



Performance Enhancement of Adaptive Neuro-Fuzzy Inference System through Population-Based Algorithms

Mohamed Abderraouf Ferradji*

Artificial Intelligence Laboratory, Department of computer science, Faculty of Sciences, Ferhat Abbas University Setif-1, Setif, Algeria.

* Corresponding Author Email: mohamed.ferradji@univ-setif.dz

Article Info:

DOI: 10.22399/ijcesn.4145
Received : 03 September 2025
Accepted : 27 October 2025

Keywords

ANFIS
Metaheuristic Optimization
Parameter Tuning
Fuzzy Logic Systems

Abstract:

The Adaptive Neuro-Fuzzy Inference System (ANFIS) is a powerful hybrid artificial intelligence model that combines the learning capabilities of neural networks with the reasoning of fuzzy logic. It is widely used to model complex relationships between input and output parameters across various domains. However, the performance of ANFIS is highly dependent on the optimal setting of its internal parameters, making their optimization a significant research focus. This work aims to enhance ANFIS performance by optimizing its parameters using seven well-recognized metaheuristic algorithms, including Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Salp Swarm Algorithm (SSA), and Grey Wolf Optimizer (GWO). The proposed hybrid models were evaluated on three different datasets. Experimental results demonstrate that the hybrid models, which integrate these optimization algorithms with ANFIS, achieve a significant performance improvement compared to the standard ANFIS model.

1. Introduction

In recent years, artificial neural networks (ANNs) have become one of the most powerful supervised learning techniques, due to their remarkable ability to learn complex nonlinear relationships directly from data. Their flexible architecture enables them to approximate complex functions and achieve high predictive accuracy across a wide range of applications. However, ANNs mainly focus on numerical learning and do not naturally incorporate human reasoning or linguistic knowledge. In contrast, fuzzy logic systems can represent uncertainty and qualitative knowledge more intuitively through human-understandable rules, providing transparency and interpretability in decision-making processes [4].

The Adaptive Neuro-Fuzzy Inference System (ANFIS) bridges this gap by integrating the adaptive learning ability of neural networks with the interpretability of fuzzy logic principles. This hybrid architecture enables ANFIS to effectively model nonlinear systems while maintaining a transparent reasoning process through its fuzzy rule base [5]. As a supervised machine learning model, ANFIS learns from labeled datasets to accurately approximate nonlinear input–output relationships. However, its

performance heavily relies on the optimal configuration of parameters, including membership functions and consequent parameters. The standard gradient-based learning approach often suffers from limitations such as convergence to local minima and sensitivity to initial conditions, which highlights the necessity for more robust optimization strategies [6].

Recent advances in metaheuristic optimization have provided promising solutions to these challenges. Algorithms such as Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Grey Wolf Optimizer (GWO), and Salp Swarm Algorithm (SSA) have demonstrated exceptional capability in solving complex optimization problems. These population-based algorithms offer global search capabilities that can effectively navigate the parameter space of ANFIS models, potentially overcoming the limitations of traditional training methods [7]. This paper presents a comprehensive investigation into the optimization of ANFIS parameters using seven metaheuristic algorithms: PSO, GA, GWO, and SSA, SCSO, FOX and GGO. Our study aims to systematically evaluate the performance of these hybrid approaches across multiple benchmark datasets. The experimental results demonstrate significant improvements in

prediction accuracy and model efficiency compared to the conventional ANFIS approach, demonstrating the potential of metaheuristic algorithms for complex modeling tasks.

2. Adaptive Neuro-Fuzzy Inference System (ANFIS) Architecture

The Adaptive Neuro-Fuzzy Inference System (ANFIS) architecture is fundamentally based on a set of fuzzy "if-then" rules enhanced with learning mechanisms to approximate complex nonlinear functions [8]. The synergy between neural networks and fuzzy systems has proven highly effective across numerous research domains, demonstrating particular strength in modeling and control applications where interpretability and learning are simultaneously required. Structurally, ANFIS is implemented as an adaptive network consisting of interconnected nodes and directed links, where each node processes incoming signals according to its specific function. The parameters that determine these functions are modifiable and are categorized as premise parameters (related to input membership functions) and consequent parameters (related to output functions) [9]. This adaptive nature allows the system to refine its knowledge representation through learning from data. The mathematical foundation of ANFIS closely follows the Takagi-Sugeno-Kang (TSK) fuzzy model, employing a hybrid learning mechanism that combines gradient descent and least squares estimation. This combination significantly enhances parameter optimization efficiency. During the forward pass, consequent parameters are identified using the least squares method, while premise parameters are updated through gradient descent in the backward pass. This hybrid approach accelerates convergence by effectively reducing the search space dimensionality during error backpropagation [8, 9, 10].

