines the strength of the LP influence. The influence occurs at predefined intervals.

3.2.4. Control parameters

Key control parameters in XCOA include:

- *LP size:* determines the number of elite coyotes, influencing the balance between guidance strength and population diversity.
- *Influence frequency:* Controls how often the LP interacts with regular packs.

Fig. 1 represents the workflow of XCOA. The left side depicts the normal COA process whereby there is pack initiation, alpha selection, cultural update, migration, and birth and death cycles. The enhancement of the XCOA is emphasized in the right side whereby an LP of the best coyotes of all packs facilitates the transfer of knowledge and speedy convergence.

3.2.5. Validation of XCOA

The proposed XCOA was validated using fourteen benchmark functions. These functions span an extensive range of optimization problems multimodal, non-separable, separable, and unimodal [11].

Table 2 summarizes the configuration of the benchmark functions used in this study (F1–F14). Each function is defined by its dimensionality, initial search domain, and known global optimum value ($Fmin$).

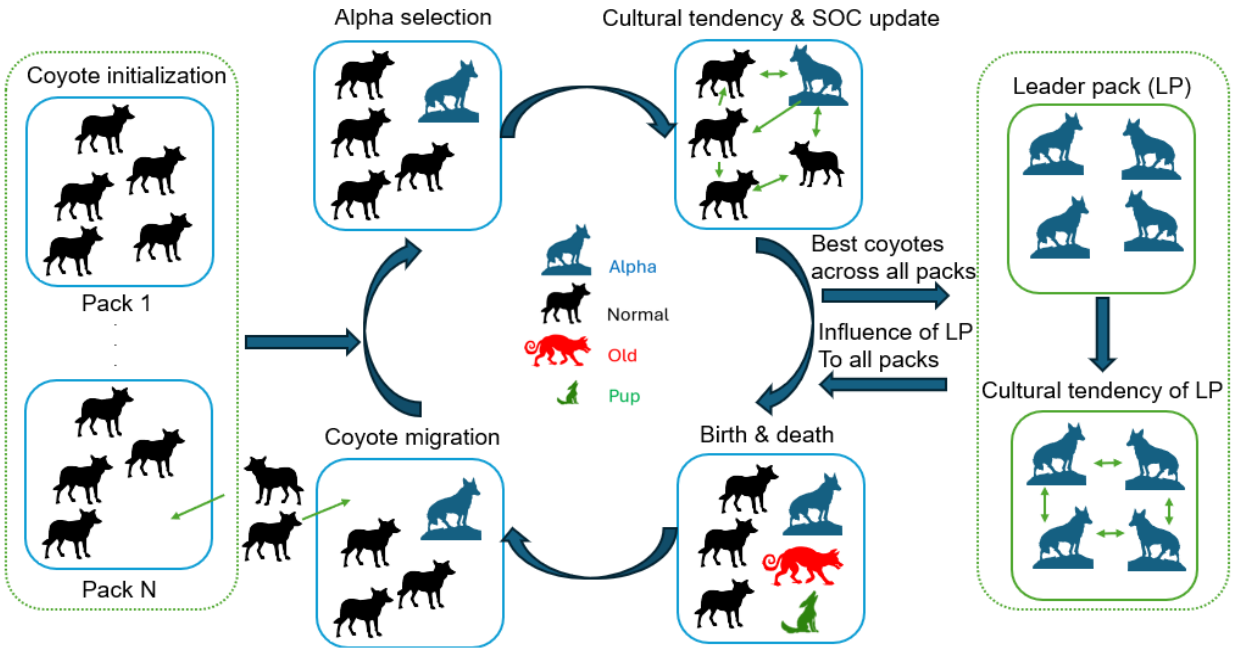

Figure. 1 XCOA workflow with LP technique

Table 3. Performance comparison of the original COA and proposed XCOA with multiple variations

Fn	Metrics	COA	XCOA1	XCOA2	XCOA3	XCOA4	XCOA5	XCOA6
F1	Mean	1.53e-07	6.35e-20	4.37e-17	3.61e-20	1.00e-13	2.00e-15	1.22e-14
	Best	2.00e-08	2.00e-22	4.25e-19	2.35e-23	4.90e-15	2.46e-17	1.27e-16
	Worst	5.24e-07	4.52e-19	2.56e-16	6.74e-19	3.64e-13	2.31e-14	1.46e-13
	STD	1.05e-07	8.40e-20	5.54e-17	1.34e-19	9.40e-14	4.45e-15	2.69e-14
F2	Mean	1.28e-06	0.00027	1.61e-12	7.97e-07	9.71e-12	1.10e-12	2.89e-12
	Best	4.79e-07	1.51e-12	2.77e-14	1.53e-12	1.71e-12	1.73e-13	3.84e-13
	Worst	3.63e-06	0.00815	2.23e-11	2.29e-05	2.53e-11	5.94e-12	9.06e-12
	STD	7.51e-07	0.00146	3.97e-12	4.10e-06	5.85e-12	1.13e-12	1.88e-12
F3	Mean	36.19	0.0287	0.0112	0.0144	0.331	0.0230	0.0759
	Best	8.17	0.00089	0.00058	0.00015	0.0210	0.00148	0.00748
	Worst	98.70	0.275	0.0428	0.101	1.90	0.107	0.302
	STD	19.24	0.0570	0.0103	0.0269	0.463	0.0242	0.0665
F4	Mean	5.58	7.50	7.51	7.20	8.15	7.56	11.32
	Best	3.85	4.56	4.93	3.53	5.20	5.13	5.96
	Worst	8.36	10.62	10.34	11.59	10.20	10.65	15.81
	STD	0.949	1.47	1.11	1.87	1.27	1.36	2.20
F5	Mean	20.62	33.34	18.10	7.00	17.85	20.08	9.01
	Best	1.27	0.00246	0.00334	0.00086	0.0194	0.0948	0.00754
	Worst	105.34	237.31	91.73	159.78	223.60	68.79	77.26
	STD	26.67	51.28	31.04	36.50	44.47	12.97	16.12
F6	Mean	0.0333	1.87	1.07	2.07	1.17	3.43	8.87
	Best	0.0	0.0	0.0	0.0	0.0	0.0	1.0
	Worst	1.0	8.0	4.0	7.0	8.0	10.0	16.0
	STD	0.180	1.63	1.03	1.86	1.53	3.12	3.64
F7	Mean	0.0138	0.334	0.142	0.122	0.0704	0.169	0.248
	Best	0.00566	0.145	0.0515	0.149	0.0167	0.0850	0.112
	Worst	0.0252	0.495	0.314	0.593	0.152	0.297	0.560
	STD	0.00428	0.0917	0.0486	0.0984	0.0272	0.0557	0.0926
F8	Mean	-12454	-12569	-12569	-12479	-12569	-12559	-12064
	Best	-12569	-12569	-12569	-12569	-12569	-12569	-12569
	Worst	-12569	-9982	-12569	-10137	-12569	-12265	-9462
	STD	2.31e-07	481.94	7.01e-12	437.74	9.48e-11	54.64	927.92
F9	Mean	3.54e-06	2.52e-06	3.62e-03	3.44e-08	3.76e-05	3.58e-03	3.43e-02
	Best	1.04e-07	5.51e-10	1.77e-11	0.00744	3.73e-10	0.995	9.55
	Worst	3.13e-05	36.98	6.21	42.19	0.995	53.98	65.86
	STD	6.26e-06	7.47	1.25	8.49	0.179	13.21	14.08
F10	Mean	0.000219	1.33	0.00038	2.02	7.67e-05	0.0478	2.09
	Best	7.20e-05	1.15e-09	2.98e-08	1.79e-09	6.69e-08	1.03e-07	0.00452
	Worst	0.000571	19.94	0.0111	19.96	0.00142	0.652	2.82
	STD	0.000124	4.96	0.00200	5.97	0.000264	0.161	0.542
F11	Mean	0.00115	0.00241	0.00165	0.00104	0.00111	0.00780	0.00934
	Best	3.77e-08	9.99e-16	5.55e-16	4.44e-16	8.33e-15	2.22e-16	9.99e-16
	Worst	0.0148	0.0807	0.0489	0.0855	0.0637	0.0246	0.0514
	STD	0.00358	0.0239	0.0131	0.0279	0.0145	0.00819	0.0128
F12	Mean	1.08e-08	0.0138	0.00079	0.0426	1.90e-12	0.702	1.11
	Best	1.69e-09	3.76e-21	9.00e-18	1.60e-19	7.40e-15	9.81e-16	3.23e-13
	Worst	6.00e-08	0.104	0.0234	1.28	2.21e-11	13.52	9.24
	STD	1.16e-08	0.0352	0.00419	0.229	4.46e-12	2.74	2.52
F13	Mean	-1.0316	-1.0316	-1.0316	-1.0316	-1.0316	-1.0316	-1.0316
	Best	-1.0316	-1.0316	-1.0316	-1.0316	-1.0316	-1.0316	-1.0316
	Worst	-1.0316	-1.0316	-1.0316	-1.0316	-1.0316	-1.0316	-1.0316
	STD	4.48e-16	2.22e-16	2.22e-16	2.22e-16	2.22e-16	4.94e-08	2.22e-16
F14	Mean	1.99	2.0	1.95	1.93	2.0	2.0	1.98
	Best	1.71	2.0	0.391	2.0	2.0	0.000873	1.49
	Worst	2.0	2.0	2.0	2.0	2.0	2.0	2.0
	STD	0.0523	0.0	0.289	0.0	0.0	0.359	0.0916

