INTERNATIONAL JOURNAL OF RESEARCH AND INNOVATION IN APPLIED SCIENCE (IJRIAS)
ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue VIII August 2026

Binarization of Metaheuristic Optimization Algorithm: A Comparative Analysis

Eris Zeqo[1], Edjola Naka[1*], Vassil Guliashki[2]

[1]Department of Computer Science, Aleksandër Moisiu University, Durres, Albania

[2]Institute of Information and Communication Technologies, Bulgarian Academy of Sciences, Sofia, Bulgaria

***Corresponding Author**

DOI: https://doi.org/10.51584/IJRIAS.2026.11080013

Received: 17 August 2026; Accepted: 22 August 2026; Published: 29 August 2026

ABSTRACT

Metaheuristic algorithms are a kind of optimization algorithms, which are widely used in reducing the number of variables, and choosing the most relevant, and identifiable ones in medium or high dimensional datasets. This topic, known as feature selection is widely researched and applied where binary metaheuristic algorithms are developed using as an evaluator of the quality of the feature subsets machine learning algorithms. The binary metaheuristics could be created by transforming the continuous space of the metaheuristic by using some operators named transfer function, into a binary one, since feature selection is a binary optimization problem. This paper aims to offer a comparative analysis of three distinct binary variants of metaheuristic optimization algorithms utilizing five different groups of transfer functions to assess their predictive efficacy. Four S-shaped, V-shaped, U-shaped, Q-shaped, and two X-shaped types are proposed for the binarization process. Salp Swarm Optimization, Ant Lion Optimization, and Teaching Learning-Based Optimization algorithms are chosen as the foundational algorithms for assessing each transfer function and comparing their impact on key metrics, including accuracy, F-score, sensitivity, average number of selected features, fitness, and time efficiency. The current work utilizes a medical dataset to predict the existence of gallstones. The experiment aims to compare and propose new solutions applied to the feature selection process. The optimal combinations of transfer functions and metaheuristic algorithms are produced using the combination of the Q3 transfer function together with a binary teaching-based learning method and the V2 transfer function combined with binary antlion optimization, achieving an accuracy of approximately 78%. The research emphasizes the necessity of appropriate binarization strategies and provides insights into selecting transfer functions to improve feature selection in medical data analysis.

Keywords: Metaheuristic algorithm, Binary Optimization, Transfer functions, Feature selection, Gallstone prediction, Medical Analysis

INTRODUCTION

Heuristic approaches are frequently suggested as solutions for large-scale NP-hard or nonlinear problems (Marti et al., 2025). Metaheuristic optimization algorithms (MHAs) are general algorithmic frameworks that guide lower-level heuristics to efficiently explore a search space and arrive at high-quality solutions without guaranteeing optimality. Metaheuristic algorithms have also been widely applied to improve machine learning models themselves, including tasks such as hyper parameter tuning, feature selection (FS), and data partitioning.

FS is a binary optimization problem, aiming to identify the most relevant subset of features while reducing dimensionality, computational cost, and overfitting. Most metaheuristic algorithms update solutions as real-valued vectors, but many optimization problems require binary decisions. Consequently, continuous metaheuristic algorithms must be modified to function within binary search spaces. Two-step binarization is an approach used in binary/metaheuristic optimization when the underlying algorithm naturally works in a continuous space, but the problem itself is binary or discrete. The two-step binarization technique involves using transfer functions (TF) to transform continuous values within the range of 0 to 1 and then transferring the real number using methods such as standard or complement to convert them into binary values, 0 or 1. The value 1 in FS indicates the selection of the feature; if not, it remains unselected. The MHA's optimal solution involves storing the more significant features of the dataset in a vector of features. Transfer functions in the field of optimization were introduced by Kennedy & Eberhart (1997). Their main advantage is that they provide a probability between 0 and 1 at a low computational cost. These methodologies facilitate the utilization of continuous metaheuristics without necessitating any modifications to the operators. Thaher et al. (2021) stated that a suitable binarization mechanism with a TF has a substantial impact on the exploitative and exploratory potentials of the utilized binary algorithm.

A recent work about metaheuristics and their binarization is conducted from Becerra-Rozas et al. (2023), where the study uses a standard systematic review consisting of the analysis of 512 publications. They conclude that there is no single general technique capable of efficient binarization; instead, there are multiple forms with different performances, however the most used method to binarize continuous metaheuristics is the transfer function.

There are several types of transfer functions, with the most used S-shaped and V-shaped TFs (Mirjalili & Lewis, 2013), as well as more recent variants such as X-shaped (Beheshti, 2021), U-shaped (Mirjalili et al, 2020), Z-shaped (Guo et al., 2020), linear (Wang et al., 2008), and quadratic functions (Too et al., 2019). Despite this diversity, comparative analyses across multiple transfer function families and metaheuristic algorithms remain limited.

In this paper, we contribute with a comparative analysis of multiple transfer function families applied to binary metaheuristic optimization algorithms. Specifically, we investigate four S-shaped, V-shaped, U-shaped, and Q-shaped transfer functions mentioned above, as well as two X-shaped variants, combined with both standard and complement binarization rules. S and V-shaped are classical TF, meanwhile U, Q, and X-shaped are included for broader comparative coverage. The evaluation is performed using three representative metaheuristic algorithms: Binary Salp Swarm Algorithm (BSSA), Binary Ant Lion Optimizer (BALO), and Binary Teaching–Learning-Based Optimization (BTLBO).

The main contributions of this study are as follows:

- A comprehensive comparison of multiple transfer function families within a unified experimental framework.
- An analysis of the impact of binarization strategies on feature selection performance.
- The identification of effective combinations of transfer functions and metaheuristic algorithms.
- An application to a real-world medical dataset for gallstone prediction.
- The findings are intended to advance understanding of the role of transfer functions in binary metaheuristic algorithms and their impact on feature selection.

LITERATURE REVIEW

The utilization of transfer functions in feature selection derives from the binary formulation of Particle Swarm Optimization (BPSO) introduced by Kennedy & Eberhart (1997). This method utilized a sigmoid transfer function to correlate continuous velocities with probabilities, facilitating the selection of each feature. This framework was subsequently utilized in several feature selection studies due its simplicity and ease of implementation. Rashedi et al. (2010) expanded the notion of binarization to the Binary Gravitational Search Algorithm utilizing the tanh function (Type V2 of V-shaped), with findings indicating its efficacy in identifying relevant features for classification tasks.