2.1 Network Architecture and Layer Description

The ANFIS architecture consists of five distinct layers, each performing a specific computational role. Figure 1 illustrates the complete network structure with two inputs (x, y) and a single output (f), implementing two fuzzy if-then rules according to the Sugeno fuzzy inference model.

2.1.1 Layer 1 (Fuzzification Layer):

Each node in this layer corresponds to a linguistic label and generates membership grades for input variables. The node functions are defined as:

$$O_{1,i} = \mu_{A_i}(x), \quad i = 1, 2 \quad (1)$$

$$O_{1,i} = \mu_{B_{i-2}}(y), \quad i = 3, 4 \quad (2)$$

where $\mu_{A_i}(x)$ and $\mu_{B_{i-2}}(y)$ represent membership functions, typically using bell-shaped functions parameterized by $\{a_i, b_i, c_i\}$ (premise parameters).

2.1.2 Layer 2 (Rule Layer):

Every node in this layer is a fixed node labeled Π that calculates the firing strength of each rule via the product operator:

$$O_{2,i} = w_i = \mu_{A_i}(x) \cdot \mu_{B_i}(y), \quad i = 1, 2 \quad (3)$$

This layer effectively implements the T-norm operator for fuzzy AND operations.

2.1.3 Layer 3 (Normalization Layer):

Nodes in this layer compute normalized firing strengths:

$$O_{3,i} = \bar{w}_i = \frac{w_i}{w_1 + w_2}, \quad i = 1, 2 \quad (4)$$

Normalization ensures that the contribution of each rule to the final output is proportional to its relative firing strength.

2.1.4 Layer 4 (Consequent Layer)

Each adaptive node in this layer computes the following output:

$$O_{4,i} = \bar{w}_i f_i = \bar{w}_i (p_i x + q_i y + r_i), \quad i = 1, 2 \quad (5)$$

Where $\{p_i, q_i, r_i\}$ are the consequent parameters.

2.1.5 Layer 5 (Output Layer)

The single node in this layer computes the overall output as the weighted sum of all rule outputs:

$$O_{5,i} = \sum_{i=1}^2 \bar{w}_i f_i = \frac{\sum_{i=1}^2 w_i f_i}{\sum_{i=1}^2 w_i} \quad (6)$$

The hybrid learning mechanism combines the least squares method for identifying consequent parameters with gradient descent for updating premise parameters, resulting in accelerated convergence and enhanced precision. This parameter optimization process enables ANFIS to achieve superior learning capability while maintaining the interpretability of fuzzy reasoning.

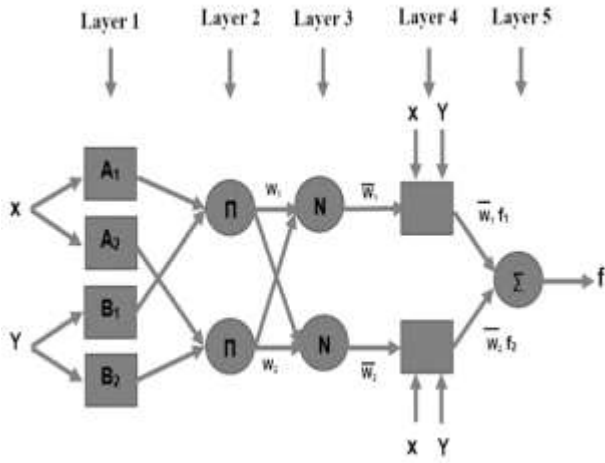


Figure 1. Architecture of the Adaptive Neuro-Fuzzy Inference System (ANFIS).

