DOI: 10.24996/ijs.2026.67.3.32

ISSN: 0067-2904

ICBA: Integrating Chaotic Maps into the Bat Algorithm for Enhanced Feature Selection

Ali Abdulkadhim Taher [1]*, Saif Ali Abd Alradha Alsaidi [2], Mohammed Abomaali [3]

[1] College of Arts ,Wasit University, Wasit, Iraq

[2] College of Computer Science and Information Technology, Wasit University, Wasit, Iraq

[3] College of Information Technology Engineering, Al-Zahraa University for Women, Karbala, Iraq

Received: 7/4/2024 Accepted: 10/3/2025 Published: 30/3/2026

Abstract

Feature selection is an effective way to decrease dataset dimensions and increase classification accuracy. However, feature selection is a complex and challenging procedure requiring a highly efficient algorithm. In this enhanced bat algorithm (ICBA), the issue of feature selection is initially conceptualized and subsequently transformed into a fitness function, which assesses the quality of feature subsets based on how well they improve classification performance. We also propose an ICBA to address the issue of feature selection. To enhance the BA and expand its applicability to feature selection issues, we integrated a chaotic map into the BA. Ultimately The proposed algorithm ICBA is benchmarked against binary PSO (BPSO), Binary dragonfly algorithm (BDA), Binary grey wolf optimization approach (BGWO), Binary bat algorithm (BBA), and enhanced binary bat algorithm (EBBA). To evaluate these algorithms, five datasets were sourced from the UC Irvine Machine Learning Repository. The experimental findings reveal that the ICBA algorithm outperforms other comparative algorithms across all datasets. In the Breastcancer dataset, the accuracy rate for ICBA was (0.9941) compared to the closest algorithm's (0.9786). In the BreastEW dataset, the accuracy rate for ICBA was (0.9857) compared to the closest algorithm's (0.9614). In the Congress dataset, the accuracy rate for ICBA was (0.9893), whereas it was (0.9793) for the nearest algorithm. In the SpectEW dataset, the accuracy rate for ICBA was (0.8691) compared to the nearest algorithm, where it was (0.7407). In the tic-tac-toe dataset, the accuracy rate was (0.9688), while the closest algorithm was (0.8521).

Keywords: Bat Algorithm; Feature Selection; Optimization; Chaotic Map.

دمج دالة خريطة الفوضى في خوارزمية الخفاش لتحسين اختيار الميزات

علي عبد الكاظم طاهر [1]*, سيف علي عبد الرضا [2], محمد فائز عيسى [3]

