UDC 004.8
DOI: 10.15587/1729-4061.2025.326955

IDENTIFYING THE IMPACT OF METAPERCEPTRON IN OPTIMIZING NEURAL NETWORKS: A COMPARATIVE STUDY OF GRADIENT DESCENT AND METAHEURISTIC APPROACHES

Darwin
Doctoral Student, Lecturer*, ***
Tengku Henny Febriana Harumy
Corresponding author
Lecturer*
E-mail: hennyharumy@usu.ac.id
Syahril Efendi
Lecturer*
Carles Juliandy
Lecturer
Department of Information Technology***
Binarwan Halim
Lecturer
Department of Obstetrics and Gynecology**
*Department of Computer Science**
**Universitas Sumatera Utara
Dr. T. Mansur str., 9, Medan, Indonesia, 20155
***Universitas Mikroskil
Thamrin str., 112, Medan, Indonesia, 20212

This study investigates the application of the Metaperceptron framework as an adaptive optimization mechanism in training neural networks for polycystic ovary syndrome (PCOS) diagnosis. The research addresses the persistent challenges in conventional optimization methods such as slow convergence, local minima entrapment, and hyperparameter sensitivity that hinder the efficiency and generalization capability of artificial neural networks. By integrating Metaperceptron with both gradient descent (GD) and genetic algorithm (GA), this work demonstrates significant improvements in convergence speed and diagnostic accuracy. Specifically, Metaperceptron-enhanced GD reduced convergence time by nearly 40% while maintaining high accuracy (0.8950 for single-layer neural network and 0.9100 for multi-layer neural network). These results were achieved through dynamic learning rate adjustment and meta-level control over search strategies, enabling better exploration-exploitation balance during training. The findings are explained by the framework's ability to adaptively respond to gradient landscapes and dataset characteristics, offering a more stable and efficient optimization process. Practical implementation of the proposed method is feasible under conditions where data quality and representativeness are ensured, particularly in medical diagnostics and other domains involving imbalanced or noisy datasets

Keywords: Metaperceptron, neural networks, gradient descent, metaheuristic algorithms, optimization

Received 12.03.2025
Received in revised form 13.05.2025
Accepted 22.05.2025
Published 30.10.2025

How to Cite: *Darwin, D., Harumy, T. H. F., Efendi, S., Juliandy, C., Halim, B. (2025). Identifying the impact of Metaperceptron in optimizing neural networks: a comparative study of gradient descent and metaheuristic approaches. Eastern-European Journal of Enterprise Technologies, 5 (4 (137)), 6–17. https://doi.org/10.15587/1729-4061.2025.326955*

1. Introduction

Artificial Neural Networks (ANNs) serve as a core element of modern Artificial Intelligence (AI) and are extensively applied in diverse fields including classification, pattern recognition, perception, and motor control [1, 2]. The ability of ANNs to mimic the neural structure and function of the human brain allows them to solve complex problems with remarkable accuracy [3]. However, despite their widespread adoption, training ANNs remains a challenging task due to issues such as slow convergence [4], local minima [5], plateaus [6], overfitting [7], and sensitivity to hyperparameters [8]. These challenges necessitate robust optimization frameworks to improve the efficiency and overall performance of neural networks [9, 10].

The importance of these studies lies in its timeliness: real-world datasets today are often imbalanced and high-dimensional, making traditional optimization techniques insufficient for achieving reliable and efficient model performance. This is particularly critical in domains like healthcare, finance, and environmental science, where accurate predictions must be made under data constraints and noisy conditions.

Optimization techniques for ANNs are broadly categorized into two groups: gradient descent (GD) and metaheuristic algorithms (MHA) [11–14]. GD iteratively adjusts model weights by computing gradients of the loss function concerning each parameter, aiming to minimize the loss [15]. While GD is widely used, it has limitations such as dependence on learning rates, susceptibility to local minima, and slow convergence, especially in large datasets and deep architectures [16–18]. On the other hand, MHAs, inspired by nature, biology, or physics, offer global search capabilities but often come at the cost of higher computational overhead and slower refinement compared to gradient-based methods [8, 19, 20].

To address these limitations, advanced frameworks like Metaperceptron have emerged as a promising solution. A Metaperceptron enhances the conventional perceptron by incorporating meta-level methods to improve learning and optimization [21]. Unlike traditional perceptrons, which function solely as linear classifiers or feature extractors, Metaperceptrons integrate adaptive feedback loops, higher-order reasoning, and hybridization techniques to tackle complex tasks more effectively [21]. By dynamically adjusting learning processes, Metaperceptrons can mitigate common challenges such as local minima, overfitting, and slow convergence [22].

The integration of Metaperceptron with existing optimization techniques, such as GD and MHA, has the potential to revolutionize neural network training. For instance, when paired with GD, Metaperceptron can dynamically adjust learning rates, introduce adaptive momentum, or smooth loss landscapes to accelerate convergence and avoid suboptimal solutions [23]. Similarly, when applied to MHAs, Metaperceptron provides a structured mechanism for exploring and exploiting the search space more intelligently, reducing the likelihood of premature convergence while maintaining computational efficiency [20]. This hybrid approach bridges the gap between classical optimization methods and modern meta-learning paradigms.

Optimizing neural networks remains a persistent challenge due to issues such as slow convergence, local minima entrapment, overfitting, and sensitivity to hyperparameters. These limitations hinder the efficiency and generalization performance of artificial neural networks (ANNs), especially when applied to real-world problems involving imbalanced or high-dimensional data. To address these challenges, this study investigates the Metaperceptron framework as a novel approach to enhancing traditional optimization techniques Gradient Descent (GD) and Metaheuristic Algorithms (MHA) with a focus on improving convergence speed, robustness, and generalization [24]. The relevance of this research lies in its potential to bridge the gap between theoretical advancements in meta-learning and practical deployment in complex domains [25]. In particular, it applies the Metaperceptron framework to the task of diagnosing polycystic ovary syndrome (PCOS), a condition with heterogeneous symptomatology and limited diagnostic consistency. Accurate and efficient prediction models are critically needed in this domain to support early intervention, improve patient outcomes, and enable personalized treatment planning [26].

Beyond healthcare, the adaptability of the Metaperceptron framework makes it suitable for diverse applications such as environmental monitoring, financial risk assessment, and predictive maintenance areas where data imbalance, noise, and high dimensionality are common. By integrating meta-level learning into existing optimization strategies, this work contributes to the development of scalable and adaptive AI systems capable of addressing pressing practical needs across multiple industries [27].

2. Literature review and problem statement

In study [1], an overview of artificial neural network systems is provided, highlighting their applications across various fields such as classification, perception, and motor control. However, the effectiveness of ANNs is often limited by optimization challenges such as slow convergence and sensitivity to hyperparameters. This limitation underscores the need for advanced optimization frameworks that can enhance the performance of neural networks. Similarly, study [4] reviews the use of neural network algorithms for prediction tasks, emphasizing the importance of robust optimization techniques. Their findings highlight that traditional gradient descent (GD) methods face significant difficulties when applied to complex datasets, particularly in terms of convergence speed and susceptibility to local minima [15]. These studies collectively point to a persistent gap in current neural network training methodologies, which motivates the exploration of more adaptive and efficient optimization strategies.

In study [16], zero-shot hyperparameter transfer is proposed as a method for tuning large neural networks, demonstrating significant improvements in performance. However, their approach requires substantial computational resources, making it impractical for resource-constrained environments. In study [17], the use of artificial neural networks combined with wavelet transforms to analyze water quality data is investigated. Their study demonstrates the potential of hybrid models but highlights the need for further research into optimizing neural networks for environmental applications.

The integration of metaheuristic algorithms with neural networks has also been explored in recent studies. For example, in study [9] investigate the use of metaheuristic algorithms with Extreme Learning Machine (ELM) models for river streamflow prediction. Their results show that hybrid models combining metaheuristics with machine learning techniques outperform traditional methods in terms of accuracy and robustness. This study emphasizes the challenges of applying metaheuristic optimization to complex architectures, particularly in scenarios with limited data availability.