3. Metaheuristic Algorithms

Metaheuristic optimization techniques have gained significant popularity in recent years for solving complex optimization problems across various domains. According to the literature, metaheuristics can be classified into two main categories based on the number of solutions they manipulate: population-based methods and single-solution methods [11]. Population-based meta-heuristics maintain and improve multiple candidate solutions simultaneously. This category can be subdivided into two major subfields: Swarm Intelligence (SI) algorithms, which are inspired by the collective behavior of decentralized, self-organized systems found in nature, such as swarms of birds or colonies of ants; and Evolutionary Algorithms (EAs), which are based on principles of biological evolution, including selection, crossover, and mutation [12]. This paper focuses specifically on swarm intelligence algorithms, which have demonstrated remarkable effectiveness in tuning the parameters of complex models. The core objective of this work is to employ selected SI algorithms to optimize the Adaptive Neuro-Fuzzy Inference System (ANFIS), enhancing its performance and predictive accuracy by fine-tuning its membership functions and consequent parameters.

3.1 Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) [13] is a population-based metaheuristic algorithm inspired by the social behavior of species such as birds and fish. Developed by Kennedy and Eberhart in 1995, PSO simulates the movement of particles in a multidimensional search space, where each particle represents a potential solution. The algorithm

operates by updating particle positions based on two key components: the personal best position (pbest) and the global best position (gbest). The velocity and position update equations are defined as:

$$v_i^{t+1} = wv_i^t + c_1r_1(pbest_i - x_i^t) + c_2r_2(gbest - x_i^t) \quad (7)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (8)$$

Where w is the inertia weight, c_1 and c_2 are acceleration coefficients, and r_1 , r_2 are random values in $[0, 1]$. PSO is widely appreciated for its simplicity, fast convergence, and effectiveness in solving complex nonlinear optimization problems.

3.2 Genetic Algorithm (GA)

Genetic Algorithm (GA) [14] is an evolutionary optimization technique based on the principles of natural selection and genetics. GA operates on a population of candidate solutions, called chromosomes, which evolve over generations through genetic operators: selection, crossover, and mutation. Selection favors individuals with higher fitness, crossover combines parent chromosomes to produce offspring, and mutation introduces random changes to maintain diversity. The algorithm is particularly effective for multimodal optimization, as it explores multiple regions of the search space simultaneously and avoids premature convergence to local optima. GA has been successfully applied in various fields, including engineering, economics, and artificial intelligence.

3.3 Grey Wolf Optimizer (GWO)

Grey Wolf Optimizer (GWO) [15] is a metaheuristic algorithm inspired by the social hierarchy and hunting behavior of grey wolves. The algorithm models the leadership structure of a wolf pack, which includes four types of wolves: alpha (α), beta (β), delta (δ), and omega (ω). The optimization process mimics the encircling and attacking of prey, where the positions of wolves are updated based on the positions of the alpha, beta, and delta wolves. The position update equations are defined as:

$$\vec{D} = |\vec{C} \cdot \vec{X}_p(t) - \vec{X}(t)| \quad (9)$$

$$\vec{X}(t+1) = \vec{X}_p(t) - \vec{A} \cdot \vec{D} \quad (10)$$

Where $\vec{A}$ and $\vec{C}$ are coefficient vectors, and $\vec{X}_p$ represents the position of the prey. GWO is known for its simplicity, low computational cost, and strong performance in solving complex optimization

problems, often outperforming established algorithms.

3.4 Salp Swarm Algorithm (SSA)

Salp Swarm Algorithm (SSA) [16] is a recent metaheuristic inspired by the collective behavior of salps, which form chain-like swarms in deep oceans. The algorithm divides the population into leaders and followers. The leader guides the swarm toward the food source (best solution), while followers update their positions sequentially to maintain chain structure. The position update for the leader is defined as:

$$x_j^1 = \begin{cases} F_j + c_1 \cdot ((ub_j - lb_j) \cdot c_2 + lb_j) & \text{if } c_3 \geq 0.5 \\ F_j - c_1 \cdot ((ub_j - lb_j) \cdot c_2 + lb_j) & \text{if } c_3 \leq 0.5 \end{cases} \quad (11)$$

where F_j is the food source position, ub_j and lb_j are the upper and lower bounds, and c_1, c_2, c_3 are control parameters. Followers update their positions using:

$$x_j^i = \frac{1}{2} (x_j^i + x_j^{i-1}) \quad (12)$$

SSA is recognized for its simplicity, robustness in multimodal search spaces, and ability to avoid local optima, making it suitable for complex optimization problems.