Mirjalili & Lewis (2013), proposed more definitely the V-shaped transfer functions as an alternative to sigmoid TF. The superiority of V-shaped transfer functions over S-shaped ones was empirically validated across multiple benchmark datasets. The integration of transfer functions into newer metaheuristic algorithms further advanced feature selection area. In the work of Emary et al. (2016) are proposed binary variants of the Grey Wolf Optimize using both S-shaped and V-shaped transfer functions for feature selection. V-shaped transfer functions consistently achieved higher classification accuracy with fewer selected features. In the study of Mafarja et al. (2017) there are used three V-shaped and S-shaped TF for the process of binarization of binary ant lion optimization. S-shaped functions may converge faster but are more prone to premature

convergence. They conclude that the choice of TF significantly impacts binary ALO performance, and V-shaped TF provide a better balance between exploration and exploitation. In another work of Naka (2023), there are employed the four S-shaped and V-shaped for binarization of the volleyball premier league algorithm adapted for feature selection. The experiments are conducted on 10 different Parkinson datasets, and for each one is selected the best one from an overall of 8 transfer functions. V3 is the more employed from the V-shaped functions, and S2, and S3 from the S-shaped ones. In the study of Nadimi-Shahraki (2021) it is created the binary moth flame optimization using S-shaped, V-shaped, and U-shaped transfer functions to convert the moth flame optimization (MFO) algorithm from continuous to binary. These categories of binary MFO were evaluated on seven medical datasets. The U-shaped functions outperform the V-shaped and S-shaped in terms of classification accuracy and minimized the number of selected features. In the work of Kumar et al. (2021) it is applied the S-shaped, and V-shaped to binarize Seagull Optimization Algorithm (SOA). The results revealed that BSOA_V4 has the highest performance among all. In the study of Bas (2024), twelve different transfer functions were used (four Z-shaped, four U-shaped, and four Taper-shaped) on the Binary Dwarf Mongoose Optimization algorithm (BinDMO). The most successful BinDMO variation was the one generated from Z-shaped TFs.

There are also cases when new TFs shows better performance compared with older ones, as in the study of Beşkirli (2025). The pied kingfisher optimizer (PKO) algorithm is modified for the uncapacitated facility location problem (UFLP) using mode-based TF, evaluated on fifteen distinct Cap issues. They perform exceptionally well on all problems regarding both optimal and average metrics, and perform better than traditional TFs. A new proposed type V-TF were used as a binary version approach for SSA. The SSA-New V3-TF produces the highest accuracy and less runtime compared to other algorithms in sentiment analysis (Kristiyanti et al., 2023). Despite these advances, most existing studies focus on a limited number of transfer function types or a single metaheuristic algorithm. In-depth comparisons between various transfer function families and algorithms continue to be necessary for understanding their impact. Comparative studies between various transfer function families and metaheuristic algorithms are still few despite this broad range. This serves as the motivation for the current study, which offers a unified framework for evaluating how various transfer functions affect binary metaheuristic optimization in feature selection problems studies.

MATERIALS AND METHODS

This section presents the methodological framework of the comparative analysis, including the selected metaheuristic algorithms, the binarization mechanism based on transfer functions, the dataset, and the experimental settings.

Metaheuristic algorithms

Metaheuristic algorithms have been widely employed for feature selection and other optimization tasks due to their ability to efficiently explore large and complex search spaces. In the present study, three population-based metaheuristic algorithms are used as the basis for binary feature selection: Teaching–Learning-Based Optimization (TLBO), Ant Lion Optimization (ALO), and Salp Swarm Algorithm (SSA). Their binary variants are denoted as BTLBO, BALO, and BSSA, respectively.

These algorithms are applied to identify the most informative subset of features from the gallstone dataset. Each candidate solution is evaluated by a fitness function that combines classification error and subset size. The original continuous formulations of the algorithms are adapted to the binary domain through transfer-function-based binarization. All methods were implemented in the R programming language, building on the original formulations of TLBO (Heris, 2015), SSA (Too, 2021, Mirjalili et al., 2017], and ALO (Emary et al., 2016, Mirjalili, 2015].

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue VIII August 2026

Teaching–Learning-Based Optimization

Teaching–Learning-Based Optimization, introduced by Rao et al. (2011), is inspired by the teaching and learning process in a classroom. The algorithm consists of two main phases: the teacher phase and the learner phase. In the teacher phase, the population is interpreted as a group of students, while the optimization variables correspond to subjects. The best solution in the population acts as the teacher and attempts to improve the mean performance of the class. In the learner phase, learners improve their knowledge through pairwise interactions with other learners. If a learner interacts with another learner of higher quality, the former updates its position accordingly. Both teacher and learner phases are repeated iteratively until a criterion is fulfilled.

Ant Lion Optimization

Ant Lion Optimization, introduced by Mirjalili (2015), is a nature-inspired algorithm that replicates the predatory behaviour of antlions. Within these framework ants symbolize alternative solutions, whereas antlions denote guiding solutions that define the search process.

The exploration phase is conducted via random walks of ants inside the search space. During exploitation, each ant is influenced by two antlions: one selected via roulette-wheel selection and the elite antlion, which is the best solution found so far. This technique supports diversity while simultaneously strengthening convergence toward attractive areas of the search space.

Salp Swarm Algorithm

The Salp Swarm Algorithm, introduced by Mirjalili et al. (2017), was inspired by the swarming behaviour of salps moving in chain formations in the ocean. The population is divided into a leader and followers. The leader updates its position with respect to the best solution found so far, while the followers update their positions according to the salp immediately ahead of them.

SSA provides a balance between exploration and exploitation through a parameter that decreases over the iterations. Owing to its adaptability, robustness, and ability to perform global search, SSA has been successfully applied to a variety of optimization problems, including feature selection (Kristiyanti et al., 2023, Mirjalili et al., 2017).

Two-step binarization method

Since the selected metaheuristic algorithms are originally designed for continuous search spaces, they must be adapted to operate in binary domains. In this work, binarization is performed using the widely adopted two-step strategy. In the first step, a transfer function transforms a continuous position value into a probability-like value in the interval [0,1]. In the second step, this value is converted into a binary decision using a discretization rule. In the context of feature selection, a binary value of 1 indicates that the corresponding feature is selected, whereas 0 indicates that it is excluded. Five groups of transfer functions are examined: S-shaped, V-shaped, U-shaped, Q-shaped, and X-shaped. In total, four S-shaped, four V-shaped, four U-shaped, four Q-shaped, and two X-shaped transfer functions are evaluated. The mathematical definitions of the transfer functions considered in this study are presented consequently. Equations (1) – (4) represent the S-shaped ones. The solution generated by the metaheuristic x is given as an input to the specific transfer function.

$$(S1)\ T(x) = \frac{1}{1+e^{-2x}} \tag{1}$$

$$(S2)\ T(x) = \frac{1}{1+e^{-x}} \tag{2}$$

$$(S3)\ T(x) = \frac{1}{1+e^{-\frac{x}{2}}} \tag{3}$$

$$(S4)\ T(x) = \frac{1}{1+e^{-\frac{x}{3}}} \tag{4}$$

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue VIII August 2026

Equations (5) – (8) present the V-shaped types.