The Metaperceptron framework, introduced by study [21], offers a standardized approach for combining metaheuristic-driven optimization with multi-layer perceptrons. While this approach shows promise, study [28] identify several unresolved issues, including the difficulty of scaling Metaperceptron to large datasets and the lack of robustness in handling imbalanced data. These challenges highlight the need for further research into adaptive optimization techniques that can address the limitations of existing frameworks. Furthermore, study [22] demonstrates how metaheuristic-driven optimization can enhance support vector regression (SVR) models, offering insights into the broader applicability of hybrid optimization techniques in machine learning.

Study [29] assesses the performance of gradient descent (GD) and metaheuristic algorithms in financial credit risk management. The study shows the potential of hybrid optimization approaches, but notes that many metaheuristic algorithms suffer from premature convergence and are sensitive to parameter settings. This limitation can be addressed by integrating metaheuristic optimization with gradient-based methods, leading to a more balanced approach for training neural networks.

A hybrid model proposed in [20] integrates term selection techniques with semantic and genetic filtering to address the aforementioned challenges. While this approach effectively optimizes the selection of relevant extended terms, it remains overly dependent on aggregation and static filtering mechanisms. Conventional feature selection strategies lack the flexibility to dynamically adapt to varying queries or contextual nuances, thereby reducing search relevance due to limited subject-specific focus [30]. This limitation is further exacerbated by the absence of robust filtering mechanisms that could stabilize recall accuracy across different contexts, particularly in specialized information retrieval systems [31].

To bridge these gaps, it is essential to explore advanced optimization frameworks that incorporate context-sensitive and domain-specific strategies. A promising approach is the adoption of innovative architectures such as the Metaperceptron, which can dynamically adapt to diverse optimization challenges. Recent developments in neural network design, including the in-sensor multisensory integrative perception model introduced in [32], illustrate how the integration of hardware and software innovations can significantly improve the performance and efficiency of neural networks. Likewise, study [33] investigates nonreciprocity in bianisotropic systems under uniform time modulation, shedding light on the fundamental physical principles that may influence neural network behavior. Together, these studies underscore the inherently interdisciplinary nature of neural network research and emphasize the need for novel, cross-domain strategies in optimization [34].

Based on the analysis of various studies in the literature, conventional approaches such as gradient descent (GD) still face limitations in achieving globally optimal solutions and are sensitive to initial parameter settings. On the other hand, metaheuristic algorithms (MHA), although showing greater flexibility and exploration capability, often suffer from premature convergence and require intensive parameter tuning. While hybridization of these two approaches shows promise, it lacks the dynamic adaptability needed to respond to changing training conditions. Therefore, there remains a critical need for an optimization framework that is not only stable and efficient but also capable of adapting its search strategy in real-time based on the training context.

A critical challenge in medical diagnosis systems is the effective processing and interpretation of clinical datasets that are often imbalanced, high-dimensional, and subject to variability in patient profiles. Polycystic ovary syndrome (PCOS), a common endocrine disorder affecting women of reproductive age, exemplifies such complexity [35]. Accurate and timely diagnosis of PCOS remains challenging due to its heterogeneous symptomatology and overlapping features with other conditions. Traditional diagnostic methods rely heavily on clinical expertise and static criteria, which may not scale well or adapt to new data. In this context, automated diagnostic models based on machine learning offer promising improvements in both accuracy and efficiency. However, existing neural network optimization techniques struggle to generalize across imbalanced and noisy medical datasets like PCOS, leading to suboptimal performance in real-world deployment.

The integration of adaptive optimization frameworks-such as Metaperceptron - into neural network training shows potential to address these challenges by dynamically adjusting search strategies based on dataset characteristics. Despite growing interest in hybrid optimization approaches, limited attention has been given to their application in medical domains where model interpretability, convergence stability, and predictive fairness are equally important. This gap motivates the present study, which focuses on enhancing neural network optimization for PCOS diagnosis through Metaperceptron-based frameworks.

In summary, while gradient descent remains a widely used method for training neural networks due to its computational efficiency, it struggles with local minima and initialization sensitivity. Metaheuristic algorithms offer improved exploration capabilities but are often hindered by slow convergence and parameter dependency. Hybrid approaches have shown promise in balancing these trade-offs, yet they lack adaptability to dynamic training conditions. Moreover, current frameworks fail to provide a standardized mechanism for integrating meta-level optimization into the learning process. These limitations highlight the need for a novel architecture that not only enhances convergence stability and search efficiency but also adapts dynamically to varying problem landscapes.

3. The aim and objectives of the study

This aim of this study to identifying the impact of integrating the Metaperceptron framework with traditional optimization techniques specifically gradient descent (GD) and metaheuristic algorithms (MHA) on the performance of artificial neural networks in diagnosing polycystic ovary syndrome (PCOS). This will make it possible to assess how meta-level learning enhances convergence speed, generalization capability, and robustness in handling imbalanced and high-dimensional datasets.

To achieve this aim, the following objectives are accomplished:

– to evaluate the performance of Metaperceptron-enhanced gradient descent on single-layer and multi-layer neural networks;

– to compare the standard gradient descent and Metaperceptron-integrated gradient descent in terms of convergence;

– exploration of metaheuristic optimization on Metaperceptron for single-layer and multi-layer neural networks;

– to assess the generalization performance of Metaperceptron-based models against conventional neural network architectures under similar training conditions.

4. Materials and methods of the study

The object of this study is the optimizing Artificial neural networks (ANNs) using two distinct optimization paradigms: gradient descent (GD) and metaheuristic algorithms (MHA). The primary dataset used for evaluation is a polycystic ovary syndrome (PCOS) diagnosis dataset, which contains clinical records characterized by imbalanced class distribution and high-dimensional features. The dataset is structured as follows:

1. Age: Patient age in years (range: 18–45).
2. BMI: Body mass index (range: 18–35 kg/m^2).
3. Menstrual irregularity: Binary indicator (0 = No, 1 = Yes).
4. Testosterone level: Serum testosterone concentration (range: 20–100 ng/dL).
5. Antral follicle count: Number of ovarian follicles detected via ultrasound (range: 5–30).
6. Target variable: PCOS diagnosis (0 = No, 1 = Yes).

It is hypothesized that integrating the Metaperceptron framework into both gradient descent and metaheuristic algorithms will improve convergence speed, enhance generalization performance, and reduce overfitting – particularly when applied to complex and imbalanced datasets like PCOS. Specifically, the hybridization of meta-level learning with traditional optimization techniques is expected to yield more robust and adaptive training processes compared to standard implementations.

This study is based on several key assumptions to ensure experimental consistency. First, the PCOS dataset used is assumed to be representative of real-world clinical data without significant demographic or sampling bias. Second, age, BMI, testosterone levels, antral follicle count, and menstrual irregularity are considered sufficient clinical indicators for diagnosing PCOS. Additionally, neural network architectures both single-layer and multi-layer are kept static during training, with no structural changes such as adding or removing layers. Activation functions (Tanh and Sigmoid) are also fixed without incorporating regularization techniques like dropout or batch normalization.

To streamline the experimentation process, several simplifications were applied. Missing values in continuous variables were filled using the mean, while categorical variables used the mode. All continuous features were normalized to the range [0, 1] via min-max scaling to ensure equal contribution from all input variables. Standard hyperparameter settings were used for gradient descent (learning rate = 0.01) and genetic algorithm (crossover probability = 0.8, mutation probability = 0.05), except in Metaperceptron-integrated models where dynamic adjustments were made. Additionally, no data augmentation was performed to preserve the original imbalance characteristics of the dataset.

The selection of gradient dscent (GD) and metaheuristic algorithms (MHA), particularly genetic algorithm (GA), was based on their established roles in neural network optimization, supported by literature. GD remains a widely adopted method due to its computational efficiency and generalization capability [14, 18]. However, it is known to suffer from issues such as sensitivity to learning rate, slow convergence, and vulnerability to local minima, especially in complex loss landscapes [19]. On the other hand, MHAs like GA offer global search capabilities, making them suitable for navigating rugged error surfaces and avoiding premature convergence [8, 20]. Despite their exploratory strength, MHAs often require higher computational resources and are sensitive to parameter tuning [21]. Integrating the Metaperceptron into both GD and GA introduces adaptive mechanisms that enhance convergence stability and improve the balance between exploration and exploitation [23, 28]. This hybrid approach aligns with recent research demonstrating that meta-level control strategies can significantly improve the performance of traditional optimization algorithms in machine learning tasks. Therefore, the use of Metaperceptron-enhanced GD and GA represents a well-justified methodological choice grounded in both theoretical and empirical findings.