3.5 Sand Cat Swarm Optimization (SCSO)

Sand Cat Swarm Optimization (SCSO) [17] is a recent metaheuristic algorithm inspired by the hunting behavior of the sand cat. This small predator is renowned for its ability to detect subsurface prey through the sand using its highly sensitive hearing. The algorithm simulates two key behaviors: the search for prey, which corresponds to exploring the search space, and the attack, which corresponds to exploiting promising solutions.

The core mechanism of SCSO relies on an auditory sensitivity operator that adaptively switches between exploration and exploitation. The position of each sand cat (agent) is updated according to the equation:

$$\vec{X}(t+1) = \vec{X}_{best}(t) - \vec{r} \cdot \cos(\theta) \cdot |rand(0,1) \cdot \vec{X}_{best}(t) - \vec{X}| \quad (13)$$

where r is a random vector, θ is a random angle. SCSO is appreciated for its effective balance between exploration and exploitation, its implementation simplicity, and its performance on complex benchmark functions.

3.6 FOX Optimizer

The FOX Optimizer [18] is a metaheuristic algorithm inspired by the hunting behavior of red foxes, which locate prey under snow cover by measuring sound propagation time and execute precise jumps to capture them. Mathematically, the algorithm alternates between an exploration phase:

$$X_{(t+1)} = BestX_t \times rand(1, dimension) \times MinT \times a \quad (14)$$

and an exploitation phase based on the estimated distance to prey ($Dist_Fox_Prey$) and jump height ($Jump$):

$$X_{(t+1)} = Dist_Fox_Prey \times Jump \times c_i \quad (15)$$

where $MinT$ is the minimum average time, a is a decreasing adaptive parameter, t is the average sound travel time, $Jump = 0.5 \times 9.81 \times t^2$ and $c_i \in \{0.18, 0.82\}$ determined by a probability p .

Through its balance between exploration and exploitation and dynamic jump mechanisms, FOX avoids local optima and converges rapidly toward optimal solutions, outperforming well-known algorithms on benchmark functions and engineering problems.

3.7 Greylag Goose Optimization (GGO)

Greylag Goose Optimization (GGO) [19] is a nature-inspired metaheuristic algorithm that mimics the collective behavior of greylag geese during their migratory flights. When geese fly in a distinctive "V" formation, they optimize their energy efficiency by reducing air resistance, enabling them to travel much farther as a group. This intelligent group behavior is mathematically captured in the algorithm's fundamental update mechanism :

$$\vec{X}(t+1) = \vec{X}_{best}(t) - \vec{A} \cdot |\vec{C} \cdot \vec{X}_{best}(t) - \vec{X}(t)| \quad (16)$$

where $\vec{X}_{best}$ is the current best solution (leader) and A and C are parameter vectors that guide the exploration movement. The algorithm dynamically divides the population into exploration and exploitation groups, adjusting their sizes based on search progress. GGO has been successfully applied to feature selection and engineering design problems, demonstrating superior performance compared to other metaheuristic algorithms due to its effective balance between global exploration and local refinement.