$$(V1)\ \ T(x) = \left|\operatorname{erf}\left(\frac{\sqrt{\pi}}{2}x\right)\right| \qquad (5)$$
$$(V2)\ \ T(x) = |\tanh(x)| \qquad (6)$$
$$(V3)\ \ T(x) = \left|\frac{x}{\sqrt{1+x^2}}\right| \qquad (7)$$
$$(V4)\ \ T(x) = \left|\frac{2}{\pi}\arctan\left(\frac{\pi}{2}x\right)\right| \qquad (8)$$

Equations (9) – (12) show the formulation of U-shaped TF.

$$(U1)\ \ T(x) = |x^{1.5}| \qquad (9)$$
$$(U2)\ \ T(x) = |x^2| \qquad (10)$$
$$(U3)\ \ T(x) = |x^3| \qquad (11)$$
$$(U4)\ \ T(x) = |x^4| \qquad (12)$$

Equations from (13) to (16) show the Q-shaped TF.

$$(Q1)\ \ T(x) = \begin{cases} \left|\frac{x}{0.5x_{max}}\right| & if\ x < 0.5x_{max} \\ 1 & otherwise \end{cases} \qquad (13)$$
$$(Q2)\ T(x) = \begin{cases} \left(\frac{x}{0.5x_{max}}\right)^2 & if\ x < 0.5x_{max} \\ 1 & otherwise \end{cases} \qquad (14)$$
$$(Q3)\ T(x) = \begin{cases} \left(\frac{x}{0.5x_{max}}\right)^3 & if\ x < 0.5x_{max} \\ 1 & otherwise \end{cases} \qquad (15)$$
$$(Q4)\ T(x) = \begin{cases} \left(\frac{x}{0.5x_{max}}\right)^{1/2} & if\ x < 0.5x_{max} \\ 1 & otherwise \end{cases} \qquad (16)$$

Finally, two forms of X-shaped TF are illustrated below (Eq. 17, and Eq. 18):

$$(X1)\ \ T(x) = \frac{-x}{1+|-x|*0.5} + 0.5 \qquad (17)$$
$$(X2)\ \ T(x) = \frac{x-1}{1+|x-1|*0.5} + 0.5 \qquad (18)$$

Binarization rules

Two binarization rules are considered in the second step of the procedure: the standard rule and the complement rule. These are among the most commonly used discretization mechanisms in binary metaheuristic optimization (Lemus-Romani et al., 2023).

For the standard rule, the updated binary position is determined by comparing the value of the transfer function T(x) with a uniformly distributed random number r=rand [0,1]. The binary update is defined as in Eq. 19:

$$x_i^{t+1} = \begin{cases} 0, & r < TF(x_i^t) \\ 1, & \text{otherwise} \end{cases} \qquad (19)$$

For the complement rule, the current binary value is flipped when the generated random number is smaller than the transfer-function output. This update can be written as in Eq. 20:

$$x_i^{t+1} = \begin{cases} x_i^{t-1} & r < TF(x_i^t) \\ x_i^t & \text{otherwise} \end{cases} \qquad (20)$$

Here $x_i(t)$ shows the position of the agent of each of the three metaheuristics, i is the order of the agent in the population, t is the current iteration, and TF(.) is the selected transfer function. In our case, the standart rule is applied on S-shaped and X-shaped, meanwhile the complement method is applied on V-shaped, U-shaped, and Q-shaped TF.

Dataset and classification model

The experiments are conducted on the gallstone dataset, publicly available in the UC Irvine Machine Learning Repository (Esen, et al., 2024). The dataset contains 319 instances collected from the Internal Medicine Outpatient Clinic of Ankara VM Medical Park Hospital. Each row is characterized by 38 numerical variables and one binary variable denoting the absence or presence of gallstones. The dataset is complete and contains no missing values; therefore, no imputation procedure is required. In addition, the categorical and binary variables are already encoded, which eliminates the need for additional preprocessing. The class distribution is proportional, consisting of 161 patients who have gallstones and 158 healthy control groups. To evaluate the selected feature subsets, a k-nearest neighbors classifier with Euclidean distance and (k = 5) is employed. In order to reduce overfitting and obtain a more reliable estimate of classification performance, 5-fold cross-validation is used in the fitness evaluation.

All algorithms are executed 20 times using different random seeds in order to reduce the effect of stochastic variability. Overall final published outcomes represent the average values from these 20 separate tests.

The experiments are performed in RStudio 2023.06.0 on a computer equipped with an Intel Core i5-8365U processor and 16 GB RAM. For all algorithms, the population size is set to 6 and the maximum number of iterations is 100. The number of populations was chosen 6 for the purpose of comparability between algorithms, and for the reason of computational efficiency in multiple independent runs. The dimensionality of the search space is equal to the total number of features in the dataset.

The fitness function used in this study is a well-known one that can balance classification performance and subset size (Eq. 21):

$$F = 0.99 \cdot CE + 0.01 \cdot \frac{\text{selected features}}{\text{total features}} \qquad (21)$$

where (CE) illustrates the classification error. The optimization objective is to minimize the fitness value, thereby reducing classification error while also encouraging smaller feature subsets.

The performance of the binary metaheuristic variants is assessed using the following indicators:

- mean number of selected features,
- mean execution time,
- mean accuracy,
- mean fitness value,
- sensitivity,
- F-score.

Accuracy measures the proportion of correctly classified instances among all predictions. Sensitivity reflects the proportion of actual positive cases that are correctly identified. The F-score is another metrics that balances the evaluation of precision and recall through their harmonic mean.

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue VIII August 2026

All binary metaheuristic variations were assessed under the same experimental conditions, including the population size, number of iterations, classifier, fitness function, and cross-validation to provide a fair comparison. Thus, the observed performance differences can be attributed mainly to the interaction between the transfer functions and the search dynamics of the corresponding metaheuristic algorithms.

RESULTS AND DISCUSSION

Tables 1, 2, and 3 (see Appendix) summarize the experimental results obtained for the three binary metaheuristic algorithms, namely BALO, BTLBO, and BSSA, combined with the selected transfer functions. The reported performance indicators include the average number of selected features (nsf), average fitness (cost), accuracy (acc), sensitivity (Sen), F-score (F1), and execution time (time in min) over 20 independent runs.

Results for BALO

Table 1 presents the results obtained with the Binary Ant Lion Optimizer. Among all tested transfer functions, the V-shaped family shows particularly competitive performance. The smallest average number of selected features is obtained by V1, with 12.35 selected features, while the V-shaped transfer functions yield the smallest feature subsets on average (15.26).

About the fitness value, where lower values signify superior solutions, V2 attains the optimal outcome, with an average fitness of around 0.226. This transfer function also achieves the highest classification accuracy, with 77.5%. In terms of sensitivity and F-score, the best performance is observed for V3, which achieves approximately 70.85% sensitivity and 75.49% F-score. Regarding computational efficiency, the X-shaped transfer functions results with the shortest execution times, with X1 and X2 requiring 1.96 and 1.34 min, respectively.