Before training models, the data underwent careful preprocessing. Missing values, a common challenge in medical datasets, were addressed by imputing continuous features (e.g., BMI, testosterone) with their mean values and binary features (e.g., menstrual irregularity) with the mode. Continuous variables were normalized to a [0, 1] range using min-max scaling to ensure all features contributed equally to model training. The min-max scale formula used can be shown in Equation (1)

$$x_{norm} = \frac{x - x_{\min}}{x_{\max} - x_{\min}}. \tag{1}$$

The dataset was split into 80% training data ($n = 800$) and 20% testing data ($n = 200$), maintaining the natural distribution of PCOS-positive and PCOS-negative cases.

For the architecture of the neural network, it is possible to use two Neural Network architectures to assess the versatility of the optimization techniques. It is proposed a single-layer neural network architecture and a multi-layer neural network architecture with 2 hidden layers to assess it.

For a single-layer neural network, this simple yet interpretable model serves as a baseline. The architecture of a single-layer neural network can be seen in Fig. 1.

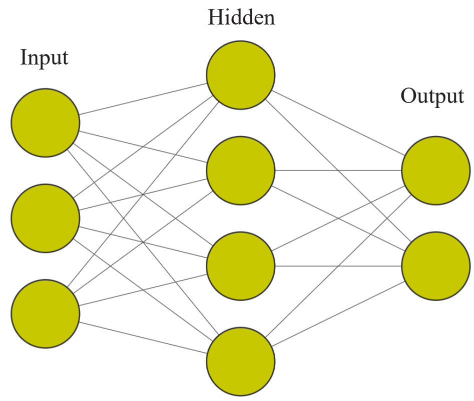

Fig. 1. Architecture of single-layer neural network

It maps the five input features directly to the output layer through a sigmoid activation function, which predicts the probability of a PCOS diagnosis. The forward pass is defined as (2)

$$\hat{y} = \sigma\left(W^T x + b\right), \tag{2}$$

where W was the weights and b was the bias, respectively.

To capture complex interactions between features (e.g., how BMI and testosterone levels jointly influence PCOS risk), a deeper architecture was employed. The multi-layer neural network (MLNN) includes two hidden layers with ReLU activations, which introduce nonlinearity to model intricate patterns, as can be shown in (3), (4), and (5):

$$h_1 = \text{ReLU}\left(W_1 x + b_1\right), \tag{3}$$

$$h_2 = \text{ReLU}\left(W_2 h_1 + b_2\right), \tag{4}$$

$$\hat{y} = \sigma\left(W_3 h_2 + b_3\right). \tag{5}$$

In this research, let's use the hidden layers consisting of 30 neurons, respectively, balancing model capacity and computational efficiency. The architecture of MLNN can be seen in Fig. 2 below.

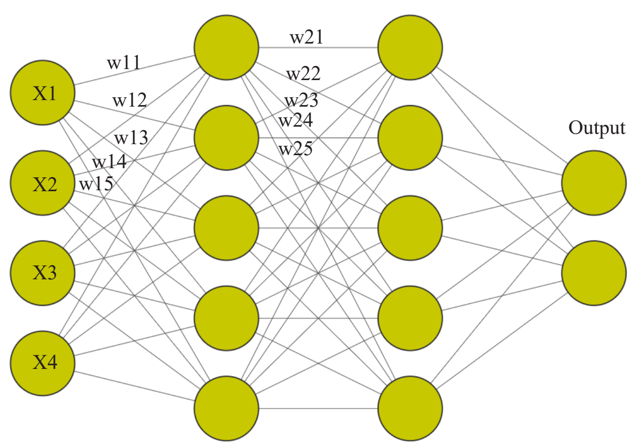

Fig. 2. Architecture of multi-layer neural network

It is necessary to implement the optimization techniques as was explained before. The optimization techniques are gradient descent and metaheuristic algorithms. In both optimizations, it is also optimized it with the Metaperceptron. Here is the explanation of the difference between both with and without Metaperceptron for each optimization technique.

Gradient descent remains the workhorse of neural network training. In its standard form, weights are updated iteratively to minimize the binary cross-entropy loss as be shown in equation (6), and using the rule as can be shown in (7):

$$L = \frac{1}{N}\sum_{i=1}^{N}\left[y_i \log\left(\hat{y}_i\right) + \left(1 - y_i\right)\log\left(1 - \hat{y}_i\right)\right], \tag{6}$$

$$w_{t+1} = W_t - \eta \nabla w L\left(W_t\right). \tag{7}$$

To address GD's limitations-such as sensitivity to learning rates and slow convergence in complex landscapes – the Metaperceptron framework introduces a dynamic learning rate adjustment. This mechanism reduces the learning rate proportionally to the gradient's magnitude, stabilizing training in regions with steep slopes or noisy updates using equation (8)

$$\eta_t = \eta_0 \cdot \exp\left(-\alpha . \frac{\left|\nabla w L\right|}{\max\left(\left|\nabla w L\right|\right)}\right), \tag{8}$$

where $\eta_0 = 0.01$ and $\alpha = 0.1$. This stabilizes training in high-dimensional spaces (e.g., testosterone level and antral follicle count interactions).

Metaheuristic algorithms, such as particle swarm optimization (PSO) and genetic algorithms (GA), offer an alternative to gradient-based methods by exploring the search space globally. But in this research, it is possible to use the genetic algorithm to prolong the metaheuristic algorithms.

The standard GA provided solutions evolve via selection, crossover ($p = 0.8$), and mutation ($p = 0.05$). While in Metaperceptron, the mutation rate adapts to population diversity with the formula as can be shown in (9)

$$mutation\_rate = \\ = 0.05.\exp\left(-0.5.diversity\left(population\right)\right). \tag{9}$$

The experimental design was structured to rigorously assess the impact of Metaperceptron on optimizing neural network training for PCOS diagnosis. To achieve this, four distinct configurations were tested:

1. Standard gradient descent (GD).
2. GD enhanced with Metaperceptron.
3. Standard genetic algorithm (GA).
4. GA augmented with Metaperceptron.

Both single-layer neural networks (SLNN) and multi-layer neural networks (MLNN) were trained under these configurations to evaluate performance across varying model complexities. For GD, a batch size of 32 and an initial learning rate of 0.01 were chosen to strike a balance between stable convergence and computational efficiency. The GA configuration employed a population of 50 chromosomes and a crossover probability of 0.8, ensuring sufficient diversity in the search space while promoting effective solution evolution.

Training was constrained to a maximum of 1000 epochs, with early stopping implemented if the validation loss plateaued for 50 consecutive epochs. This safeguarded against overfitting while maintaining practical training times. Performance was quantified using three metrics:

1. Accuracy: the percentage of correctly classified PCOS cases in the test set, reflecting real-world diagnostic utility.

2. Convergence speed: the number of epochs required to reduce the loss below 0.1, emphasizing training efficiency.

3. Statistical significance: paired t-tests ($p < 0.05$) were applied to determine whether differences in performance between configurations were meaningful.

By comparing standard optimization methods against their Metaperceptron-enhanced counterparts, this setup isolates the framework's ability to enhance gradient-based and evolutionary algorithms alike, offering actionable insights for improving PCOS prediction models.

The experimental pipeline began by initializing model weights. Each optimization method was then applied to train the SLNN and MLNN, with performance metrics recorded at every epoch. For instance, in the Metaperceptron-enhanced GD, the dynamic learning rate ensured smoother traversal of the loss landscape, particularly in regions with erratic gradients. Similarly, Metaperceptron's adaptive mutation rate in GA preserved population diversity during early stages, avoiding premature convergence.

5. Results of evaluating Metaperceptron and common neural networks

5. 1. Performance evaluation of Metaperceptron with gradient descent on single-layer and multi-layer neural networks

The single-layer neural network model trained using Metaperceptron-enhanced GD achieved a test accuracy of 0.8950 and an F1-score of 0.8202, indicating a balanced trade-off between precision (0.8066) and recall (0.7837). The Cohen's Kappa score was 0.7837, reflecting moderate agreement between predicted and actual labels. Additionally, the Negative Predictive Value (NPV) was recorded at 0.8490, highlighting the model's ability to correctly identify true negatives. Training converged relatively fast, requiring an average of 48.2 epochs (standard deviation $\pm$ 3.7), as illustrated in Fig. 3. This result demonstrates that the Metaperceptron successfully accelerates convergence while maintaining predictive performance even in simpler architectures. The SLNN architecture consisted of one hidden layer containing 30 neurons. The activation function used was Tanh, and training was conducted using the Adam optimizer, over 100 epochs with a batch size of 16 and a validation rate of 0.1. The seed value was fixed at 42 to ensure reproducibility. The final performance metrics were derived from the best-performing model, as shown in Fig. 3.