toward a global optimum, with the explicit objective

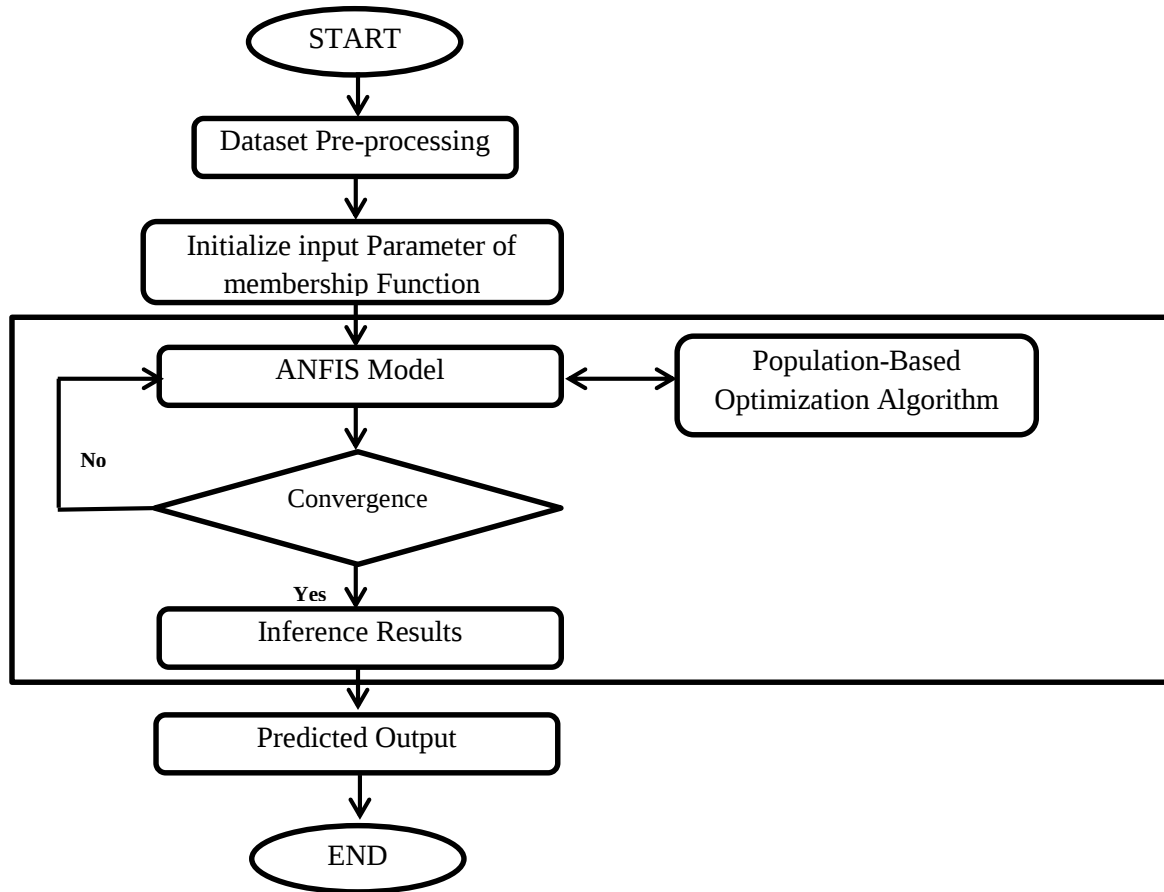


Figure 2. Flowchart of the hybrid ANFIS optimization methodology with population-based optimization algorithm

4. Adaptive Neuro-Fuzzy Inference System Parameter Tuning Using Population Based Optimization Algorithms

The integration of the ANFIS architecture with population-based optimization algorithms is illustrated in Figure 2. In this hybrid approach, seven population-based methods (PSO, GA, GWO, SSA, SCSO, FOX and GGO) are used to optimize the parameters of ANFIS in order to improve its learning ability and prediction accuracy. The main objective of this work is to evaluate the impact of these population-based algorithms on ANFIS optimization and to analyze their performance on various datasets. The adopted methodology implements a structured, hybrid learning pipeline to develop a reliable predictive model. The process begins with an initial phase of dataset preprocessing to ensure data quality and compatibility. Following this, the parameters controlling the system's membership functions are initialized, establishing a foundational configuration for the model's architecture. The core of the learning process employs a population-based optimization algorithm to automatically and iteratively refine these parameters. This metaheuristic search operates by evolving a population of candidate solutions

of minimizing a predefined cost function, typically the mean squared error between predictions and actual values. By integrating the guided stochastic search of the optimization algorithm, the hybrid ANFIS achieves efficient convergence and produces a high-performance model with robust generalization capabilities.

5. Results and Discussion

To evaluate the performance of the ANFIS model optimized by population-based algorithms, three benchmark datasets were employed. These datasets differ in nature and complexity, covering both synthetic and real-world classification problems. A summary of the main characteristics of these datasets is provided in the table 1.

Table 2 summarizes the performance of the Adaptive Neuro-Fuzzy Inference System (ANFIS) and its optimized variants using seven population-based optimization algorithms across three benchmark datasets: Banana, Blood Transfusion, and Banknote Authentication. For each algorithm, the mean (Avg) and standard deviation (Std) of the mean squared error (MSE) were computed over 30 independent runs to ensure statistical robustness. The basic

ANFIS model consistently showed higher testing MSE values (0.9763, 0.0989, and 0.0293 for Banana, Blood Transfusion, and Banknote datasets, respectively), indicating limited generalization and highlighting the need for metaheuristic optimization. Across all datasets, the application of optimization algorithms significantly reduced testing MSE (by approximately 40–60%) compared with the baseline ANFIS. The Friedman test confirmed that these differences were statistically significant in all cases ($p < 0.05$), indicating that the algorithms have distinct optimization behaviors. Wilcoxon post-hoc tests were then applied to identify the best-performing methods and verify pairwise statistical differences.