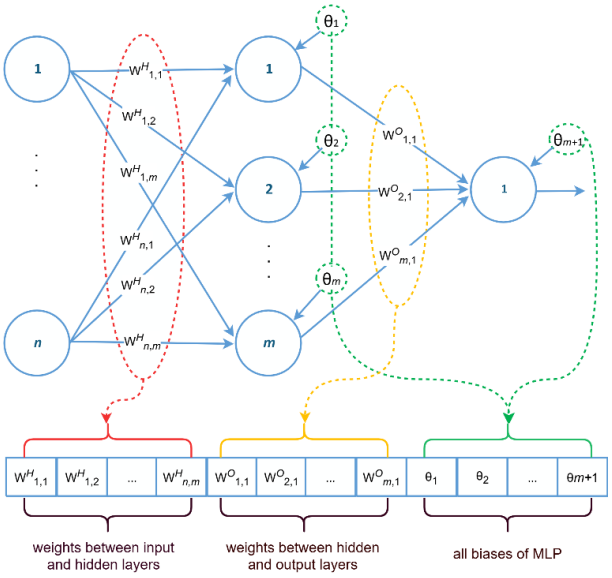

Figure. 2 Mapping weights and biases to a coyote vector

The XCOA and the original COA were tested using a population of 20 packs, with 5 coyotes. Several XCOA variants were evaluated, differing in influence intervals (10, 20, 30, and 40 iterations with LP size = 5) and LP sizes (10 and 15 coyotes with an influence interval of 30), denoted as XCOA1–XCOA6. These sensitivity variants were designed to investigate the effect of LP control parameters on performance. Table 3 shows the results of COA, XCOA, and their variants on standard benchmark functions.

Shorter influence intervals (XCOA1, XCOA3) positively affected stability on multimodal functions, and longer intervals (XCOA5) positively affected adaptability in complex landscapes but not in velocity of convergence. Increasing the size of the LP (XCOA5, XCOA6) enhanced the performance on high-variance functions but had a little worse-case errors, which points to the possibility of over-centralization. Overall, LP size and influence interval were critical parameters, though some functions (F13–F14) showed minimal sensitivity. This ablation study confirms XCOA's parameter sensitivity and the significant role of the LP technique.

3.3 The proposed model: XCOA-MLP

This study integrated XCOA with an MLP neural network to develop an efficient classification model termed XCOA-MLP. The aim is to leverage XCOA's optimization capabilities to determine the optimal weight and bias parameters for MLP in medical data classification. The selection of MLP was prompted by its ability to learn the non-linear nature of certain processes and complex decision boundaries required during the classification of medical informatics [2].

3.3.1. Solution representation

In XCOA-MLP, the vector encoding is used to represent each coyote, and each vector is mapped to a solution in an MLP. The MLP was designed so that the weights and the biases were arranged in a manner that every coyote could map directly to a potential solution within the network, and that according to the format prescribed by the XCOA algorithm. The vector is split into input-hidden weights, hidden output weights and biases. The size of the vector is the sum of the weights and biases in the network.

Fig. 2 depicts an example of mapping the weights and biases of the MLP with two inputs, a single hidden layer with three nodes, and one output neuron to a coyote vector. If there are n input nodes, m hidden nodes, and r output nodes, then the length of each coyote vector (D) is calculated using Eq. (19).

$$D = (n \times m) + (m \times r) + m + r \tag{19}$$

This unified representation allows the XCOA to manipulate the entire parameter set as a single search agent, facilitating systematic exploration of the high-dimensional space of network configurations. Throughout the evolutionary process, XCOA iteratively adapts D to minimize the training error, thereby improving the predictive performance of the MLP.

3.3.2. Fitness function

Each solution vector (coyote) in the population, represented as MLP_j, where $1 \leq j \leq N$ and N denotes the population size, is evaluated using a fitness function to assess the solution quality. Specifically, the MSE, as defined in Eq. (20), serves as the fitness function for evaluating the performance of each candidate solution.

$$fitness(MLP) = MSE = \frac{1}{n} \sum_{k=1}^{n} (y_k - \acute{y}_k)^2 \tag{20}$$

where n is the number of training samples, y'_k is the predicted output for the k^{th} training sample, and y_k is the actual binary target. MSE in classification tasks indicates the difference between the calculated real output and the required binary classification output.