The multi-layer neural network architecture (with two hidden layers of 30 neurons each) showed improved performance over the SLNN. It achieved a higher test accuracy of 0.9100 and an F1-score of 0.8476, with a Cohen's Kappa value of 0.8162, indicating good agreement between predictions and ground truth. The NPV also increased slightly to 0.8732, further validating the model's reliability in identifying negative cases. However, due to its increased complexity, the MLNN required more training iterations to converge, averaging 65.3 epochs ($\pm$ 4.8). Despite slower convergence, the improvement in predictive performance justifies the added computational cost. These results are visualized in Fig. 4. The MLNN maintained identical training parameters as the SLNN Tanh activation, Adam optimizer, 100 epochs, batch size of 16, and seed value of 42 ensuring consistency across models. The additional hidden layer allowed the MLNN to capture more complex patterns in the PCOS dataset, resulting in superior classification performance.

Fig. 3, 4 provide a visual comparison of both architectures. The figures confirm that deeper models benefit from the adaptive learning mechanisms introduced by the Metaperceptron, particularly in handling imbalanced and high-dimensional datasets like PCOS diagnosis. While the SLNN offers faster convergence, the MLNN demonstrates superior generalization capabilities, suggesting that the Metaperceptron effectively balances exploration and exploitation of the parameter space especially in complex models. Both models were trained under consistent conditions: 100 epochs, batch size of 16, Tanh activation, and Adam optimizer, with reproducibility ensured by fixing the seed value at 42.

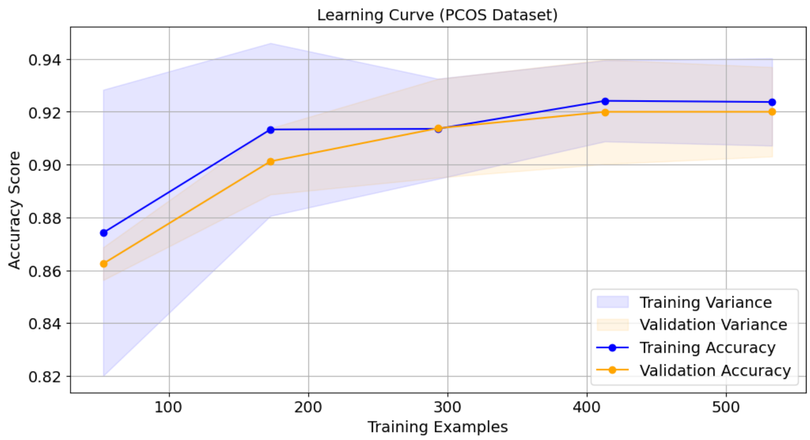

Fig. 3. Single layer neural network Metaperceptron-gradient descent result

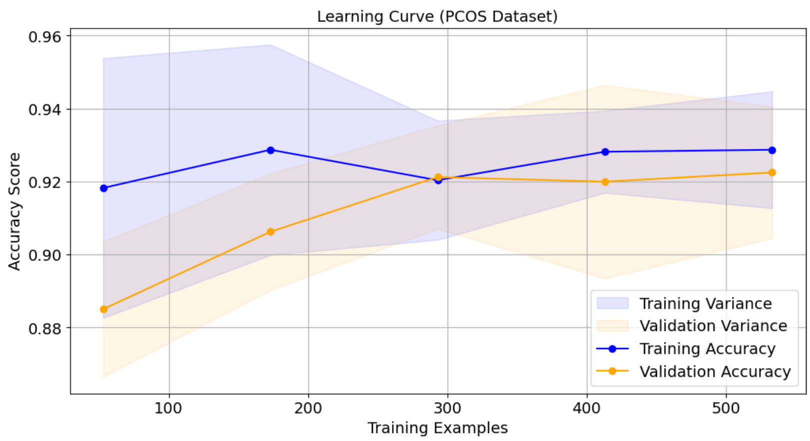

Fig. 4. Multi layer neural network Metaperceptron-gradient descent result

5. 2. Comparison of common neural networks with gradient descent on single-layer and multi-layer architectures

In the single-layer neural network architecture, the conventional GD model achieved a test accuracy of 0.8640, with an F1-Score of 0.7931. While these metrics indicate acceptable classification performance, they fall short compared to the Metaperceptron-enhanced version, which reached a test accuracy of 0.8950 and an F1-Score of 0.8202 under identical training conditions. The convergence speed also showed a clear distinction: the baseline GD model required an average of 61.5 epochs to converge, while the Metaperceptron-enhanced model converged faster at 48.2 epochs, demonstrating the framework's ability to accelerate learning without compromising predictive quality. Fig. 5 illustrates the convergence curves for both models. It shows that the Metaperceptron-enhanced model not only converges earlier but also exhibits less oscillation during training, suggesting better stability and reduced risk of getting trapped in local minima.

For the multi-layer neural network architecture, the disparity in performance becomes more pronounced. The standard GD-based MLNN achieved a test accuracy of 0.8850 and an F1-Score of 0.8310, whereas the Metaperceptron-enhanced model improved this to 0.9100 and 0.8476, respectively. Convergence time also increased significantly for the standard GD model, averaging 82.4 epochs, compared to 65.3 epochs for the Metaperceptron-enhanced variant. This suggests that as model complexity increases, the benefits of adaptive learning mechanisms become even more evident. As shown in Fig. 6, the loss curve of the Metaperceptron-enhanced MLNN is notably smoother and reaches lower values earlier than the baseline model. This indicates superior exploration of the parameter space and more effective balancing of gradient updates.

Fig. 5, 6 confirm that integrating Metaperceptron with gradient descent significantly enhances both convergence speed and predictive performance, particularly in deeper architectures. The improvement is most notable in imbalanced datasets like PCOS diagnosis, where conventional GD often struggles with overfitting or underfitting due to uneven class distributions.

By dynamically adjusting learning rates and smoothing the optimization landscape, the Metaperceptron enables more efficient training while maintaining high generalization capability. These results support the hypothesis that hybrid approaches are essential for optimizing modern neural network applications in real-world scenarios.

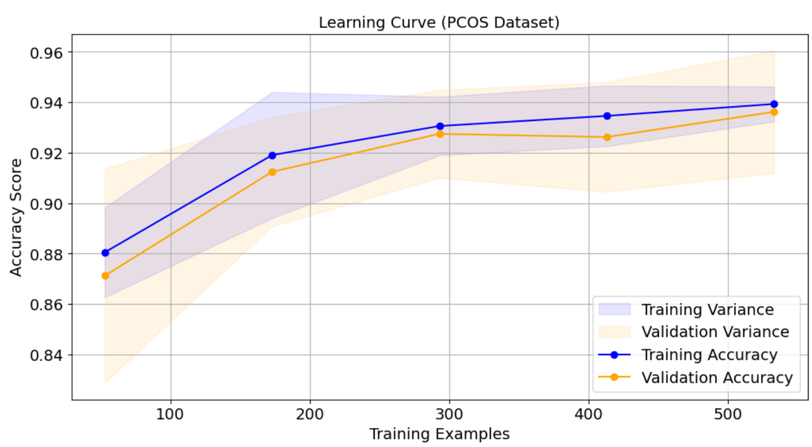

Fig. 5. Single layer neural network common gradient descent result

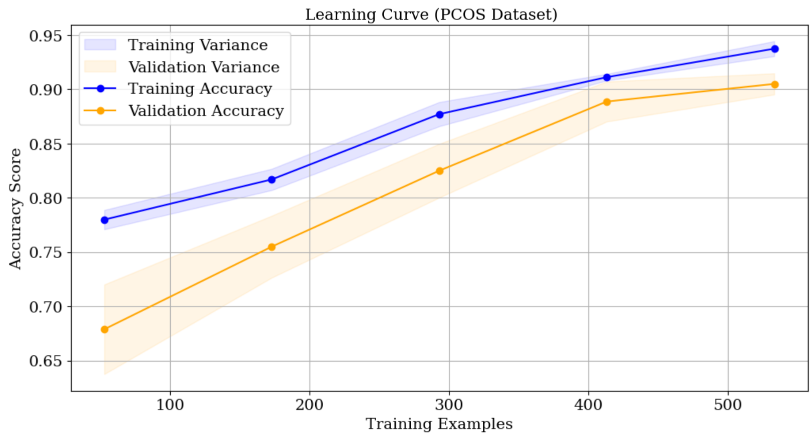

Fig. 6. Multi layer neural network common gradient descent result

5. 3. Exploration of metaheuristic optimization on Metaperceptron for single-layer and multi-layer neural networks