For the Banana dataset, the Friedman test produced a result of $\chi^2(6, N = 30) = 17.46$ with $p = 0.0078$, revealing that not all optimizers performed equally. PSO achieved the lowest testing MSE (0.4735) with a relatively low variance (0.0781), outperforming most competitors. The Wilcoxon test (Table 3) confirmed that PSO significantly outperformed GWO, SSA, SCSO, GGO, and FOX, while its difference with GA ($p = 0.214$) was not statistically significant. This indicates that both PSO and GA achieved comparable predictive accuracy. Algorithms such as GWO and FOX, with higher MSE values (0.6097 and 0.6460, respectively), likely suffered from premature convergence.

On the Blood Transfusion dataset, the Friedman test produced $\chi^2(6, N = 30) = 18.27$ with $p = 0.0056$, confirming significant performance differences among algorithms. The SCSO algorithm achieved the lowest testing MSE (0.0947) with a small standard deviation (0.0030), reflecting both accuracy and convergence stability. Wilcoxon pairwise tests revealed that SCSO significantly outperformed all algorithms except PSO ($p = 0.181$), confirming statistically comparable results between these two optimizers. The superior performance of SCSO can be attributed to its adaptive exploration–

exploitation strategy inspired by sand cat hunting behavior, which enables dynamic control of search intensity and prevents stagnation in local minima.

For the Banknote Authentication dataset, the Friedman test resulted in $\chi^2(6, N = 30) = 21.67$ with $p = 0.0013$, again showing clear performance differences. PSO achieved the lowest testing MSE (0.0136), marking a 53.6% reduction compared with the non-optimized ANFIS model. The Wilcoxon analysis confirmed that PSO significantly outperformed SSA, GGO, FOX, and GWO, while its differences with GA ($p = 0.118$) and SCSO ($p = 0.136$) were not significant. This finding highlights PSO's excellent generalization capability and adaptability to different data distributions, driven by its dynamic velocity–position update mechanism that effectively balances exploration and exploitation.

Metaheuristic optimization significantly improves ANFIS's predictive performance, stability, and generalization ability. PSO emerged as the most robust and generalizable optimizer, delivering the best results on both the Banana and Banknote datasets. SCSO showed the strongest performance on the Blood Transfusion dataset, achieving excellent accuracy with minimal variance. GA showed competitive but slightly less stable results, while GWO, SSA, GGO, and FOX often displayed limited exploration ability, resulting in higher test errors.

The Friedman and Wilcoxon statistical analyses confirm that PSO and SCSO are the most effective algorithms for optimizing ANFIS parameters. Their ability to achieve low MSE, consistent convergence, and balanced exploration–exploitation dynamics establishes them as the leading strategies among population-based metaheuristics. These findings provide reliable evidence that hybrid ANFIS–metaheuristic frameworks can significantly enhance predictive accuracy, generalization performance, and robustness across diverse problem domains.

Table 1. Summary of the benchmark datasets used in this study.

Dataset	Instances	Features	Class Distribution	Feature Type	Target Variable
Banana [20]	5301	2	2651 (Class 1), 2650 (Class 2)	Continuous	Class (binary)
Blood Transfusion Service Center [21]	748	4	570 (No Donation), 178 (Donation)	Numerical	Donated_Blood (0 = No, 1 = Yes)
Banknote Authentication [22]	1372	4	762 (Genuine), 610 (Forged)	Continuous	Class (0 = Forged, 1 = Genuine)

Table 2. Performance comparison of ANFIS and ANFIS optimized by seven population-based optimization algorithms across three benchmark datasets.