Since MSE is a quadratic function, it enforces high values for significant discrepancies between predicted and target values. Hence, meta-heuristic methods can be used as a fitness function to determine the MLP weights and biases [14]. The goal of the XCOA-MLP model is to minimize the fitness function f. The quality of each solution was

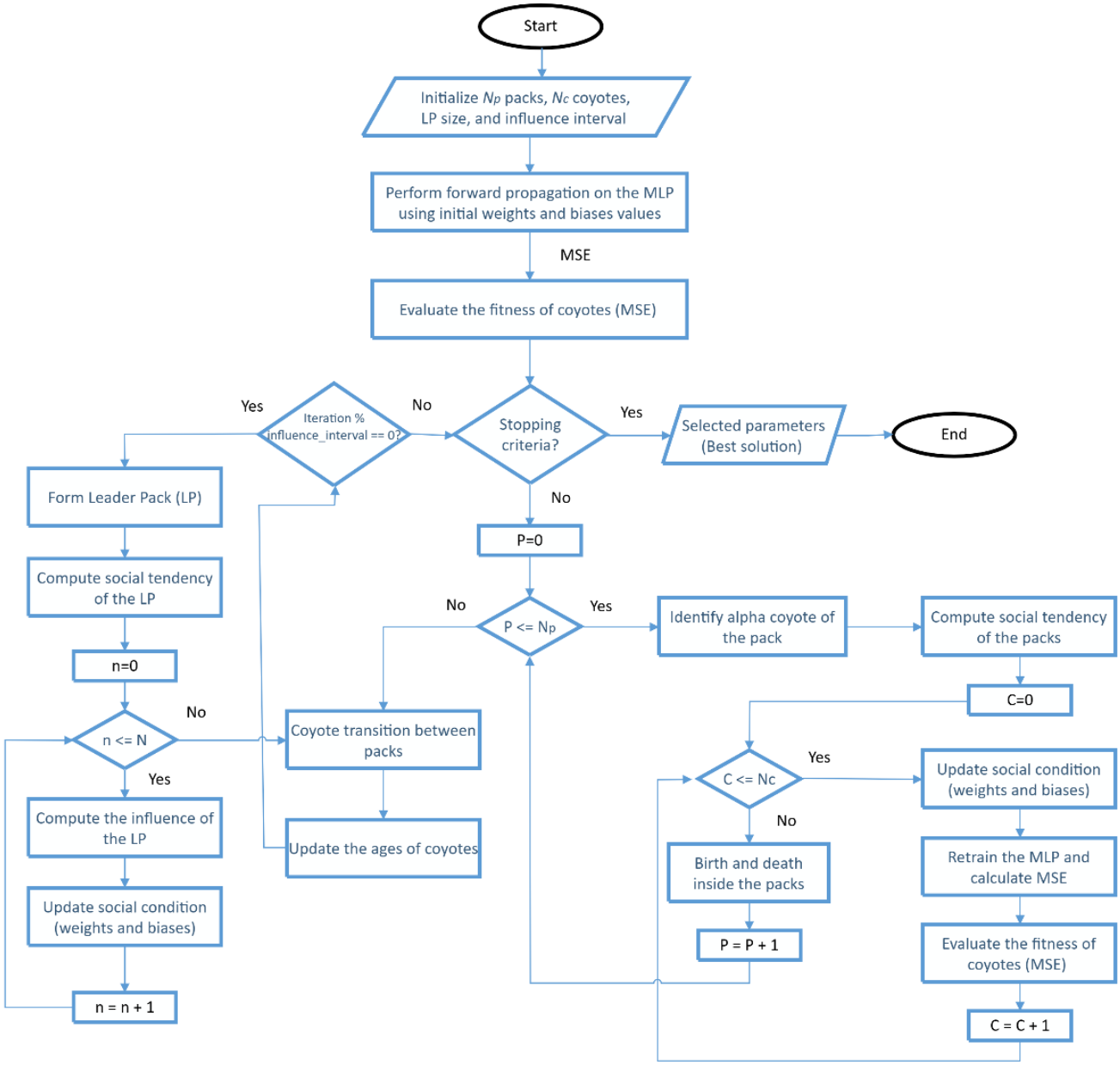

Figure. 3 Flowchart of the XCOA-MLP training process for optimizing MLP in medical data classification

quantified using a previously described method (21). The optimal MLP configuration is identified by selecting the solution with the lowest fitness, as described in Eq. (22).

$$f(MLP_j) = fitness(MLP_j) \tag{21}$$

$$MLP_{best} = MLP_l | f(MLP_l) < f(MLP_j) \forall j \neq l \tag{22}$$

3.3.3. Training process in XCOA-MLP

The training process of MLPs in the XCOA-MLP model starts with the random initialization of coyote agents. Input data are propagated to the network, and outputs are evaluated using the MSE fitness function. The XCOA iteratively refines solutions through:

1. **Coyote initialization:** A population of coyotes, N_c, is generated by assigning random weights and biases to MLP. The population is divided into N_p packs.
2. **Forward propagation:** Training samples of data are used for forward propagation in the MLP.
3. **Fitness evaluation:** The fitness of each coyote is evaluated using MSE, which measures the quality of the weights and biases assigned to the MLP.

4. **Cultural tendency update:** Each pack computes a cultural tendency vector that reflects the central behavior of its members.
5. **Pack evolution:**

- **Alpha identification:** The coyote with the lowest MSE in each pack is selected as alpha.
- **Social update:** Coyotes update their positions in the solution space by learning from the alpha and the cultural tendency of the pack.
- **Birth and death:** Offspring are generated by combining characteristics of selected parents. If a new coyote has better fitness than the worst member of the pack, it replaces that individual.

6. **LP formation:** If the current iteration aligns with the influence interval, an LP is formed by selecting top-performing coyotes across all packs.
7. **LP cultural tendency:** The cultural tendency of the LP was computed, and this information was used to influence other coyotes periodically.
8. **Migration and Aging**: Apply pack transitions and increment coyote ages.

Algorithm 1 represents the pseudo-code of the XCOA-MLP model, and Fig. 3 shows a flowchart of the XCOA-MLP training process for MLP optimization in medical data classification.

Algorithm 1 Pseudo code for XCOA-MLP model

```
define coyote vector (D) (Eq. (19)), influence_interval, LP_size
set up Np packs, Nc coyotes representing weights and biases (Eq. (2))
perform forward propagation in the MLP based on the coyote vector
evaluate the fitness of each coyote based on MSE (Eq. (2)0)
while stopping criterion is not achieved do
    for each pack p do
        identify alpha coyote (best MLP in each pack) (Eq. (4))
        calculate social tendency of pack based on the alpha (Eq. (5))
        for each coyote c of the pack p do
            revise the social condition (weights and biases) (Eq. (11))
            evaluate updated social condition by retraining MLP
            update coyote's adaptation based on (Eq. (13))
        end for
        run the birth and death process (Eq. (6))
    end for
    if iteration % influence_interval == 0 then
        form the LP with top-performing coyotes in all packs (Eq. (15))
        calculate the cultural tendency of the LP (Eq. (16))
        for each pack p do
            for each coyote c of the pack p do
                compute the influence of the LP (Eq. (17))
                update social condition with the LP influence (Eq. (18))
            end for
        end for
    end if
    transitions of coyotes among packs (Eq. (19))
    increment the age of each coyote
end while
```

3.4 Experiment configuration

The MLP architecture used throughout the experiments was a feed-forward network with a single hidden layer and a Sigmoid activation function. The weights and biases of MLPs were constrained to the range [-1, 1], as adopted from [2, 30, 31]. The number of input neurons (n) corresponds to the dataset features, and hidden neurons (m) are calculated using a heuristic from [32], balancing complexity and generalization:

$$m = 2 \times n + 1 \tag{23}$$

Each experimental run was initialized with new, random parameters, and the experiment was executed 30 times independently. The size of the MHAs consisted of 50 populations. The experiments were