Overall, for BALO, the V-shaped transfer functions provide the most favourable trade-off between predictive performance and feature subset size, whereas the X-shaped functions are advantageous in terms of execution time.

Results for BTLBO

Table 2 summarizes the results of the Binary Teaching–Learning-Based Optimization algorithm. In contrast to BALO, BTLBO yields remarkably small feature subsets for several transfer functions. The smallest average number of selected features is obtained by V1, with only 2.95 selected features, while the V-shaped family again produces the lowest average subset size overall (3.76).

The best fitness value is achieved by Q3, with an average of 0.2235. This combination also provides the highest classification accuracy, equal to 77.58%. Furthermore, Q3 yields the best sensitivity and F-score, reaching 73.19% and 76.27%, respectively. The most efficient variant, measured in terms of computational time, is achieved with X2, necessitating an average of 15.7 minutes.

The results demonstrate that BTLBO is highly effective when utilized with Q-shaped transfer functions, particularly Q3, which provides the optimal balance of prediction accuracy, feature reduction, and fitness.

Results for BSSA

The results obtained with the Binary Salp Swarm Algorithm are reported in Table 3. The smallest number of selected features is produced by V2, with 4.05 features on average, while the V-shaped transfer functions again yield the smallest subsets overall, with an average of 5.238 selected features.

However, the best fitness value for BSSA is achieved by U3, with an average fitness of approximately 0.2375. The same transfer function also reaches the highest classification accuracy, equal to 75.84%, and the highest F-score, approximately 74.41%. The highest sensitivity values are obtained by U2 and U3, both close to 70%, with only minor differences between them. The shortest execution time is observed for S2, with an average of

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue VIII August 2026

9.88 min.

Therefore, for BSSA, the U-shaped transfer functions appear to be the most effective in terms of predictive performance, whereas the V-shaped functions are more suitable for stronger feature reduction.

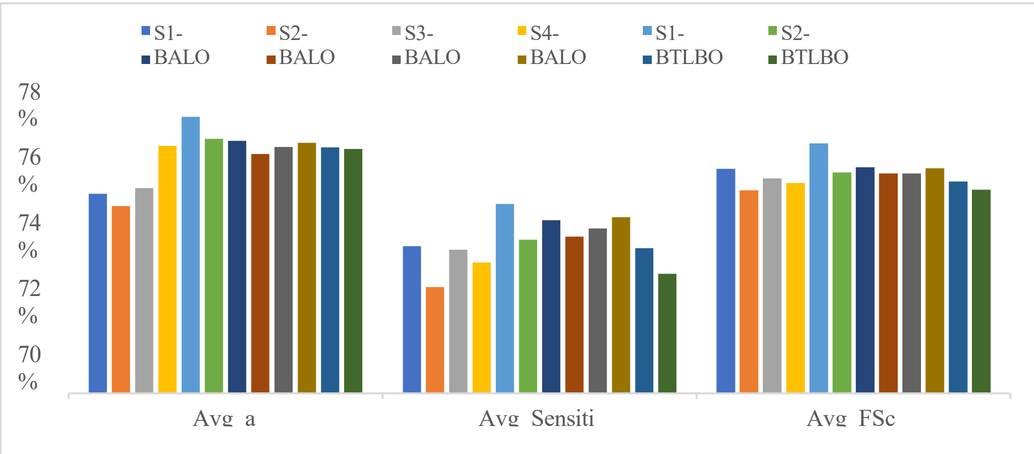
Comparison across transfer function families

The purpose of the comparative analysis is to examine how the behaviour of transfer functions varies across the three metaheuristic algorithms and to identify the most effective combinations with respect to the selected performance indicators.

S-shaped transfer function

Fig.1 presents the findings integrating accuracy, sensitivity, and F-Score. S1-BTLBO demonstrates superior results regarding the three metrics, respectively 76.16%, 70.43%, and 74.43%. This result indicates that BTLBO is more effective than BALO and BSSA when coupled with S-shaped transfer functions.

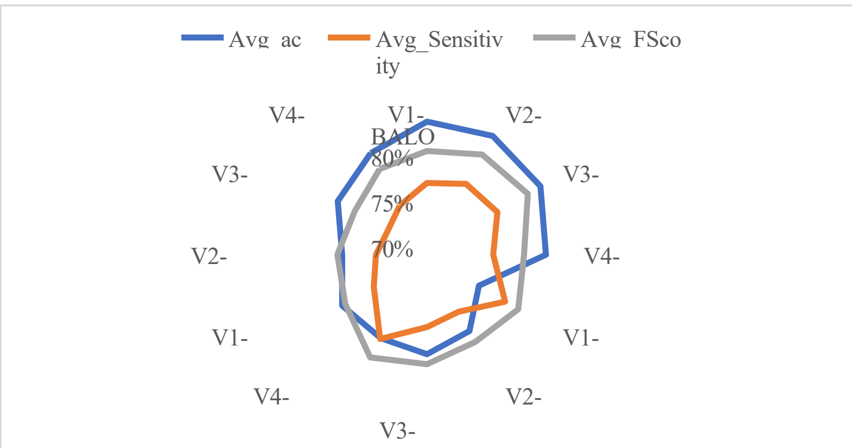
Figure. 1. Comparative performance of S-shaped transfer functions



V-shaped transfer function

Fig. 2 shows the results obtained with the V-shaped transfer functions. The best combinations are V2-BALO and V3-BALO, which achieve superior predictive performance compared with the corresponding BTLBO and BSSA variants. In particular, V2-BALO reaches approximately 77.53% accuracy, 70.43% sensitivity, and 74.72% F-score, while V3-BALO achieves 77.42% accuracy, 70.85% sensitivity, and 75.49% F-score. These findings confirm the strong compatibility between the V-shaped family and the BALO algorithm.

Figure. 2. Results of the metrics regarding V-shaped TF



U-shaped transfer function

Figure 3 illustrates that the optimal outcomes among the U-shaped transfer functions are achieved by U2-BTLBO and U3-BTLBO. These combinations generate performance indicators corresponding to approximately 77% accuracy, 73% sensitivity, and 76% F-score. This suggests that the U-shaped transfer functions are particularly effective when integrated with BTLBO.