		Banana dataset		Blood Transfusion Service dataset		BankNote Authentication dataset	
		Train	Test	Train	Test	Train	Test
ANFIS		0.9847	0.9763	0.1729	0.0989	0.0344	0.0293
GA	Avg	0,501541	0,476105	0,165634	0,098147	0,022222	0,021947
	Std	0,08117	0,0785	0,002383	0,004339	0,001638	0,001602
PSO	Avg	0,485642	0,473484	0,156916	0,095394	0,012556	0,013611
	Std	0,077365	0,078131	0,002828	0,004077	0,00419	0,003932
GWO	Avg	0,623062	0,609663	0,16405	0,095533	0,023426	0,023017
	Std	0,066964	0,070054	0,003998	0,004251	0,005356	0,004271
SSA	Avg	0,736508	0,718872	0,164293	0,097845	0,02122	0,021176
	Std	0,072587	0,072312	0,000978	0,001215	0,001596	0,001498
SCSO	Avg	0,611749	0,600954	0,161068	0,09466	0,020849	0,021082
	Std	0,072837	0,074588	0,001678	0,003021	0,001907	0,002
GGO	Avg	0,825326	0,812715	0,168236	0,100057	0,026948	0,025538
	Std	0,056738	0,064206	0,001269	0,002091	0,001393	0,001031
FOX	Avg	0,664124	0,645997	0,164298	0,096869	0,023078	0,023263
	Std	0,055339	0,058772	0,001862	0,002906	0,001589	0,001735

Table 3. Wilcoxon post-hoc test results comparing each algorithm with the best-performing optimizer on each dataset

Comparison (vs Best Algorithm)	Banana (Best = PSO)	Blood Transfusion (Best = SCSO)	Banknote Authentication (Best = PSO)	Significance ($p < 0.05$)
GA	0.214	0.001	0.118	No / Yes / No
SCSO	0.011	—	0.136	Yes / — / No
SSA	0.007	0.002	0.007	Yes
GGO	0.006	0.004	0.012	Yes
GWO	0.002	0.009	0.002	Yes
FOX	0.001	0.003	0.004	Yes
PSO	—	0.181	—	— / No / —

6. Conclusion

This study has demonstrated the significant potential of integrating metaheuristic optimization algorithms with the Adaptive Neuro-Fuzzy Inference System (ANFIS) to enhance its predictive performance and generalization capability. Experimental results on three distinct benchmark datasets demonstrate that the hybrid ANFIS models outperformed the standard ANFIS. The Friedman and Wilcoxon statistical tests provided robust evidence that these improvements are statistically significant, confirming that the choice of optimization algorithm substantially influences model performance.

Among the seven population-based algorithms evaluated, Particle Swarm Optimization (PSO) and Sand Cat Swarm Optimization (SCSO) emerged as the most effective optimizers. PSO demonstrated remarkable robustness and generalization, achieving the best performance on the Banana and Banknote Authentication datasets, while SCSO performed best on the Blood Transfusion dataset, achieving high accuracy with minimal variance. These algorithms successfully balanced exploration and exploitation, avoiding premature convergence and effectively navigating the complex parameter space of ANFIS.

The findings confirm the critical importance of metaheuristic-based parameter tuning in overcoming the limitations of conventional ANFIS training. Future research will focus on extending this approach to larger and more complex datasets and adapting these hybrid models for real-world applications in dynamic environments.

Author Statements:

- **Ethical approval:** The conducted research is not related to either human or animal use.
- **Conflict of interest:** The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper
- **Acknowledgement:** The authors declare that they have nobody or no-company to acknowledge.
- **Author contributions:** The authors declare that they have equal right on this paper.
- **Funding information:** The authors declare that there is no funding to be acknowledged.
- **Data availability statement:** The data that support the findings of this study are available on