Table 4. Parameter settings of XCOA and other MHAs

Algorithm	Parameter	Value
XCOA-MLP	Packs	20
	Coyotes per pack	5
	LP size	5
	Influence interval	30
BAT	Loudness	0.5
	Pulse rate	0.5
	Min-Max frequency	0-1
CS	Step size	0.3
	Lévy distribution	1.5
	Nest replacement prob.	0.2
DE	Crossover prob.	0.9
	Weight	0.5
GA	Crossover prob.	0.9
	Mutation prob.	0.01
	Selection prob.	0.5
	Selection type	Roulette
GWO	*a*	2 to 0 linear
MFO	Spiral constant	1
MVO	Min wormhole prob	0.2
	Max wormhole prob	1.0
PSO	Inertia weight	0.7
	Cognitive component	1.1
	Social component	1.7

Table 5. Characteristics of the datasets used in this study

Dataset	Cols	Rows	Label 1	Label 2	Network Structure Eq. (23)	Vector Length Eq. (19)
Breast cancer	9	699	(458)	(241)	9-19-1	210
Diabetes	8	768	(268)	(500)	8-17-1	171
Liver	6	345	(200)	(145)	6-13-1	105
Parkinsons	22	195	(147)	(48)	22-45-1	1081
Vertebral	6	310	(210)	(100)	6-13-1	105

conducted on an Intel i7 2.11 GHz processor (8 cores, 16 GB RAM). The parameter configurations for COA, XCOA, and all competing MHAs are summarized in Table 4.

It is important to note that BP-MLP may be implemented with either a fixed or a varying random seed. The former provides deterministic results, while the latter employs a stochastic initialization that aligns with MHA training. A variable random seed was used in this study to guarantee the fair estimation of variance in 30 independent runs.

The efficacy of the XCOA-MLP is assessed by using five medical datasets from the UCI [12]: Breast cancer, Diabetes, Liver, Parkinsons, and Vertebral. A brief description of all the datasets, along with the network configuration and vector length, is provided in Table 5, based on Eqs. (19) and (23).

The datasets were separated into two sets: training (66%) and testing (34%). The stratified sampling technique was used [33]. The max-min

normalization procedure was used to scale all attribute values, as shown in Eq. (24).

$$X'_i = \frac{X_i - min_X}{max_X - min_X} \tag{24}$$

where X_i is the initial value of feature 'X' for instance i, and X'_i is its normalized value. The min_X and max_X represent the lower and upper limits of the feature X_i.

For comparison, the XCOA-MLP model was benchmarked with some popular MHAs such as bat algorithm (BAT), cuckoo search (CS), differential evolution (DE), genetic algorithm (GA), grey wolf optimizer (GWO), moth-flame optimization (MFO), multi-verse optimizer (MVO), and particle swarm optimization (PSO). The original COA was also considered as a baseline to identify the improvements. In addition, conventional ML classifiers like Naïve bayes (NB), decision tree (DT), random forest (RF), support vector machine (SVM) and extreme gradient boosting (XGBoost) were considered to get a broader performance perspective.

In order to measure performance, accuracy, sensitivity, specificity, and MSE were observed as tools. These measures were calculated on the basis of the true positive (TP), true negative (TN), false positive (FP) and false negative (FN) with the use of the following Eqs. (25) to (27).

$$Sensitivity = \frac{TP}{(TP+FN)} \tag{25}$$

$$Specificity = \frac{TN}{(TN+FP)} \tag{26}$$

$$Accuracy = \frac{(TN+TP)}{(TN+TP+FN+FP)} \tag{27}$$

4. Results and discussion

4.1 Classification accuracy

Table 6 presents the mean and STD of accuracy values. XCOA-MLP achieved the highest mean test accuracy in three datasets: Breast cancer (0.978), Diabetes (0.782), and Parkinsons (0.877). On the Vertebral dataset, DE slightly outperformed XCOA-MLP, while PSO achieved the best accuracy for the Liver dataset. Nonetheless, XCOA-MLP remained

Table 6. Mean classification accuracy with standard deviation across all algorithms

MHA	Breast cancer	Diabetes	Liver	Parkinsons	Vertebral
BAT-MLP	0.954 0.007	0.704 0.014	0.657 0.013	0.808 0.013	0.797 0.013
CS-MLP	0.965 0.002	0.731 0.013	0.689 0.014	0.835 0.012	0.840 0.012
DE-MLP	0.965 0.003	0.728 0.010	0.704 0.012	0.845 0.006	**0.852 0.006**
GA-MLP	0.963 0.002	0.728 0.010	0.705 0.013	0.833 0.005	0.838 0.007
GWO-MLP	0.959 0.006	0.680 0.015	0.652 0.015	0.800 0.014	0.783 0.012
MFO-MLP	0.966 0.002	0.731 0.008	0.668 0.013	0.846 0.009	0.847 0.010
MVO-MLP	0.966 0.002	0.731 0.008	0.693 0.012	0.852 0.008	0.837 0.007
PSO-MLP	0.958 0.003	0.710 0.014	**0.721 0.011**	0.823 0.012	0.825 0.004
BP-MLP	0.960 0.009	0.731 0.032	0.579 0.008	0.830 0.029	0.682 0.012
COA-MLP	*0.968 0.009*	*0.771 0.014*	0.704 0.032	*0.862 0.019*	0.841 0.027
XCOA-MLP	**0.978 0.006**	**0.782 0.015**	*0.710 0.032*	**0.877 0.023**	*0.851 0.024*

Table 7. Mean specificity and sensitivity across all algorithms

MHA	Breast cancer		Diabetes		Liver		Parkinsons		Vertebral	
	Sp	***Sn***	***Sp***	***Sn***	***Sp***	***Sn***	***Sp***	***Sn***	***Sp***	***Sn***
BAT-MLP	0.987	0.892	0.842	0.267	0.425	0.830	0.593	0.878	0.925	0.555
CS-MLP	0.987	0.934	0.863	0.483	0.538	0.804	0.546	0.868	0.905	0.702
DE-MLP	0.986	0.923	0.871	0.463	0.553	0.818	0.739	0.878	0.920	0.705
GA-MLP	0.987	0.920	0.722	0.702	0.528	0.841	0.718	0.868	0.927	0.649
GWO-MLP	**0.988**	0.907	**0.980**	0.122	0.412	0.831	0.426	0.921	0.925	0.487
MFO-MLP	0.987	0.928	0.879	0.453	0.470	0.818	*0.743*	0.881	**0.928**	0.673
MVO-MLP	0.987	0.927	0.869	0.472	0.546	0.805	**0.749**	0.885	0.918	0.664
PSO-MLP	0.987	0.905	*0.919*	0.323	*0.589*	0.821	0.655	0.878	0.918	0.625
BP-MLP	0.965	0.892	0.887	0.437	0.027	0.852	0.438	0.923	0.927	0.016
COA-MLP	0.972	**0.963**	0.866	*0.582*	0.561	*0.809*	0.606	*0.947*	0.852	*0.782*
XCOA-MLP	0.972	*0.960*	0.870	**0.612**	**0.600**	**0.861**	0.640	**0.949**	0.855	**0.824**

Table 8. Mean and standard deviation of MSE values across all algorithms

MHA	Breast cancer	Diabetes	Liver	Parkinsons	Vertebral
BAT-MLP	0.044 0.009	0.186 0.010	0.234 0.009	0.112 0.009	0.155 0.009
CS-MLP	0.025 0.001	0.158 0.003	0.196 0.007	*0.084 0.007*	0.135 0.003
DE-MLP	*0.024 0.003*	0.161 0.004	0.205 0.007	0.089 0.004	0.137 0.004
GA-MLP	0.025 0.001	0.167 0.002	0.214 0.007	0.093 0.002	0.139 0.009
GWO-MLP	0.070 0.005	0.196 0.006	0.194 0.009	0.140 0.009	0.177 0.007
MFO-MLP	0.024 0.001	0.166 0.003	0.208 0.007	0.087 0.005	0.139 0.005
MVO-MLP	0.024 0.002	0.160 0.004	0.201 0.007	**0.085 0.004**	0.136 0.003
PSO-MLP	0.032 0.002	0.182 0.006	0.199 0.007	0.097 0.008	0.144 0.003
BP-MLP	0.040 0.009	0.269 0.012	0.421 0.008	0.170 0.019	0.318 0.012
COA-MLP	0.025 0.005	*0.154 0.007*	*0.184 0.012*	0.099 0.012	*0.111 0.013*
XCOA-MLP	**0.023 0.004**	**0.152 0.012**	**0.183 0.013**	0.096 0.025	**0.106 0.024**

highly competitive across all benchmarks, consistently ranking among the top performers. These findings highlight the effectiveness of the leader pack technique in improving convergence and generalization in medical classification tasks.