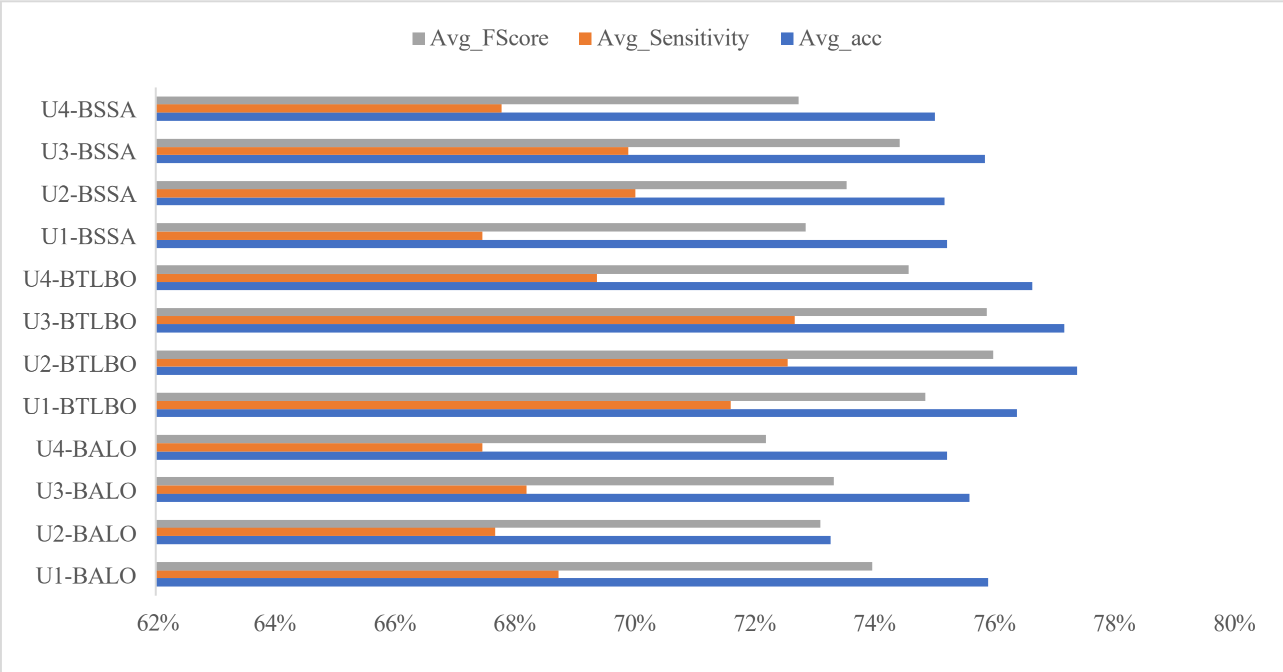
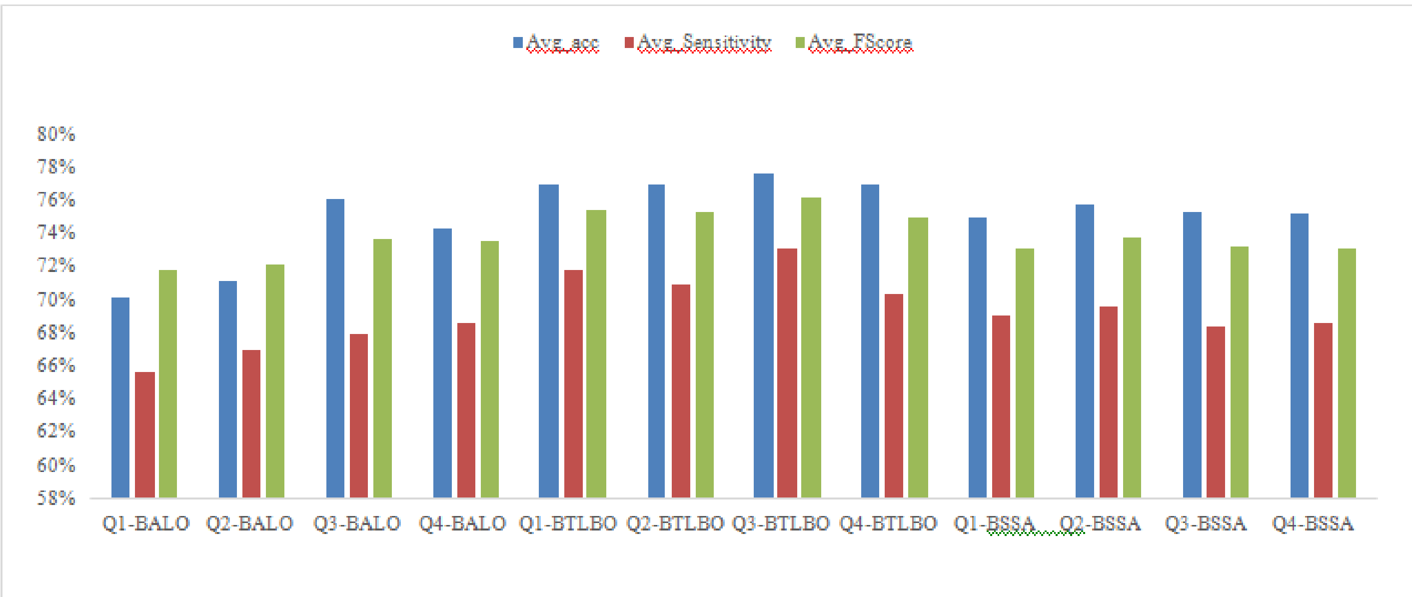


Figure. 3. Results of the metrics regarding U-shaped TF

Q-shaped transfer function

Figure 4 presents the results for the Q-shaped transfer functions. Among all tested combinations, Q3-BTLBO clearly provides the strongest performance, with 77.58% accuracy, 73.19% sensitivity, and 76.27% F-score. In general, the Q-shaped transfer functions integrated with BTLBO usually gives stable results, with all three measures over 70%. This indicates that the Q-shaped family is highly suitable for the binary TLBO framework.

Figure. 4. Results of the metrics regarding Q-shaped TF



X-shaped transfer function

The results for the X-shaped transfer functions are shown in Figure 5. The highest accuracy is obtained by X1-BTLBO, with 75.37%, whereas X2-BTLBO achieves the best sensitivity and F-score, approximately 70% and 76%, respectively. Although the X-shaped transfer functions do not outperform the best V-shaped or Q-shaped combinations in predictive accuracy, they show competitive performance and, as noted earlier, are associated with lower execution times.