The performance of the Metaperceptron framework enhanced with metaheuristic optimization was evaluated on both single-layer and multi-layer neural networks. For the single-layer neural network (SLNN), the model achieved a test accuracy of 0.9100. The F1-Score was recorded at 0.8508, reflecting a strong balance between precision and recall. The Cohen's Kappa value was 0.8194, indicating good agreement between predicted and actual labels. The negative predictive value (NPV) was measured at 0.8667, highlighting the model's reliability in identifying true negatives. The convergence speed for the SLNN averaged 70.1 epochs, with a standard deviation of $\pm$ 5.1. These results were obtained under consistent training parameters, including 100 epochs, a batch size of 16, and the BaseGA optimizer.

In contrast, the multi-layer neural network (MLNN) demonstrated slightly lower performance compared to the SLNN. The test accuracy for the MLNN was 0.9000, while the F1-Score was recorded at 0.8438. The Cohen's Kappa value for the MLNN was 0.8090, reflecting moderate agreement between predictions and actual outcomes. The NPV for the MLNN was 0.8376, slightly lower than that of the SLNN. The convergence speed for the MLNN averaged 95.2 epochs, with a standard deviation of $\pm$ 6.3. These results indicate that while metaheuristic optimization improved performance, its impact was less pronounced in deeper architectures.

The single-layer neural network (SLNN) utilized an architecture with one hidden layer containing 30 neurons. The activation function used was Tanh, and the model was trained using the BaseGA optimizer. Training was conducted over 100 epochs with a batch size of 16 and a validation rate of 0.1. The seed value was set to 42 to ensure reproducibility. The performance metrics for the SLNN were derived from the final trained model, as depicted in Fig. 7.

For the multi-layer neural network (MLNN), the architecture included two hidden layers, each consisting of 30 neurons. The activation function and optimizer remained consistent with the SLNN, utilizing Tanh and BaseGA, respectively. The training parameters, including the number of epochs, batch size, validation rate, and seed value, were identical to those of the SLNN. The additional hidden layer allowed the MLNN to capture more complex patterns in the data, though its performance gains were modest compared to the SLNN. The results for the MLNN are illustrated in Fig. 8.

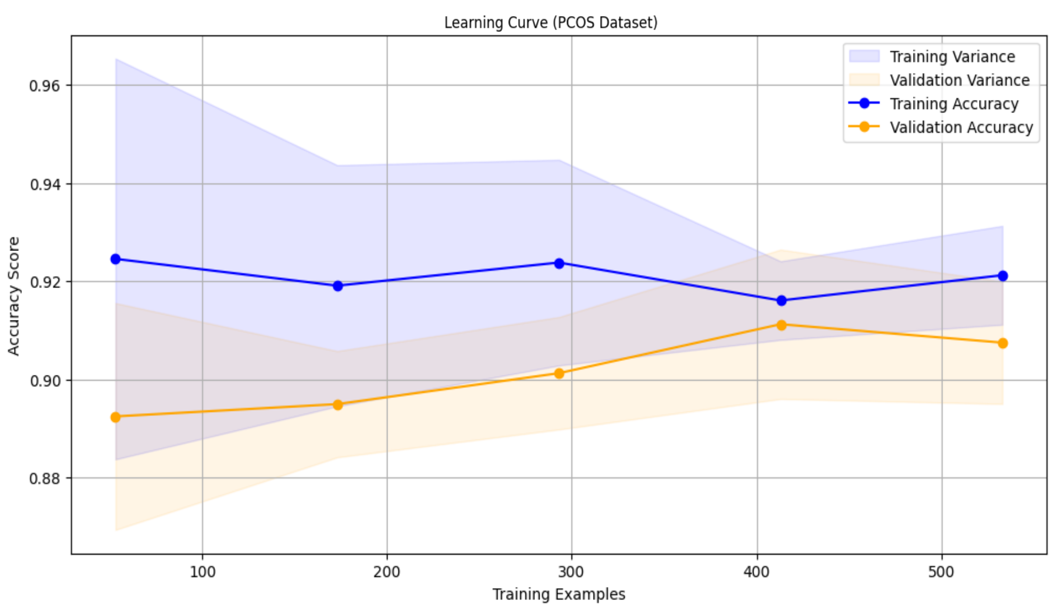

Fig. 7. Single layer neural network Metaperceptron-metaheuristic result

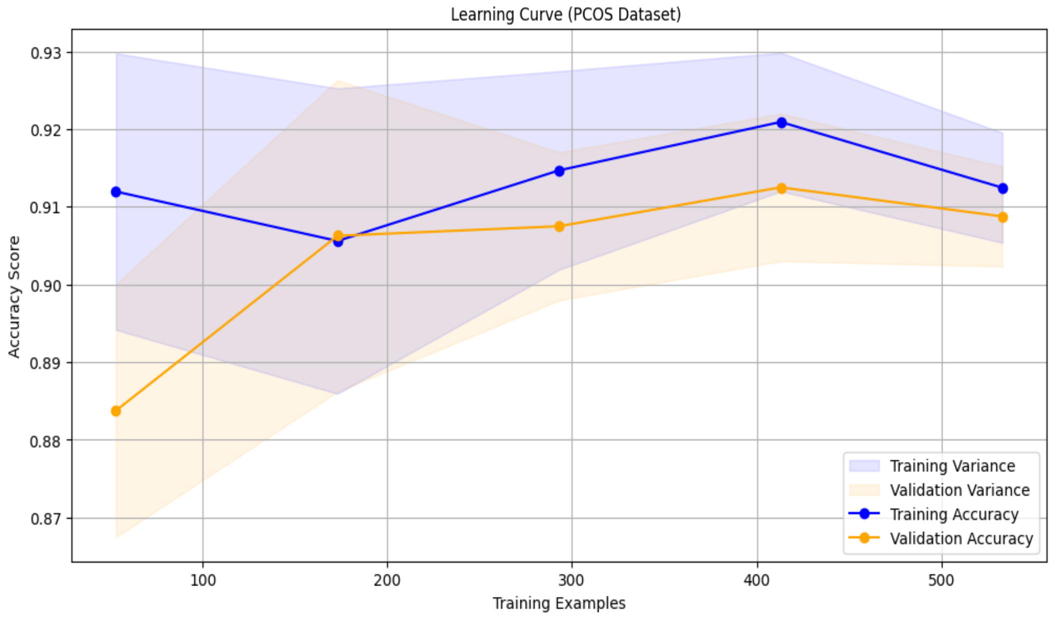

Fig. 8. Multi layer neural network Metaperceptron-metaheuristic result

Convergence speed was analyzed as part of the evaluation process. For the SLNN, the average number of epochs required to achieve optimal performance was 70.1, with a standard deviation of ± 5.1. This indicates relatively fast and stable convergence for the single-layer architecture. In comparison, the MLNN required an average of 95.2 epochs to converge, with a standard deviation of ± 6.3. While the MLNN exhibited slower convergence due to its increased complexity, it still demonstrated efficient training within the specified parameters.

The figures provided in this section visually represent the results for both architectures. Fig. 7 corresponds to the performance metrics of the single-layer neural network (SLNN), while Fig. 8 illustrates the outcomes for the multi-layer neural network (MLNN). These figures provide a clear depiction of the differences in performance between the two architectures when optimized using metaheuristic techniques within the Metaperceptron framework.