4.2 Specificity and sensitivity

The balance between true positive, sensitivity (Sn) and true negative, specificity (Sp) predictions is reported in Table 7. XCOA-MLP provided a stable trade-off between Sp and Sn across all datasets. These results indicate that the proposed XCOA-MLP model reduces the class bias and maintains diagnostic reliability across medical datasets.

4.3 Mean squared error (MSE)

Table 8 presents the mean and STD of the MSE values. XCOA-MLP model consistently achieved the lowest or near-lowest MSE values across all datasets. It ranked first in Breast cancer (0.023), Diabetes (0.152), Liver (0.183), and Vertebral (0.106) conditions. These results confirm the learning

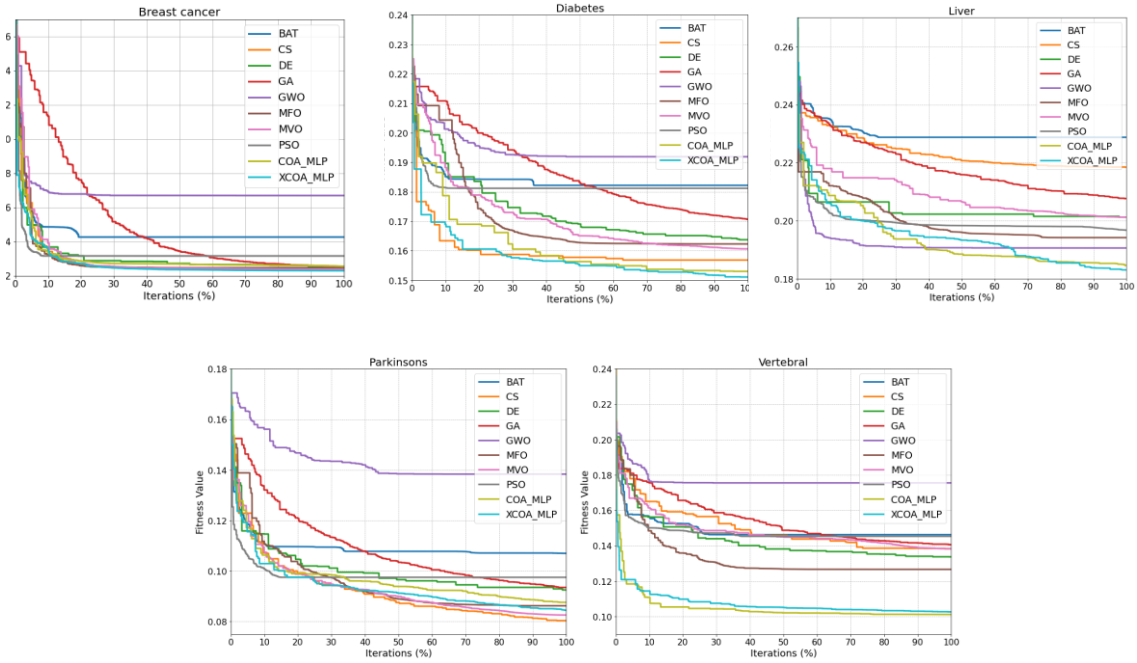

Figure. 4 Convergence curves of fitness across the five datasets: Breast cancer, Diabetes, Liver, Parkinsons, and Vertebral

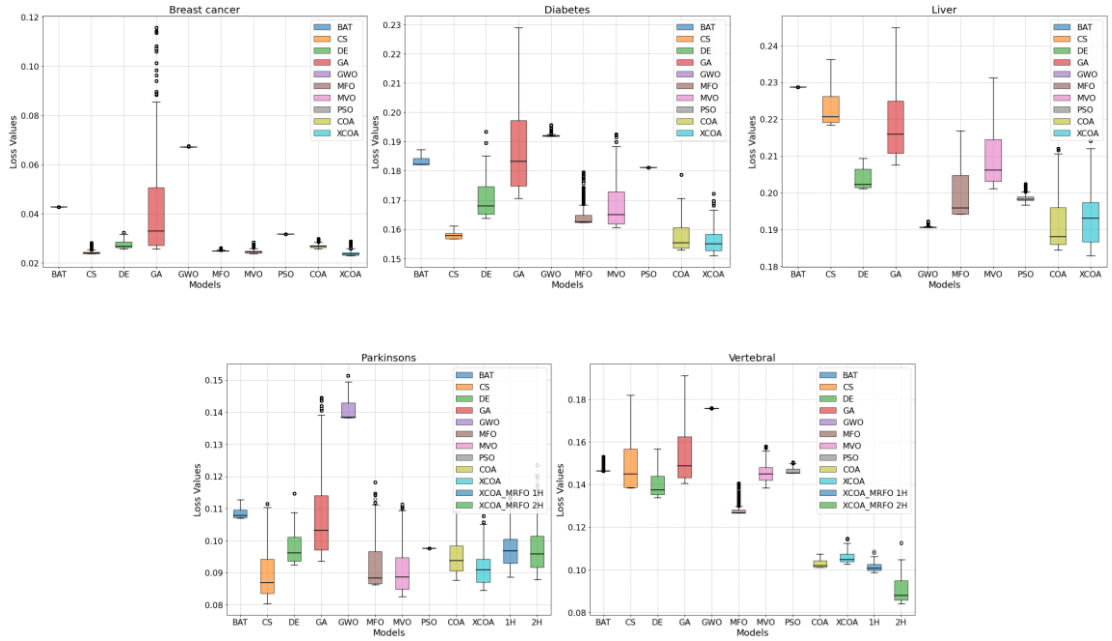

Figure. 5 Box plots of fitness distributions across the five datasets: Breast cancer, Diabetes, Liver, Parkinsons, and Vertebral

efficiency of XCOA, which effectively balances exploration and exploitation, reducing prediction error while preserving robustness.

4.4 Discussion

The proposed XCOA-MLP model consistently demonstrated superior or highly competitive performance across all datasets. Its strength lies in achieving low MSE, high accuracy, and balanced sensitivity and specificity, which are critical factors in medical data classification. Low STDs further highlight the model's stability and reliability. The incorporation of the LP significantly improved convergence, and generalization compared to other methods.