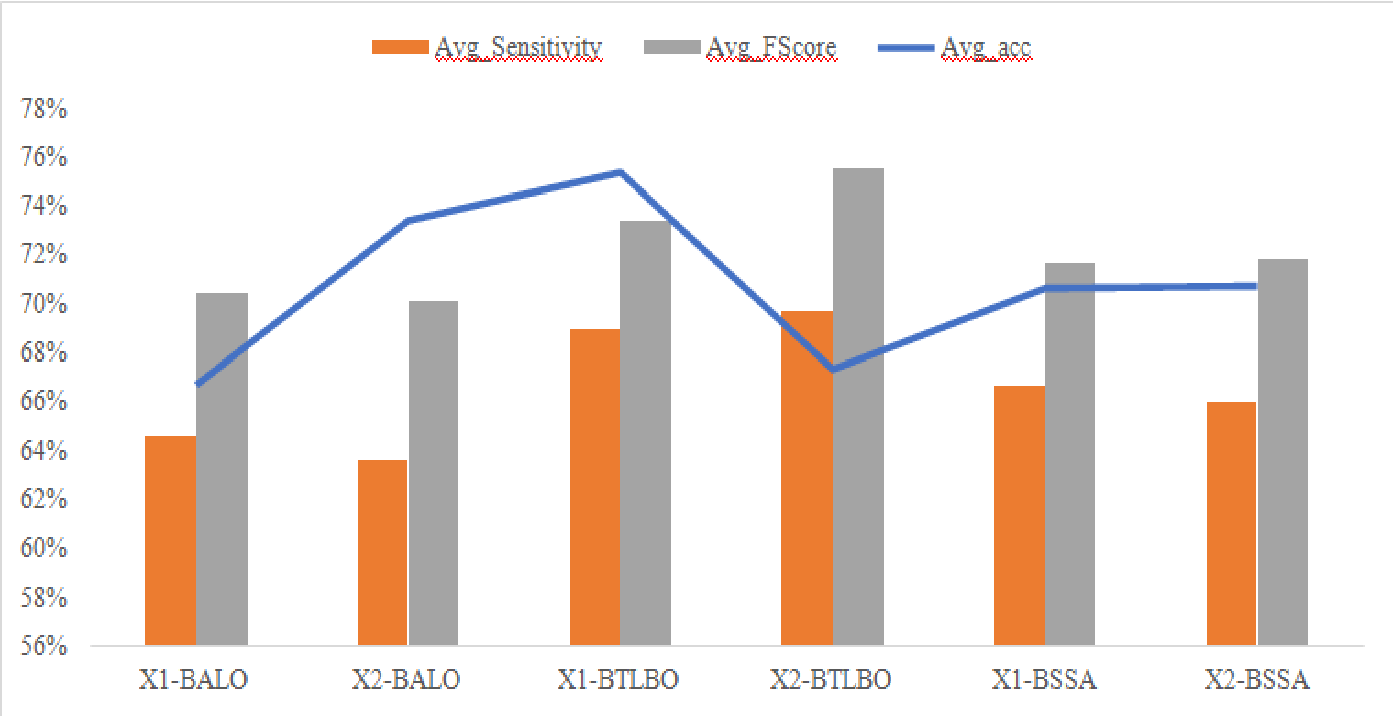


Figure. 5. Results of the metrics regarding X-shaped TF

Best performing combinations

To facilitate a direct comparison between the three binary metaheuristics, the best-performing combination for each algorithm is summarized in Table 4 based on the highest level of accuracy.

Table 4. The best combination results for each metaheuristic algorithm

TF-MHA	NSF	COST	ACC	SEN	F1	TIME
V2_BALO	13.65	0.23	78%	70%	75%	17.8
Q3-BTLBO	5.8	0.22	78%	73%	76%	24.67
U3-BSSA	11.35	0.24	76%	70%	74%	14.81

The results show that both V2-BALO and Q3-BTLBO achieve the highest classification accuracy, approximately 78%. However, Q3-BTLBO provides a smaller feature subset, a lower fitness value, and higher sensitivity and F-score than the competing alternatives. Although this combination requires more computational time, it offers the best overall predictive performance and feature selection quality. Therefore, Q3-BTLBO can be regarded as the most effective combination among all evaluated transfer function–metaheuristic pairs.

Best performing combinations

Figure 6 summarizes the average execution time for each transfer function family. The X-shaped transfer functions yield the shortest average execution time, approximately 11.32 min, whereas the Q-shaped family

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue VIII August 2026

requires the longest computational time, approximately 26.08 min on average.

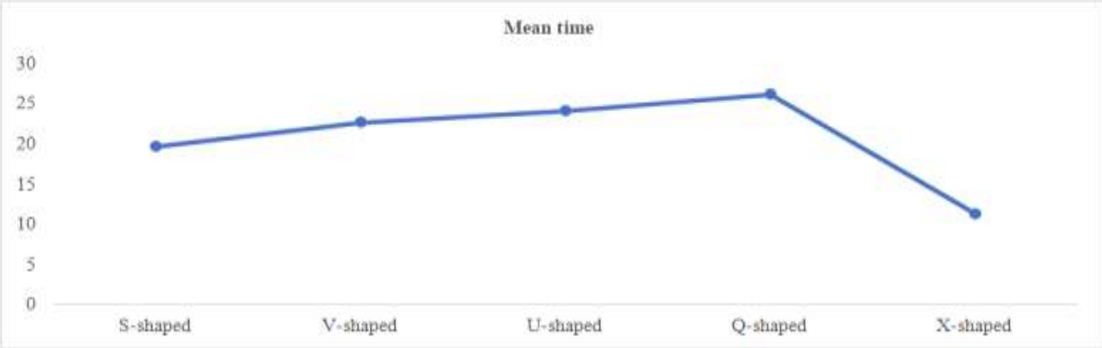


Figure. 6. Average execution time by TF family

The V-shaped transfer functions provide the smallest feature subsets, averaging 8.09 selected features, whereas the X-shaped functions produce the biggest subsets, averaging 32.58 features (see Figure 7). These findings reveal a clear trade-off: V-shaped functions are more effective for aggressive feature reduction, whereas X-shaped functions are computationally efficient but less selective.

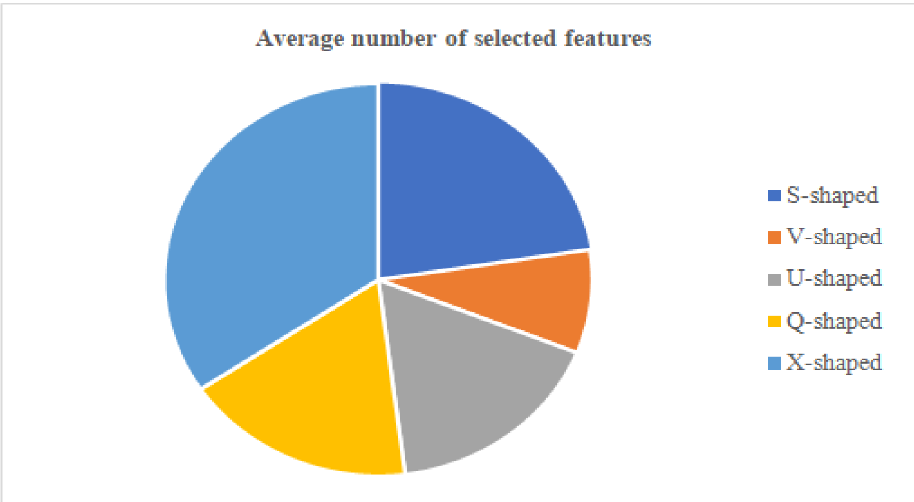


Figure. 7. Average number of selected features by TF family

Overall, the results demonstrate that the influence of transfer functions is substantial and depends on the interaction with the underlying metaheuristic algorithm. No single transfer function family is universally optimal across all algorithms and criteria; however, Q3-BTLBO and V2-BALO emerge as the strongest combinations for the considered gallstone prediction task.