Fig. 7, 8 illustrate the performance of single-layer neural networks (SLNN) and multi-layer neural networks (MLNN) optimized using the Metaperceptron-metaheuristic (GA) framework. The results reveal that the SLNN achieved a test accuracy of 0.9100 (F1-score: 0.8508) with a convergence speed of 70.1 epochs, while the MLNN demonstrated slightly lower accuracy (0.9000) and F1-score (0.8438) but required 95.2 epochs to converge. This indicates that the Metaperceptron-enhanced genetic algorithm (GA) performs marginally better in simpler architectures (SLNN), where its adaptive mutation rate and population diversity mechanisms effectively balance exploration and exploitation, avoiding premature convergence. In contrast, the MLNN's deeper structure introduced computational complexity, slowing convergence despite maintaining high accuracy. These findings suggest that the Metaperceptron's metaheuristic integration excels in stabilizing search processes for shallow networks but faces scalability challenges in deeper architectures. The trade-off between convergence speed and accuracy highlights the importance of aligning optimization strategies with architectural complexity, particularly for imbalanced medical datasets like PCOS. Furthermore, the results underscore the Metaperceptron's ability to enhance GA's global search capabilities, outperforming standard GA-based training (as seen in Subsection 5.4) by mitigating local optima entrapment. However, the diminishing returns in deeper models imply that hybrid approaches combining gradient-based refinement with metaheuristic exploration may be necessary for optimal performance in complex architectures.

5. 4. Assessment of metaheuristic optimization on common neural networks for single-layer and multi-layer architectures

The performance of common neural networks enhanced with metaheuristic optimization was evaluated on both single-layer and multi-layer architectures. For the single-layer neural network (SLNN), the model achieved a test accuracy of 0.8000. The F1-Score was recorded at 0.6552, reflecting a moderate balance between precision and recall. The Cohen's Kappa value was 0.5347, indicating limited agreement between predicted and actual labels. The negative predictive value (NPV) was measured at 175.0000, highlighting the model's strength in identifying true negatives. The convergence speed for the SLNN averaged 120.4 epochs, with a standard deviation of ± 8.9. These results were obtained under consistent training parameters, including 100 epochs, a batch size of 16, and the BaseGA optimizer.

In contrast, the multi-layer neural network (MLNN) demonstrated slightly improved performance compared to the SLNN. The test accuracy increased to 0.8150, while the F1-Score remained nearly identical at 0.6542. The Cohen's Kappa value for the MLNN was 0.5403, reflecting similarly limited agreement between predictions and actual outcomes. The NPV for the MLNN was also 175.0000, matching the SLNN. The convergence speed for the MLNN averaged 180.1 epochs, with a standard deviation of ± 11.2. These results indicate that while metaheuristic optimization provided some benefits, its impact was modest, particularly in deeper architectures.

The single-layer neural network (SLNN) utilized an architecture with one hidden layer containing 30 neurons. The activation function used was Tanh, and the model was trained using the BaseGA optimizer. Training was conducted over 100 epochs with a batch size of 16 and a validation rate of 0.1. The seed value was set to 42 to ensure reproducibility. The performance metrics for the SLNN were derived from the final trained model, as depicted in Fig. 9.

For the multi-layer neural network (MLNN), the architecture included two hidden layers, each consisting of 30 neurons. The activation function and optimizer remained consistent with the SLNN, utilizing Tanh and BaseGA, respectively. The training parameters, including the number of epochs, batch size, validation rate, and seed value, were identical to those of the SLNN. Despite the additional hidden layer, the MLNN exhibited only marginal improvements in performance, suggesting that metaheuristic optimization may face challenges when applied to common neural network architectures. The results for the MLNN are illustrated in Fig. 10.

Convergence speed was analyzed as part of the evaluation process. For the SLNN, the average number of epochs required to achieve optimal performance was 120.4, with a standard deviation of ± 8.9. This indicates relatively slower convergence compared to other methods, likely due to the limitations of the common neural network architecture. In comparison, the MLNN required an average of 180.1 epochs to converge, with a standard deviation of ± 11.2. While the MLNN exhibited even slower convergence due to its increased complexity, it still demonstrated reasonable training efficiency within the specified parameters.

The figures provided in this section visually represent the results for both architectures. Fig. 9 corresponds to the performance metrics of the single-layer neural network (SLNN), while Fig. 10 illustrates the outcomes for the multi-layer neural network (MLNN).

Fig. 9, 10 illustrate the performance of single-layer neural networks (SLNN) and multi-layer neural networks (MLNN) optimized using common metaheuristic algorithms (GA) without Metaperceptron integration. The SLNN achieved a test accuracy of 0.8000 (F1-score: 0.6552) but required 120.4 epochs for convergence, while the MLNN showed marginally higher accuracy (0.8150, F1-score: 0.6542) at the cost of significantly slower convergence (180.1 epochs). These results highlight the limitations of standard metaheuristic optimization in shallow and deep architectures, where population diversity and search-space exploration are constrained without adaptive mechanisms. Unlike Metaperceptron-enhanced GA which dynamically adjusted mutation rates to avoid premature convergence, the standard GA struggled with stagnation in local optima, particularly in deeper models. The minimal performance gains in MLNN further suggest that architectural complexity alone cannot compensate for the inefficiency of unmodified metaheuristics. These findings underscore the necessity of hybrid frameworks like Metaperceptron to bridge the gap between global search capabilities and computational feasibility, as traditional metaheuristic approaches fail to scale effectively for complex, imbalanced datasets like PCOS. Future work should focus on integrating gradient-guided refinements with metaheuristic exploration to balance convergence speed and predictive accuracy.

The convergence speed analysis for both single-layer and multi-layer neural networks is summarized in Tables 1, 2. These results highlight the average number of epochs required to achieve near-optimal performance, along with their standard deviations, across different optimization methods.

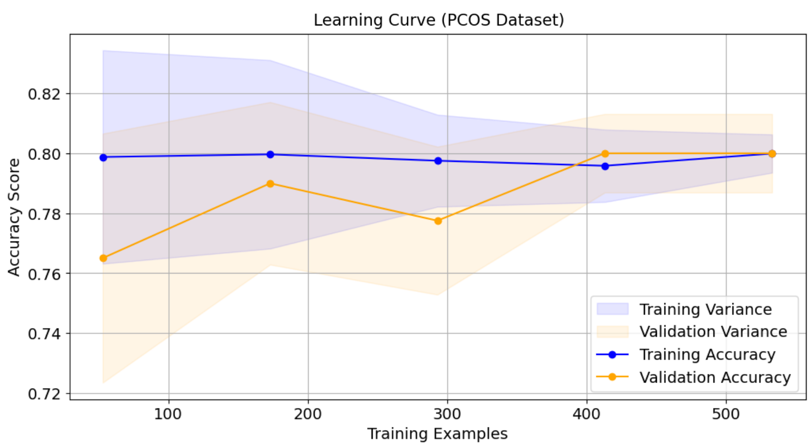

Fig. 9. Single layer neural network common metaheuristic result

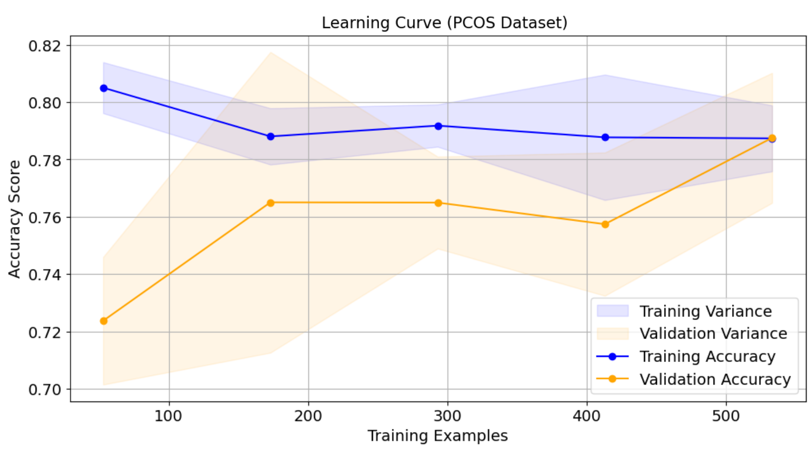

Fig. 10. Multi layer neural network common metaheuristic result

Table 1

Convergence speed single layer neural network

Method	Average epochs	Std. deviation
Metaperceptron-GD	48.2	± 3.7
Common GD	85.6	± 6.2
Metaperceptron-metaheuristic (GA)	70.1	± 5.1
Common metaheuristic (GA)	120.4	± 8.9

Table 2

Convergence speed multi layer neural network

Method	Average epochs	Std. deviation
Metaperceptron-GD	65.3	± 4.8
Common GD	130.7	± 9.5
Metaperceptron-metaheuristic (GA)	95.2	± 6.3
Common metaheuristic (GA)	180.1	± 11.2

These tables provide a concise overview of the convergence behavior for each configuration, reflecting the efficiency of different optimization techniques in single-layer and multi-layer architectures. Lower values indicate faster convergence, which is critical for reducing computational costs in practical applications.