Fig. 4 illustrates the convergence behavior, showing that XCOA-MLP rapidly approaches

Table 9. Wilcoxon Signed Rank test results on the accuracy of XCOAMLP and the other algorithm

XCOA vs.	Metric	Breast cancer	Diabetes	Liver	Parkinsons	Vertebral
BAT	p-value	1.86e-09	1.86e-09	3.54e-08	3.73e-09	5.59e-09
	Effect size (r)	1.000	1.000	1.000	1.000	1.000
	95% CI	[0.020380, 0.027419]	[0.069161, 0.081206]	[0.043044, 0.067395]	[0.060035, 0.079975]	[0.043979, 0.065399]
CS	p-value	5.59e-09	1.86e-09	0.001864	1.02e-07	0.052263
	Effect size (r)	1.000	1.000	0.568	0.972	0.354
	95% CI	[0.010201, 0.015722]	[0.038599, 0.051847]	[0.010909, 0.035272]	[0.033187, 0.052823]	[-0.002788, 0.019118]
DE	p-value	5.59e-09	1.86e-09	0.328470	1.68e-06	0.700033
	Effect size (r)	1.000	1.000	0.178	0.874	-0.070
	95% CI	[0.010038, 0.015885]	[0.047073, 0.057526]	[-0.004108, 0.020288]	[0.025389, 0.043375]	[-0.010348, 0.009011]
GA	p-value	1.86e-09	1.86e-09	0.416130	1.86e-08	0.005013
	Effect size (r)	1.000	1.000	0.148	1.000	0.512
	95% CI	[0.012207, 0.017716]	[0.047073, 0.057526]	[-0.005102, 0.019282]	[0.038538, 0.056226]	[0.002518, 0.022145]
GWO	p-value	1.86e-09	1.86e-09	1.30e-08	1.86e-09	1.86e-09
	Effect size (r)	1.000	1.000	1.000	1.000	1.000
	95% CI	[0.014593, 0.021742]	[0.093771, 0.105403]	[0.049073, 0.073846]	[0.067921, 0.088090]	[0.058029, 0.079034]
MFO	p-value	8.01e-08	1.86e-09	8.33e-07	1.99e-06	0.308521
	Effect size (r)	0.980	1.000	0.900	0.868	0.186
	95% CI	[0.009207, 0.014716]	[0.044992, 0.055138]	[0.031890, 0.056290]	[0.023022, 0.041742]	[-0.007819, 0.012483]
MVO	p-value	9.31e-09	1.86e-09	0.018529	2.08e-05	0.001864
	Effect size (r)	1.000	1.000	0.430	0.777	0.568
	95% CI	[0.010062, 0.015638]	[0.044740, 0.054946]	[0.006885, 0.031295]	[0.018194, 0.036570]	[0.003598, 0.023066]
PSO	p-value	1.86e-09	1.86e-09	0.100397	9.31e-09	7.99e-06
	Effect size (r)	1.000	1.000	-0.300	1.000	0.815
	95% CI	[0.017134, 0.022978]	[0.064080, 0.075662]	[-0.020931, 0.003111]	[0.045284, 0.064726]	[0.015468, 0.034154]
BP	p-value	1.86e-08	3.86e-07	1.86e-09	2.05e-07	1.86e-09
	Effect size (r)	1.000	0.927	1.000	0.948	1.000
	95% CI	[0.015599, 0.022804]	[0.032943, 0.060492]	[0.131558, 0.156578]	[0.034931, 0.059845]	[0.159292, 0.180331]
COA	p-value	0.001039	0.009932	0.279315	0.003497	0.008143
	Effect size (r)	0.599	0.471	0.198	0.533	0.483
	95% CI	[0.005074, 0.013926]	[0.006891, 0.022193]	[-0.001883, 0.018832]	[0.008528, 0.026248]	[0.007243, 0.018625]

minimal fitness values in four of the five datasets. In contrast, CS achieved the best result for Parkinsons but stagnated in Diabetes and Vertebral. Similarly, GWO was frequently trapped in local minima across four datasets.

Fig. 5 provides further evidence through box plots, where XCOA-MLP achieved the most favorable distributions in four datasets, with narrow interquartile ranges, lower median losses, and fewer outliers. In comparison, GWO, BAT, and GA exhibited wider variability and inferior central tendencies, reflecting weaker optimization stability.

To validate the superiority of XCOA-MLP, the

Table 10. Friedman ranks of algorithms for all datasets

Algorithm	Avg. Rank	Friedman Rank
XCOA-MLP	**2.6**	**1**
COA-MLP	3.1	2
MFO-MLP	4.1	3
DE-MLP	4.4	4
MVO-MLP	4.6	5
CS-MLP	5	6
GA-MLP	6	7
PSO-MLP	6.8	8
BAT-MLP	9.4	9
GWO-MLP	9.8	10
BP-MLP	10.2	11

Table 11. Comparison of XCOA-MLP and ML models

Dataset	Model	Acc.	MSE	Sp	Sn
Breast cancer	DT	0.916	0.084	0.942	0.866
	RF	0.950	0.050	0.968	0.915
	SVM	0.945	0.055	0.955	0.927
	KNN	0.958	0.042	*0.973*	0.927
	XGBoost	0.954	0.046	0.962	*0.939*
	XCOA-MLP	**0.978**	**0.023**	**0.972**	**0.960**
Diabetes	DT	0.698	0.302	0.778	0.549
	RF	0.714	0.286	0.807	0.538
	SVM	*0.744*	*0.256*	*0.865*	0.516
	KNN	0.721	0.279	0.825	0.527
	XGBoost	0.714	0.286	0.801	*0.549*
	XCOA-MLP	**0.782**	**0.152**	**0.870**	**0.612**
Liver	DT	0.678	0.322	0.588	0.746
	RF	**0.737**	*0.263*	*0.667*	0.791
	SVM	0.703	0.297	0.451	**0.896**
	KNN	0.602	0.398	0.588	0.612
	XGBoost	0.703	0.297	**0.686**	0.716
	XCOA-MLP	*0.710*	**0.183**	0.600	*0.861*
Parkinsons	DT	0.791	0.209	**0.812**	0.784
	RF	0.776	0.224	*0.750*	0.784
	SVM	0.851	0.149	0.500	0.961
	KNN	0.866	0.134	0.625	0.941
	XGBoost	**0.910**	*0.090*	0.750	**0.961**
	XCOA-MLP	*0.877*	**0.096**	0.640	*0.949*
Vertebral	DT	0.783	0.217	0.805	0.724
	RF	0.811	0.189	**0.870**	0.655
	SVM	*0.823*	*0.186*	0.850	0.645
	KNN	0.745	0.255	0.831	0.517
	XGBoost	0.802	0.198	0.831	*0.724*
	XCOA-MLP	**0.851**	**0.106**	*0.855*	**0.824**

Wilcoxon Signed-Rank Test was conducted on the accuracy results over 30 runs against all competitor models, as shown in Table 9. The analysis reports two-sided p-values, signed effect sizes (r), and 95% confidence intervals (CIs) for the median paired accuracy differences. A positive r indicates that XCOA-MLP outperforms the comparator, while a negative r indicates that the competing model performs better. Confidence intervals were estimated using the normal approximation around the median paired difference.

The results show that in most cases, the obtained p-values are below the 0.05 significance threshold, indicating that the superiority of XCOA-MLP over the compared algorithms is statistically significant. The corresponding effect sizes generally exhibit large magnitudes, reflecting a strong and practically relevant improvement. Moreover, the 95% CIs are distinct and non-degenerate, capturing the variability of the median paired differences across datasets.

are strictly positive, which further confirms that XCOA-MLP achieves consistently higher accuracy.