request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

References

- [1] Xu, Y., Liu, X., Cao, X., Huang, C., Liu, E., Qian, S., ... & Zhang, J. (2021). Artificial intelligence: A powerful paradigm for scientific research. *The Innovation*, 2(4). <https://doi.org/10.1016/j.xinn.2021.100179>
- [2] Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245), 255-260. Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245), 255-260. <https://doi.org/10.1126/science.aaa8415>
- [3] Alzubi, J., Nayyar, A., & Kumar, A. (2018, November). Machine learning from theory to algorithms: an overview. In *Journal of physics: conference series* (Vol. 1142, p. 012012). IOP Publishing. <https://doi.org/10.1088/1742-6596/1142/1/012012>
- [4] Pickering, L., Cohen, K., & De Baets, B. (2025). A Narrative Review on the Interpretability of Fuzzy Rule-Based Models from a Modern Interpretable Machine Learning Perspective. *International Journal of Fuzzy Systems*, 1-20. <https://doi.org/10.1007/s40815-025-02022-z>
- [5] Jang, J. S. (1993). ANFIS: adaptive-network-based fuzzy inference system. *IEEE transactions on systems, man, and cybernetics*, 23(3), 665-685. <https://doi.org/10.1109/21.256541>
- [6] Talpur, N., Salleh, M.N.M., Hussain, K., Ali, H. (2019). Modified ANFIS with Less Model Complexity for Classification Problems. In: Omar, S., Haji Suhaili, W., Phon-Amnuaisuk, S. (eds) *Computational Intelligence in Information Systems*. CIIS 2018. *Advances in Intelligent Systems and Computing*, vol 888. Springer, Cham. https://doi.org/10.1007/978-3-030-03302-6_4
- [7] Karaboga, D., & Kaya, E. (2019). Adaptive network based fuzzy inference system (ANFIS) training approaches: a comprehensive survey. *Artificial Intelligence Review*, 52(4), 2263-2293. <https://doi.org/10.1007/s10462-017-9610-2>
- [8] Jang, J. S. (1993). ANFIS: adaptive-network-based fuzzy inference system. *IEEE transactions on systems, man, and cybernetics*, 23(3), 665-685. <https://doi.org/10.1109/21.256541>
- [9] Shihabudheen, K. V., & Pillai, G. N. (2018). Recent advances in neuro-fuzzy system: A survey. *Knowledge-Based Systems*, 152, 136-162. <https://doi.org/10.1016/j.knosys.2018.04.014>
- [10] Takagi, T., & Sugeno, M. (1985). Fuzzy identification of systems and its applications to modeling and control. *IEEE transactions on systems, man, and cybernetics*, (1), 116-132. <https://doi.org/10.1109/TSMC.1985.6313399>
- [11] Blum, C., & Roli, A. (2003). Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM computing surveys (CSUR)*, 35(3), 268-308. <https://doi.org/10.1145/937503.937505>
- [12] Boussaïd, I., Lepagnot, J., & Siarry, P. (2013). A survey on optimization metaheuristics. *Information sciences*, 237, 82-117. <https://doi.org/10.1016/j.ins.2013.02.041>
- [13] Kennedy, J., Eberhart, R., 1995. Particle swarm optimization. In: *Proceedings of ICNN'95-International Conference on Neural Networks*, 4, 1942-1948. <http://dx.doi.org/10.1109/ICNN.1995.488968>
- [14] Mitchell, M. (1998). *An introduction to genetic algorithms*. MIT press.
- [15] Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey Wolf Optimizer. *Advances in Engineering Software*, 69, 46-61. <https://doi.org/10.1016/j.advengsoft.2013.12.007>
- [16] Mirjalili, S., Gandomi, A. H., Mirjalili, S. Z., Saremi, S., Faris, H., & Mirjalili, S. M. (2017). Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems. *Advances in Engineering Software*, 114, 163-191. <https://doi.org/10.1016/j.advengsoft.2017.07.002>
- [17] Seyyedabbasi, A., & Kiani, F. (2023). Sand Cat swarm optimization: A nature-inspired algorithm to solve global optimization problems. *Engineering with computers*, 39(4), 2627-2651. <https://doi.org/10.1007/s00366-022-01604-x>
- [18] Mohammed, H., & Rashid, T. (2023). FOX: a FOX-inspired optimization algorithm. *Applied Intelligence*, 53(1), 1030-1050. <https://doi.org/10.1007/s10489-022-03533-0>
- [19] El-Kenawy, E. S. M., Khodadadi, N., Mirjalili, S., Abdelhamid, A. A., Eid, M. M., & Ibrahim, A. (2024). Greylag goose optimization: nature-inspired optimization algorithm. *Expert Systems with Applications*, 238, 122147. <https://doi.org/10.1016/j.eswa.2023.122147>
- [20] Timrie. (2023). Banana [Dataset]. Kaggle. Retrieved October 20, 2025, from <https://www.kaggle.com/datasets/timrie/banana>
- [21] Yeh, I-C. (2008). Blood Transfusion Service Center [Dataset]. UCI Machine Learning Repository. <https://doi.org/10.24432/C5GS39>
- [22] Lohweg, V. (2012). Banknote Authentication [Dataset]. UCI Machine Learning Repository. <https://doi.org/10.24432/C55P57>