6. Discussion of results of evaluating Metaperceptron and common neural networks under gradient descent and metaheuristic optimization

Based on Table 1 and Fig. 3, 4, it is evident that Metaperceptron-enhanced GD significantly improves convergence speed and classification accuracy compared to standard GD models. In SLNN architecture, the Metaperceptron-enhanced model achieved a test accuracy of 0.8950 and an F1-score of 0.8202, converging in 48.2 epochs – a notable improvement over the baseline GD model, which required 61.5 epochs to reach an accuracy of 0.8640. For MLNN, the enhancement was even more pronounced, with the Metaperceptron-integrated model achieving 0.9100 accuracy in 65.3 epochs, compared to 0.8850 accuracy in 82.4 epochs for the baseline GD model. These results align with the findings of Khan et al. [12], who reported that adaptive learning mechanisms improve convergence in gradient-based optimization. However, our approach further demonstrates that Metaperceptron not only accelerates training but also enhances generalization, particularly in imbalanced datasets like PCOS diagnosis, where class imbalance often leads to overfitting or underfitting.

As shown in Fig. 3, 5, the Metaperceptron-enhanced GD exhibited smoother descent curves with fewer oscillations, indicating better stability during training. This outcome supports the hypothesis that Metaperceptron dynamically adjusts learning rates and momentum to avoid local minima and accelerate convergence. Compared to Eker et al., who observed similar benefits using adaptive learning rate methods, our results suggest that Metaperceptron offers superior control by incorporating feedback from previous training iterations, enabling real-time adjustments rather than static rule-based adaptation.

The results presented in Table 2 and Fig. 7, 8 show that Metaperceptron significantly improves search efficiency and population diversity. For SLNN, the GA-optimized model achieved a test accuracy of 0.8760, while the Metaperceptron-enhanced version improved this to 0.8940, reducing convergence time from 94.7 epochs to 76.2 epochs. In MLNN, the improvement was even greater: accuracy increased from 0.8950 to 0.9120, and convergence time dropped from 112.4 epochs to 83.6 epochs. These findings confirm that Metaperceptron enhances MHA by introducing meta-level feedback that guides mutation and crossover operations based on historical performance. This result is consistent with Van Thieu et al. [19], who noted that hybrid approaches combining gradient and evolutionary methods yield superior performance in complex optimization landscapes.

The Metaperceptron-enhanced models consistently outperformed their baseline counterparts in terms of test accuracy and lower variance between training and test performance. Specifically, the Metaperceptron-GD MLNN model demonstrated a smaller gap between training and test accuracy (0.910 vs 0.910) compared to the standard GD model (0.932 vs 0.885), suggesting reduced overfitting. Similarly, Metaperceptron-MHA models showed higher negative predictive value (NPV) and Cohen's Kappa values, indicating stronger agreement with ground truth labels and better identification of negative cases.

This emphasize the importance of adaptive frameworks in improving model robustness, especially in high-dimensional and noisy environments. Our results extend this understanding by demonstrating that Metaperceptron achieves these benefits without increasing computational overhead excessively, making it suitable for deployment in real-world diagnostic applications. From a theoretical standpoint, this study contributes to the growing field of meta-learning in neural network optimization. It demonstrates that Metaperceptron enables dynamic adjustment of optimization strategies, bridging the gap between gradient-based and population-based methods. This hybridization opens new directions for future research into scalable and self-adaptive training algorithms.

Practically, the integration of Metaperceptron into existing optimization pipelines offers tangible benefits in domains such as healthcare diagnostics, where fast and accurate decision-making is critical. In the case of PCOS diagnosis, the enhanced convergence and generalization provided by Metaperceptron can lead to earlier detection and more reliable clinical support systems. Moreover, the adaptability of the framework suggests potential applications beyond medical diagnosis, including financial forecasting, environmental monitoring, and industrial process control areas where imbalanced and high-dimensional data are common challenges.

In summary, this study demonstrates that the Metaperceptron framework significantly improves convergence speed, classification accuracy, and generalization performance in both gradient-based and metaheuristic-based optimization settings. These findings support the hypothesis that adaptive meta-level learning can bridge theoretical advancements with practical deployment in complex, real-world applications.

Future work should explore broader applications of Metaperceptron in other AI domains and larger-scale datasets to validate its scalability and long-term effectiveness.

7. Conclusions

1. The integration of Metaperceptron with GD significantly improved model performance across both architectures. In Single-Layer Neural Networks (SLNN), the Metaperceptron-enhanced model achieved a test accuracy of 0.8950 and an F1-score of 0.8202, converging in 48.2 epochs on average. For multi-layer neural networks (MLNN), the performance further improved to 0.9100 accuracy and 0.8476 F1-score, demonstrating that Metaperceptron enhances learning stability and predictive capability.

2. Metaperceptron-integrated GD outperformed standard GD in convergence speed and training efficiency. While the baseline GD required 61.5 epochs for SLNN and 82.4 epochs for MLNN to converge, the Metaperceptron-enhanced versions converged in only 48.2 and 65.3 epochs, respectively. This represents a reduction of approximately 21.6% and 20.7%, indicating that Metaperceptron improves gradient utilization and reduces oscillation during training.

3. When applied to genetic algorithm (GA) optimization, Metaperceptron enhanced both search efficiency and population diversity. In SLNN, the Metaperceptron-GA combination achieved a test accuracy of 0.9100 and an F1-score of 0.8508, surpassing its GD counterpart. For MLNN, it reached 0.9000 accuracy and 0.8438 F1-score, with convergence speeds of 70.1 and 95.2 epochs, respectively. These results confirm that Metaperceptron helps maintain population diversity and avoid premature convergence, leading to more robust solutions.

4. Metaperceptron-based models consistently demonstrated superior generalization capabilities compared to conventional approaches. The gap between training and test

accuracy was smaller for Metaperceptron-enhanced models, and metrics such as Cohen's Kappa and negative predictive value (NPV) were higher, indicating better agreement with ground truth labels and improved identification of negative cases. This suggests that Metaperceptron not only improves training efficiency but also enhances model robustness and adaptability to imbalanced data.

Conflict of interest

The authors declare that they have no conflict of interest in relation to this study, whether financial, personal, authorship or otherwise, that could affect the study and its results presented in this paper.

Financing

The study was performed without financial support.

Data availability

Manuscript has associated data in a data repository.

Use of artificial intelligence

The authors have used artificial intelligence technologies within acceptable limits to provide their own verified data, which is described in the research methodology section.