The Friedman test, based on accuracy results, produced a statistic of $\chi^2 = 33.63$, confirming significant performance differences among the algorithms. As shown in Table 10, XCOA-MLP obtained the best rank (2.6), confirming the strength of XCOA-MLP over the compared MHA-based algorithms.

To provide a comprehensive evaluation of the proposed XCOA-MLP model, its classification performance was compared against six widely used traditional ML classifiers: DT, RF, SVM, KNN, and XGBoost. The results are summarized in Table 11.

Table 11 shows that XCOA-MLP outperformed traditional ML classifiers in most datasets, attaining the highest or near-highest accuracy and lowest MSE. It also provided superior sensitivity, essential for medical diagnosis, while maintaining strong specificity. Although XGBoost and KNN were competitive, XCOA-MLP achieved a better trade-off between accuracy and generalization, confirming the effectiveness of the proposed training approach.

To further validate the effectiveness of XCOA-MLP, its performance was compared with several recent MHA-based models, including MGO-MLP [7], FOX-MLP [16], PYYPO [34], GGA-MLP [14], DA-MLP [35], HOS-MLP [36], and MPA-MLP [37]. Table 12 presents the results across the five medical datasets.

Table 12. Comparison of XCOA-MLP with existing MHA-based models in the literature (results are taken from respective papers and may reflect different experimental settings such as preprocessing, data splits, or network sizes)

Reference / Method	Breast cancer	Diabetes	Liver	Parkinson	Vertebral
MGO-MLP	0.976	0.754	**0.765**	0.870	*0.882*
MLP-FOX	0.967	*0.780*	–	0.871	0.833
PYYPO	*0.977*	0.756	*0.762*	*0.876*	0.879
GGA-MLP	–	0.769	–	0.866	**0.885**
DA-MLP	0.958	–	–	–	–
HOS-MLP	0.968	0.778	–	–	0.924
MPA-MLP	–	0.765	0.710	0.871	0.848
XCOA-MLP	**0.978**	**0.782**	0.710	**0.877**	0.851

The comparison with recent state-of-the-art methods in Table 12 is based on their published results as reported in the respective studies. Therefore, minor variations in preprocessing, data partitioning, or network configuration may exist. The results highlight the competitiveness of XCOA-MLP, which achieved the highest accuracy on Breast cancer, Diabetes, and Parkinsons, outperforming recent state-of-the-art methods. While it was slightly behind on Liver and Vertebral datasets, it remained competitive and consistently superior to the baseline COA-MLP, confirming its robustness as a state-of-the-art solution for medical data classification.

5. Conclusion

This study introduced XCOA-MLP, a hybrid model combining the extended Coyote Optimization Algorithm with a Multilayer Perceptron for medical data classification. Employing the leader pack technique enhanced convergence speed, exploitation, and population diversity. Experimental results confirmed that the XCOA-MLP model consistently outperformed conventional training and metaheuristic-based MLPs, achieving accuracies of 97.8% (Breast cancer), 78.2% (Diabetes), and 87.7% (Parkinsons), with lower MSE and balanced specificity–sensitivity values. Statistical validation using non-parametric tests ($p < 0.01$) verified the significance of these improvements. These results highlight the scientific contribution of this work: XCOA-MLP establishes an effective and reliable training framework for neural networks in medical applications, while also providing a general strategy for enhancing swarm intelligence algorithms in high-dimensional optimization problems.

Future work will explore automated architecture selection, dynamic hyperparameter tuning, alternative fitness functions (e.g., cross-entropy), and integration with deep learning models for larger medical datasets.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

Maher Talal Al-Asaady: conceptualization, methodology, software, original draft, visualization. Teh Noranis Mohd Aris: validation, resources, supervision, project administration, writing review. Nurfadhlina Mohd Sharef: writing review and editing. Hazlina Hamdan: writing review and editing.

References

[1] M. T. Alasaady, T. N. M. Aris, N. M. Sharef, H. Hamdan, "A Proposed Approach for Diabetes Diagnosis Using Neuro-Fuzzy Technique", *Bulletin of Electrical Engineering and Informatics*, Vol. 11, No. 6, pp. 3590-3597, 2022.

[2] M. G. Rojas, A. C. Olivera, P. J. Vidal, "Optimising Multilayer Perceptron Weights and Biases through a Cellular Genetic Algorithm for Medical Data Classification", *Array*, Vol. 14, p. 100173, 2022, doi: 10.1016/j.array.2022.100173.

[3] V. G. Gudise, G. K. Venayagamoorthy, "Comparison of Particle Swarm Optimization and Backpropagation as Training Algorithms for Neural Networks", In: *Proc. of IEEE Swarm Intelligence Symposium (SIS'03)*, pp. 110-117, 2003.

[4] A. M. Hemeida, et al., "Nature-Inspired Algorithms for Feed-Forward Neural Network Classifiers: A Survey of One Decade of Research", *Ain Shams Engineering Journal*, Vol. 11, No. 3, pp. 659-675, 2020, doi: 10.1016/j.asej.2020.01.007.

[5] B. Tarle, R. Tajanpure, S. Jena, "Medical Data Classification Using Different Optimization Techniques: A Survey", *International Journal of Research in Engineering and Technology*, Vol. 5, No. 5, pp. 101-108, 2016.

[6] H. Jiang, Y. Wang, Y. Tian, X. Zhang, J. Xiao, "Feature Construction for Meta-Heuristic Algorithm Recommendation of Capacitated Vehicle Routing Problems", *ACM Transactions on Evolutionary Learning and Optimization*, Vol. 1, No. 1, 2021.

[7] M. Karabiyik, B. Turkoglu, T. Asuroglu, "Optimizing Artificial Neural Networks Using Mountain Gazelle Optimizer", *IEEE Access*, Vol. 13, pp. 50464-50479, 2025, doi: 10.1109/ACCESS.2025.3552831.

[8] S. Raziani, S. Ahmadian, S. M. J. Jalali, A. Chalechale, "An Efficient Hybrid Model Based on Modified Whale Optimization Algorithm and Multilayer Perceptron Neural Network for Medical Classification Problems", *Journal of Bionic Engineering*, Vol. 19, No. 5, pp. 1504-1521, 2022, doi: 10.1007/s42235-022-00216-x.

[9] M. T. Al-Asaady, T. N. M. Aris, N. M. Sharef, H. Hamdan, "Recent Advances on Meta-Heuristic Algorithms for Training Multilayer Perceptron Neural Network", *JOIV: International Journal on Informatics Visualization*, Vol. 9, No. 2, pp. 658-673, 2025.

[10] J. Pierezan, L. Coelho, "Coyote Optimization Algorithm: A New Metaheuristic for Global Optimization Problems", In: *Proc. of IEEE Congress on Evolutionary Computation (CEC)*, pp. 1-8, 2018, doi: 10.1109/CEC.2018.8477769.

[11] K. Hussain, M. N. M. Salleh, S. Cheng, R. Naseem, "Common Benchmark Functions for Metaheuristic Evaluation: A Review", *JOIV: International Journal on Informatics Visualization*, Vol. 1, No. 4-2, pp. 218-223, 2017, doi: 10.30630/joiv.1.4-2.65.