CONCLUSIONS

This study investigated the impact of transfer functions on the efficacy of binary metaheuristic algorithms for feature selection in predicting gallstone disease. Five families of transfer functions—S-shaped, V-shaped, U-shaped, Q-shaped, and X-shaped—were combined with three binary metaheuristic algorithms, namely BALO, BTLBO, and BSSA, and evaluated using a medical dataset with 38 features.

The results confirm that the choice of transfer function has a substantial impact on the predictive performance, feature subset size, fitness value, and computational cost of binary metaheuristic optimization. Considerable variation was observed across the tested families. In particular, the accuracy values ranged approximately from 70% to 76% for S-shaped transfer functions, from 68% to 78% for V-shaped ones, from 73% to 77% for U-shaped ones, from 70% to 78% for Q-shaped ones, and from 67% to 75% for X-shaped ones.

Among the evaluated metaheuristics, BTLBO had the most effective predictive performance, obtaining the best average accuracy, sensitivity, and F-score across various transfer function families. The best individual results were obtained by Q3-BTLBO and V2-BALO, both reaching an accuracy of approximately 78%. Nevertheless, Q3-BTLBO exhibited the optimal overall equilibrium, incorporating superior accuracy with reduced fitness, smaller feature subsets, and improved sensitivity and F-score. By contrast, U3-BSSA showed competitive performance within the BSSA framework, while X-shaped transfer functions demonstrated the lowest execution time.

In this study, the Q3 transfer function combined with BTLBO proved to be the most effective overall solution for the considered medical prediction task.

The findings suggest that transfer functions should not be treated as interchangeable components in the binarization process. Their effect is algorithm-dependent, and selecting an appropriate transfer function can significantly improve feature selection performance.

Future work may extend the analysis by including additional recently proposed transfer functions, alternative discretization strategies such as elitist and roulette-elitist rules, and further benchmark datasets from both medical and non-medical domains. Another promising direction is the statistical validation of the observed differences using non-parametric significance tests.

REFERENCES

1. Martí, R., Sevaux, M., & Sörensen, K. (2025) 50 years of metaheuristics, European Journal of Operational Research, 321(2), 345-362, doi: 10.1016/j.ejor.2024.04.004
2. Kennedy, J. & Eberhart, C. R, (1997). A discrete binary version of the particle swarm algorithm, in 1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation, Orlando, FL, USA, doi: 10.1109/ICSMC.1997.637339
3. Thaher, T., Mafarja, M., Turabieh, H., Castillo, A. P., Faris, H., & Aljarah, I. (2021). Teaching Learning-Based Optimization with Evolutionary Binarization Schemes for Tackling Feature Selection Problems, IEEE Access, 9, 41082-41103, doi: 10.1109/ACCESS.2021.3064799
4. Becerra-Rozas, M., Lemus-Romani, J., Cisternas-Caneo, F., Crawford, B., Soto, R., Astorga, G., Castro, C., & García, J. (2023). Continuous Metaheuristics for Binary Optimization Problems: An Updated Systematic Literature Review, Mathematics, 11(1), 1-32, https://doi.org/10.3390/math11010129
5. Mirjalili, S., & Lewis, A., (2013). S-shaped versus V-shaped transfer functions for binary Particle Swarm Optimization, Swarm and Evolutionary Computation, Swarm and Evolutionary Computation, 9, 1-14, https://doi.org/10.3390/math11010129
6. Beheshti, Z. (2021), A novel x-shaped binary particle swarm optimization, Soft Computing, 25, 3013-3042, doi: 10.1007/s00500-020-05360-2
7. Mirjalili, S. Z. , Zhang, H., Mirjalili, S., Chalup, S. & Noman, N. (2020). A Novel U-Shaped Transfer Function for Binary Particle Swarm Optimisation, Advances in Intelligent Systems and Computing, 1138, Singapore, Springer, 241–259, doi: 10.1007/978-981-15-3290-0_19
8. Guo, Sha-sha., Wang, Jie-sheng., & Guo, Meng-wei. (2020). Z-Shaped Transfer Functions for Binary Particle Swarm Optimization Algorithm, Computational Intelligence and Neuroscience, 6502807, 1-21, https://doi.org/10.1155/2020/6502807
9. Wang, L., Wang, X., Fu, J., & Zhen, L. (2008). A Novel Probability Binary Particle Swarm Optimization Algorithm and Its Application, Journal of Software, 3(9), 28-35, doi: 10.4304/jsw.3.9.28-35
10. Too, J., Abdullah, R. A. & Saad, M. N. (2019). A New Quadratic Binary Harris Hawk Optimization for Feature Selection, Electronics, 8, 1-27, https://doi.org/10.3390/electronics8101130
11. Rashedi, E., Nezamabadi-Pour, H., & Saryazdi, S. (2010). BGSA: binary gravitational search algorithm", Natural Computing, 9, 727–745, https://doi.org/10.1007/s11047-009-9175-3
12. Emary, E. M., Zawbaa, M. H., & Hassanien, E. A. (2016). Binary grey wolf optimization approaches for feature selection, Neurocomputing, 172, 371-381, https://doi.org/10.1016/j.neucom.2015.06.083