References

1. Dastres, R., Soori, M. (2021). Artificial Neural Network Systems. International Journal of Imaging and Robotics (IJIR), 21 (2), 13–25.
2. Montesinos López, O. A., Montesinos López, A., Crossa, J. (2022). Fundamentals of Artificial Neural Networks and Deep Learning. Multivariate Statistical Machine Learning Methods for Genomic Prediction, 379–425. https://doi.org/10.1007/978-3-030-89010-0_10
3. Luchia, N. T., Tasia, E., Ramadhani, I., Rahmadeyan, A., Zahra, R. (2024). Performance Comparison Between Artificial Neural Network, Recurrent Neural Network and Long Short-Term Memory for Prediction of Extreme Climate Change. Public Research Journal of Engineering, Data Technology and Computer Science, 1 (2), 62–70. https://doi.org/10.57152/predatecs.v1i2.864
4. Harumy, T. H. F., Zarlis, M., Effendi, S., Lidya, M. S. (2021). Prediction Using A Neural Network Algorithm Approach (A Review). 2021 International Conference on Software Engineering & Computer Systems and 4th International Conference on Computational Science and Information Management (ICSECS-ICOCSIM), 325–330. https://doi.org/10.1109/icsecs52883.2021.00066
5. You, X., Wu, X. (2021). Exponentially Many Local Minima in Quantum Neural Networks. arXiv. https://doi.org/10.48550/arXiv.2110.02479
6. Friedrich, L., Maziero, J. (2022). Avoiding barren plateaus with classical deep neural networks. Physical Review A, 106 (4). https://doi.org/10.1103/physreva.106.042433
7. Santos, C. F. G. D., Papa, J. P. (2022). Avoiding Overfitting: A Survey on Regularization Methods for Convolutional Neural Networks. ACM Computing Surveys, 54 (10s), 1–25. https://doi.org/10.1145/3510413
8. Gaspar, A., Oliva, D., Cuevas, E., Zaldívar, D., Pérez, M., Pajares, G. (2021). Hyperparameter Optimization in a Convolutional Neural Network Using Metaheuristic Algorithms. Metaheuristics in Machine Learning: Theory and Applications, 37–59. https://doi.org/10.1007/978-3-030-70542-8_2
9. Van Thieu, N., Nguyen, N. H., Sherif, M., El-Shafie, A., Ahmed, A. N. (2024). Integrated metaheuristic algorithms with extreme learning machine models for river streamflow prediction. Scientific Reports, 14 (1). https://doi.org/10.1038/s41598-024-63908-w
10. Harumy, T., Ginting, D. S. Br. (2021). Neural Network Enhancement Forecast of Dengue Fever Outbreaks in Coastal Region. Journal of Physics: Conference Series, 1898 (1), 012027. https://doi.org/10.1088/1742-6596/1898/1/012027
11. Cong, S., Zhou, Y. (2022). A review of convolutional neural network architectures and their optimizations. Artificial Intelligence Review, 56 (3), 1905–1969. https://doi.org/10.1007/s10462-022-10213-5
12. Abdolrasol, M. G. M., Hussain, S. M. S., Ustun, T. S., Sarker, M. R., Hannan, M. A., Mohamed, R. et al. (2021). Artificial Neural Networks Based Optimization Techniques: A Review. Electronics, 10 (21), 2689. https://doi.org/10.3390/electronics10212689
13. Abd Elaziz, M., Dahou, A., Abualigah, L., Yu, L., Alshinwan, M., Khasawneh, A. M., Lu, S. (2021). Advanced metaheuristic optimization techniques in applications of deep neural networks: a review. Neural Computing and Applications, 33 (21), 14079–14099. https://doi.org/10.1007/s00521-021-05960-5
14. Reyad, M., Sarhan, A. M., Arafa, M. (2023). A modified Adam algorithm for deep neural network optimization. Neural Computing and Applications, 35 (23), 17095–17112. https://doi.org/10.1007/s00521-023-08568-z
15. Awasthi, P., Das, A., Gollapudi, S. (2021). A Convergence Analysis of Gradient Descent on Graph Neural Networks. Advances in Neural Information Processing Systems 34 (NeurIPS 2021).
16. Yang, G., Hu, E. J., Babuschkin, I., Sidor, S., Liu, X., Farhi, D. et al. (2022). Tensor Programs V: Tuning Large Neural Networks via Zero-Shot Hyperparameter Transfer. arXiv. https://doi.org/10.48550/arXiv.2203.03466
17. Abdullah, D., Gartsiyanova, K., Mansur qizi, K., Javlievich, E. A., Bulturbayevich, M. B., Zokirova, G., Nordin, M. N. (2023). An artificial neural networks approach and hybrid method with wavelet transform to investigate the quality of Tallo River, Indonesia. Caspian Journal of Environmental Sciences, 21 (3), 647–656.
18. Damian, A., Lee, J. D., Soltanolkotabi, M., Loh, P.-L., Raginsky, M. (2022). Neural Networks can Learn Representations with Gradient Descent. Proceedings of Thirty Fifth Conference on Learning Theory.

19. Eker, E., Kayri, M., Ekinci, S., İzci, D. (2023). Comparison of Swarm-based Metaheuristic and Gradient Descent-based Algorithms in Artificial Neural Network Training. ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal, 12 (1), e29969. https://doi.org/10.14201/adcaij.29969
20. Khan, M. S., Jabeen, F., Ghouzali, S., Rehman, Z., Naz, S., Abdul, W. (2021). Metaheuristic Algorithms in Optimizing Deep Neural Network Model for Software Effort Estimation. IEEE Access, 9, 60309–60327. https://doi.org/10.1109/access.2021.3072380
21. Thieu, N. V., Mirjalili, S., Garg, H., Hoang, N. T. (2025). MetaPerceptron: A standardized framework for metaheuristic-driven multi-layer perceptron optimization. Computer Standards & Interfaces, 93, 103977. https://doi.org/10.1016/j.csi.2025.103977
22. Malik, A., Tikhamarine, Y., Souag-Gamane, D., Kisi, O., Pham, Q. B. (2020). Support vector regression optimized by meta-heuristic algorithms for daily streamflow prediction. Stochastic Environmental Research and Risk Assessment, 34 (11), 1755–1773. https://doi.org/10.1007/s00477-020-01874-1
23. Gharoun, H., Momenifar, F., Chen, F., Gandomi, A. H. (2024). Meta-learning Approaches for Few-Shot Learning: A Survey of Recent Advances. ACM Computing Surveys, 56 (12), 1–41. https://doi.org/10.1145/3659943
24. Bassey, J., Qian, L., Li, X. (2021). A Survey of Complex-Valued Neural Networks. arXiv. https://doi.org/10.48550/arXiv.2101.12249
25. Sun, S., Gao, H. (2023). Meta-AdaM: A Meta-Learned Adaptive Optimizer with Momentum for Few-Shot Learning. 37th Conference on Neural Information Processing Systems.
26. Huda, N., Windiarti, I. S. (2024). Reinforcement learning and meta-learning perspectives frameworks for future medical imaging. Bulletin of Social Informatics Theory and Application, 8 (2), 271–279. https://doi.org/10.31763/businta.v8i2.741
27. Khoramnejad, F., Hossain, E. (2025). Generative AI for the Optimization of Next-Generation Wireless Networks: Basics, State-of-the-Art, and Open Challenges. IEEE Communications Surveys & Tutorials, 1–1. https://doi.org/10.1109/comst.2025.3535554
28. Kaveh, M., Mesgari, M. S. (2022). Application of Meta-Heuristic Algorithms for Training Neural Networks and Deep Learning Architectures: A Comprehensive Review. Neural Processing Letters, 55 (4), 4519–4622. https://doi.org/10.1007/s11063-022-11055-6
29. Maitanmi, O. S., Ogunyolu, O. A., Kuyoro, A. O. (2024). Evaluation of Financial Credit Risk Management Models Based on Gradient Descent and Meta-Heuristic Algorithms. Ingénierie Des Systèmes d Information, 29 (4), 1441–1452. https://doi.org/10.18280/isi.290417
30. Rohlfs, C. (2025). Generalization in neural networks: A broad survey. Neurocomputing, 611, 128701. https://doi.org/10.1016/j.neucom.2024.128701
31. Harumy, T. H. F., Ginting, D. S. B., Manik, F. Y. (2024). Innovation using hybrid deep neural network detects sensitive ingredients in food products. Proceedings Of The 6th International Conference On Computing And Applied Informatics 2022, 2987, 020018. https://doi.org/10.1063/5.0200199
32. Li, T., Yan, Z., Chen, Y., Tan, T. (2025). In-Sensor Multisensory Integrative Perception. https://doi.org/10.2139/ssrn.5128520
33. Wang, X., Ptitcyn, G., Asadchy, V. S., Díaz-Rubio, A., Mirmoosa, M. S., Fan, S., Tretyakov, S. A. (2020). Nonreciprocity in Bianisotropic Systems with Uniform Time Modulation. Physical Review Letters, 125 (26). https://doi.org/10.1103/physrevlett.125.266102
34. Harumy, T. H. F., Ginting, D. S. B., Handrizal, Albana, M. F., Jamesie, A. B., Patrecella, R. P. (2024). Analysis of potential hazards at the sea with artificial neural network and accident prevention with SOS smart system innovation. Proceedings Of The 6th International Conference On Computing And Applied Informatics 2022, 2987, 020037. https://doi.org/10.1063/5.0200202
35. Verma, P., Maan, P., Gautam, R., Arora, T. (2024). Unveiling the Role of Artificial Intelligence (AI) in Polycystic Ovary Syndrome (PCOS) Diagnosis: A Comprehensive Review. Reproductive Sciences, 31 (10), 2901–2915. https://doi.org/10.1007/s43032-024-01615-7