[12] A. Frank, "UCI Machine Learning Repository", 2010, [Online]. Available: http://archive.ics.uci.edu/ml.

[13] H. H. Tan, K. H. Lim, "Review of Second-Order Optimization Techniques in Artificial Neural Networks Backpropagation", In: *Proc. of IOP Conference Series: Materials Science and Engineering*, p. 012003, 2019.

[14] P. Bansal, et al., "GGA-MLP: A Greedy Genetic Algorithm to Optimize Weights and Biases in Multilayer Perceptron", *Contrast Media and Molecular Imaging*, Vol. 2022, 2022, doi: 10.1155/2022/4036035.

[15] S. J. Mousavirad, et al., "A Novel Metaheuristic Population Algorithm for Optimising the Connection Weights of Neural Networks", *Evolving Systems*, Vol. 16, No. 1, p. 18, 2025, doi: 10.1007/s12530-024-09641-1.

[16] S. Mohapatra, M. Panda, P. P. Sarangi, B. S. P. Mishra, N. Parihar, B. Sahoo, "MLP-FOX: A Metaheuristic Approach-Based Training Multilayer Perceptron for Data Classification", In: *Proc. of 1st International Conference on Cognitive, Green and Ubiquitous Computing (IC-CGU)*, pp. 1-6, 2024, doi: 10.1109/IC-CGU58078.2024.10530732.

[17] Ş. Gülcü, "An Improved Animal Migration Optimization Algorithm to Train the Feed-Forward Artificial Neural Networks", *Arab Journal of Science and Engineering*, Vol. 47, No. 8, pp. 9557-9581, 2022, doi: 10.1007/s13369-021-06286-z.

[18] O. Altay, E. Altay, "A Novel Hybrid Multilayer Perceptron Neural Network with Improved Grey Wolf Optimizer", *Neural Computing and Applications*, Vol. 35, No. 1, pp. 529-556, 2023, doi: 10.1007/s00521-022-07775-4.

[19] Z. Yang, "FMFO: Floating Flame Moth-Flame Optimization Algorithm for Training Multi-Layer Perceptron Classifier", *Applied Intelligence*, Vol. 53, No. 1, pp. 251-271, 2023, doi: 10.1007/s10489-022-03484-6.

[20] S. A. Rather, P. S. Bala, "Lévy Flight and Chaos Theory-Based Gravitational Search Algorithm for Multilayer Perceptron Training", *Evolving Systems*, Vol. 14, No. 3, pp. 365-392, 2023, doi: 10.1007/s12530-022-09456-y.

[21] M. Hassanin, A. Shoeb, A. E. Hassanien, "Designing Multilayer Feedforward Neural Networks Using Multi-Verse Optimizer", *Handbook of Research on Machine Learning Innovations and Trends*, pp. 1076-1093, 2017.

[22] H. Faris, I. Aljarah, S. Mirjalili, "Training Feedforward Neural Networks Using Multi-Verse Optimizer for Binary Classification Problems", *Applied Intelligence*, Vol. 45, No. 2, pp. 322-332, 2016.

[23] J. H. Zhu, J. S. Wang, X. Y. Zhang, H. M. Song, Z. H. Zhang, "Mathematical Distribution Coyote Optimization Algorithm with Crossover Operator to Solve Optimal Power Flow Problem of Power System", *Alexandria Engineering Journal*, Vol. 69, pp. 585-612, 2023, doi: 10.1016/j.aej.2023.02.023.

[24] S. Wu, J. Jiang, Y. Yan, W. Bao, Y. Shi, "Improved Coyote Algorithm and Application to Optimal Load Forecasting Model", *Alexandria Engineering Journal*, Vol. 61, No. 10, pp. 7811-7822, 2022.

[25] T. T. Nguyen, T. D. Pham, L. C. Kien, L. V. Dai, "Improved Coyote Optimization Algorithm for Optimally Installing Solar Photovoltaic Distribution Generation Units in Radial Distribution Power Systems", *Complexity*, Vol. 2020, 2020.

[26] H. F. Eid, R. F. Mansour, E. Cuevas, "A Modified Variant of Coyote Optimization Algorithm for Solving Ordinary Differential Equations and Oscillatory Mechanical Problems", *Simulation*, Vol. 98, No. 12, pp. 1161-1178, 2022.

[27] Y. Hao, J. Ding, S. Huang, M. Xiao, "Improved Coyote Optimization Algorithm for Parameter Estimation of Lithium-Ion Batteries", In: *Proc. of the Institution of Mechanical Engineers, Part A: Journal of Power and Energy*, Vol. 237, No. 4, pp. 787-796, 2023, doi: 10.1177/09576509221147330.

[28] S. J. Mousavirad, et al., "A Novel Metaheuristic Population Algorithm for Optimising the Connection Weights of Neural Networks", *Evolving Systems*, Vol. 16, No. 1, p. 18, 2025, doi: 10.1007/s12530-024-09641-1.

[29] S. Gupta, K. Deep, "A Novel Hybrid Sine-Cosine Algorithm for Global Optimization and Its Application to Train Multilayer Perceptrons", *Applied Intelligence*, Vol. 50, No. 4, pp. 993-1026, 2020.

[30] A. A. Heidari, H. Faris, I. Aljarah, S. Mirjalili, "An Efficient Hybrid Multilayer Perceptron Neural Network with Grasshopper Optimization", *Soft Computing*, Vol. 23, No. 17, pp. 7941-7958, 2019, doi: 10.1007/s00500-018-3424-2.

[31] H. Faris, S. Mirjalili, I. Aljarah, "Automatic Selection of Hidden Neurons and Weights in Neural Networks Using Grey Wolf Optimizer Based on a Hybrid Encoding Scheme", *International Journal of Machine Learning and Cybernetics*, Vol. 10, pp. 2901-2920, 2019, doi: 10.1007/s13042-018-00913-2.

[32] S. Mirjalili, S. M. Mirjalili, A. Lewis, "Let a Biogeography-Based Optimizer Train Your Multi-Layer Perceptron", *Information Sciences*, Vol. 269, pp. 188-209, 2014.

[33] M. T. Alasaady, M. G. Saeed, K. H. Faraj, "Evaluation and Comparison Framework for Data Modeling Languages", In: *Proc. of 2nd International Conference on Electrical, Communication, Computer, Power and Control Engineering (ICECCPCE)*, pp. 68-73, 2019, doi: 10.1109/ICECCPCE46549.2019.203750.

[34] M. Shekhar, "Training Multi-Layer Perceptron Using Population-Based Yin-Yang-Pair Optimization", In: *Proc. of International Conference on Computational Intelligence*, pp. 417-425, 2021.

[35] Ş. Gülcü, "Training of the Feed-Forward Artificial Neural Networks Using Dragonfly Algorithm", *Applied Soft Computing*, Vol. 124, p. 109023, 2022, doi: 10.1016/j.asoc.2022.109023.

[36] M. Tunay, E. Pashaei, E. Pashaei, "Hybrid Hypercube Optimization Search Algorithm and Multilayer Perceptron Neural Network for Medical Data Classification", *Computational Intelligence and Neuroscience*, Vol. 2022, p. 1612468, 2022, doi: 10.1155/2022/1612468.

[37] J. Bagchi, T. Si, "Artificial Neural Network Training Using Marine Predators Algorithm for Medical Data Classification", In: *Proc. of International Conference on Computational Intelligence (ICCI 2020)*, pp. 137-148, 2022.