13. Mafarja, M., Eleyan, D., Abdullah, S., & Mirjalili, S. (2017). S-Shaped vs. V-Shaped Transfer Functions for Ant Lion Optimization Algorithm in Feature Selection Problem ,in ICFNDS '17: Proceedings of the International Conference on Future Networks and Distributed Systems, 21, 1-7, https://doi.org/10.1145/3102304.310232.
14. Naka, E. (2023). A Competitive Parkinson-Based Binary Volleyball Premier League Metaheuristic Algorithm for Feature Selection, Cybernetics and Information Technologies, 23(4), 91-109, https://doi.org/10.2478/cait-2023-0038
15. Nadimi-Shahraki, M. H., Banaie-Dezfouli, M., Zamani, H., Taghian, S., & Mirjalili, S. (2021). B-MFO: A Binary Moth-Flame Optimization for Feature Selection from Medical Datasets, Computers, 10, 1-18, https://doi.org/10.3390/computers10110136
16. Kumar, V., Kumar, D., Kaur, M., Singh, D., Idris, A. S., & Alshazly, H. (2021). A Novel Binary Seagull Optimizer and Its Application to Feature Selection Problem, IEEE Access, 9, 103481-103496, doi: 10.1109/ACCESS.2021.3098642
17. Bas, E., (2024). BinDMO: a new Binary Dwarf Mongoose Optimization algorithm on based Z-shaped, U-shaped, and taper-shaped transfer functions for CEC-2017 benchmarks, Neural Computing and Applications, 36, 6903–6935, https://doi.org/10.1007/s00521-024-09436-0
18. Beşkirli, A. (2025), A Transfer Function-Based Binary Version of Improved Pied Kingfisher Optimizer for Solving the Uncapacitated Facility Location Problem, Biomimetics, 10(8), 1-30, https://doi.org/10.3390/biomimetics10080526
19. Kristiyanti, D. A., Sitanggang, I. S., Annisa, Nurdiati, S. (2023). Feature Selection Using New Version of V-Shaped Transfer Function for Salp Swarm Algorithm in Sentiment Analysis, Computation, 11(3), https://doi.org/10.3390/computation11030056
20. Heris, M. K. (2015), Teaching-Learning-based Optimization in MATLAB. [Online]. January 25 2021, Available: https://yarpiz.com/83/ypea111-teaching-learning-based-optimization.
21. Too, J. Salp Swarm Algorithm for Feature Selection, [Online]. January 21 2021, Available: https://www.mathworks.com/matlabcentral/fileexchange/78913-salp-swarm-algorithm-for-feature-selection?s_tid=prof_contriblnk.
22. Emary, E., Zawbaa, H. M., & Hassanien, A. E. (2016), Binary ant lion approaches for feature selection, Neurocomputing, 213, 54-65, https://doi.org/10.1016/j.neucom.2016.03.101
23. Rao, R. V., Savsani, V. J., & Vakharia, D. P. (2011), Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems, Computer-Aided Design, 43(3), 303-315, doi: 10.1016/j.cad.2010.12.015
24. Mirjalili, S. (2015), The Ant Lion Optimizer, Advances in Engineering Software, 83, 80-98, doi: 10.1016/j.advengsoft.2015.01.010
25. 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
26. Lemus-Romani, J., Crawford, B., Cisternas-Caneo, F., Soto, R., & Becerra-Rozas, M., (2023). Binarization of Metaheuristics: Is the Transfer Function Really Important? Electronics, 8(5), 1-47, https://doi.org/10.3390/biomimetics8050400
27. Esen, İ., Arslan, H., Esen, S. A., Gülşen, M., Kültekin, N., & Özdemir, O. (2024). Early prediction of gallstone disease with a machine learning-based method from bioimpedance and laboratory data, Medicine, 103(8), doi: 10.1097/MD.0000000000037258.

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue VIII August 2026

APPENDIX

Table 1. Average performance metrics for binary ALO

TF-MHA	NSF	COST	ACC	SEN	F1	TIME
S1	33.9	0.2564	71.11%	67.66%	72.74%	17.02
S 2	33.2	0.2635	70.32%	65.00%	71.34%	17.74
S 3	34.15	0.2638	71.47%	67.45%	72.12%	12.11
S4	32.45	0.2633	74.26%	66.60%	71.81%	6.62
V 1	12.35	0.2309	77.00%	69.15%	73.24%	15.71
V2	13.65	0.226	77.53%	70.43%	74.72%	17.8
V 3	16.55	0.2279	77.42%	70.85%	75.49%	15.22
V4	18.5	0.244	75.84%	68.83%	72.94%	21.3
U 1	26.5	0.243	75.89%	68.72%	73.96%	24.17
U2	30.75	0.2509	73.26%	67.66%	73.09%	16.97
U 3	27.75	0.2491	75.58%	68.19%	73.31%	23.69
U4	26.9	0.2525	75.21%	67.45%	72.18%	11.42
X 1	35.55	0.2709	66.68%	64.57%	70.42%	1.96
X2	35.95	0.2726	73.42%	63.62%	70.13%	1.34
Q 1	33.05	0.2557	70.21%	65.64%	71.78%	6.38
Q2	33	0.2505	71.21%	67.02%	72.17%	7.07
Q 3	24.3	0.2424	76.16%	67.98%	73.65%	26.2
Q4	29.2	0.2395	74.42%	68.62%	73.60%	13.28

Table 2. Average performance metrics for binary TLBO

TF-MHA	NSF	COST	ACC	SEN	F1	TIME
S1	13.25	0.2395	76.16%	70.43%	74.43%	47.44
S2	15.65	0.2542	74.74%	68.09%	72.53%	34.91
S 3	17.4	0.2562	74.58%	69.36%	72.87%	25.2
S4	17.7	0.2584	73.74%	68.30%	72.47%	25.5
V 1	2.95	0.2462	68.00%	72.02%	74.01%	17.52
V2	3.45	0.2474	71.26%	68.40%	72.90%	37.62
V 3	4.1	0.2387	72.74%	69.26%	74.03%	42.11
V4	4.55	0.2346	72.32%	72.45%	75.19%	48.61
U 1	7.25	0.2359	76.37%	71.60%	74.84%	44.93
U2	6.4	0.2257	77.37%	72.55%	75.97%	57.92
U 3	7.65	0.2282	77.16%	72.66%	75.87%	26.8
U4	8.05	0.2335	76.63%	69.36%	74.56%	24.53
X 1	25.75	0.2506	75.37%	68.94%	73.39%	25.28
X2	34.35	0.2284	67.32%	69.68%	75.52%	15.7
Q 1	7.5	0.2302	76.95%	71.81%	75.41%	69.6
Q2	7.6	0.2292	77.05%	70.96%	75.32%	63.72
Q 3	5.8	0.2235	77.58%	73.19%	76.27%	24.67
Q4	10.25	0.2309	76.95%	70.43%	75.04%	27.79

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue VIII August 2026

Table 3. Average performance metrics for binary SSA

TF-MHA	NSF	COST	ACC	SEN	F1	TIME
S1	11.95	0.2585	74.21%	68.83%	72.46%	13.07
S2	13.35	0.2562	74.47%	69.57%	72.78%	9.88
S 3	16.35	0.2601	74.16%	67.55%	71.92%	11.62
S4	16.45	0.2612	74.05%	65.85%	71.37%	14.02
V 1	4.3	0.2528	73.05%	68.19%	72.56%	17.5
V2	4.05	0.2548	71.32%	66.81%	71.94%	16.95
V3	5.75	0.2615	73.74%	66.06%	71.14%	9.981
V4	6.85	0.2493	75.00%	67.23%	72.59%	10.66
U1	12.55	0.2487	75.21%	67.45%	72.85%	14.31
U2	12.75	0.2493	75.16%	70.00%	73.53%	14.27
U3	11.35	0.2375	75.84%	69.89%	74.41%	14.81
U4	14.2	0.2512	75.00%	67.77%	72.73%	14.25
X1	32.25	0.2643	70.63%	66.70%	71.71%	10.75
X2	31.6	0.2584	70.74%	65.96%	71.85%	12.86
Q1	12.4	0.2502	75.05%	69.15%	73.14%	14.42
Q2	11.35	0.2427	75.79%	69.68%	73.86%	14.54
Q 3	10.6	0.2466	75.37%	68.40%	73.25%	14.5
Q 4	14.45	0.2492	75.21%	68.62%	73.13%	30.75