[1]كلية الآداب، جامعة واسط، واسط، العراق

[2]كلية علوم الحاسوب وتكنلوجيا المعلومات، جامعة واسط، واسط، العراق

[3]كلية هندسة تكنولوجيا المعلومات، جامعة الزهراء للبنات، كربلاء، العراق

* Email : ataher@uowasit.edu.iq

الخلاصة

اختيار الميزات هو طريقة فعالة لتقليل أبعاد مجموعة البيانات وزيادة دقة التصنيف. ومع ذلك، فإن اختيار الميزات هو عملية معقدة وصعبة تتطلب خوارزمية عالية الكفاءة. في خوارزمية الخفاش المحسنة (ICBA)، يتم أولًا تصور مشكلة اختيار الميزات ثم تحويلها إلى دالة لياقة، حيث تقوم هذه الدالة بتقييم جودة مجموعات الميزات بناءً على مدى تحسينها لأداء التصنيف. بعد ذلك، قمنا باقتراح خوارزمية ICBA لمعالجة مشكلة اختيار الميزات، ولتحسين خوارزمية الخفاش (BA) وتوسيع تطبيقها على مشكلات اختيار الميزات، قمنا بدمج خريطة فوضوية داخل الخوارزمية. في النهاية، تم مقارنة الخوارزمية المقترحة ICBA مع عدة خوارزميات أخرى، وهي: خوارزمية سرب الجسيمات الثنائية(BPSO) ، خوارزمية اليعسوب الثنائية(BDA) ، نهج تحسين الذئب الرمادي الثنائي(BGWO) ، خوارزمية الخفاش الثنائية(BBA) ، والخوارزمية الثنائية المحسنة للخفاش .(EBBA) لتقييم أداء هذه الخوارزميات، تم استخدام خمس مجموعات بيانات مأخوذة من مستودع *UC Irvine*لتعلم الآلة، وأظهرت النتائج التجريبية أن خوارزمية ICBA تتفوق على جميع الخوارزميات المقارنة في جميع مجموعات البيانات. في مجموعة بيانات **Breastcancer**، بلغ معدل دقة ICBA (0.9941)مقارنة بأقرب خوارزمية التي حققت (0.9786)، وفي مجموعة بيانات **BreastEW**، كان معدل الدقة لـ ICBA (0.9857) مقارنة بأقرب خوارزمية التي حققت (0.9614). أما في مجموعة بيانات **Congress**، فقد بلغ معدل الدقة لـICBA (0.9893) ، في حين كان (0.9793) لأقرب خوارزمية، بينما في مجموعة بيانات **SpectEW**، بلغ معدل الدقة لـ ICBA (0.8691) مقارنة بأقرب خوارزمية التي حققت (0.7407). أخيرًا، في مجموعة بيانات **tic-tac-toe**، بلغ معدل الدقة (0.9688)، بينما كانت أقرب خوارزمية تحقق.(0.8521)

1. Introduction

Over recent years, there has been a significant increase in the volume of high-dimensional data that is available and accessible online. Consequently, machine learning algorithms have difficulty dealing with enormous amounts of data. Data must be pre-processed to use machine learning technology effectively [1][2]. Strategies for selecting features prove invaluable in supervised learning, aiming to optimize particular functions to boost predictive accuracy by identifying and selecting features pertinent to a specific class label [3]. Selection is a method to pinpoint independent features and discard unnecessary ones from the dataset [4][5]. Metaheuristic optimization techniques can explore the entirety of the search space and employ a global search strategy, significantly enhancing the ability to discover high-quality solutions within a feasible time frame [6].

The bat algorithm (BA) represents promising examples of swarm intelligence, having shown superior performance and effectiveness over other algorithms in specific applications. Inspired by the social behavior of bats, the bat algorithm (BA) stands out as a contemporary metaheuristic technique. Characterized by its straightforward equations, BA finds versatile applications across various domains, including classification, feature selection, data mining, and scheduling, among others, due to its simplicity and adaptability[7]. The bat algorithm, a recent innovation by Yang, draws inspiration from bats' echolocation capabilities, enabling them to sense distances and differentiate between prey and background obstacles [8]. The bat algorithm, along with its various adaptations, has found widespread application in numerous computing scenarios [9].

Chaotic maps are adeptly utilized to enhance the efficacy of metaheuristic approaches by facilitating escape from local optima and accelerating the convergence rate [10][11]. The sinusoidal chaotic map is employed to ascertain variable values of the step size (α) parameter using a local search area [12].

Our study incorporates a Chaotic map with the Bat Algorithm specifically for feature selection, a method we've named ICBA. This novel approach merges the strengths of the hybrid Bat Algorithm (BA) and a chaotic map to create a more effective algorithm. The BA is utilized for solution generation and plays a crucial role in improving the quality of the ultimate solution. Additionally, the sinusoidal chaotic map is applied to update the position of the bats, further enhancing the algorithm's performance. Simulations utilizing openly available datasets from the UC Irvine machine learning repository were executed to validate the problem-solving prowess of the proposed ICBA method. For a comprehensive evaluation, ICBA was benchmarked against several standard algorithms. Subsequently, we demonstrate the efficacy of the improvements incorporated into ICBA, highlighting its enhanced performance.

The structure of this study is outlined as follows: Section 2 delves into an examination of pivotal works related to swarm intelligence algorithms, feature selection, and chaotic map methodologies. In Section 3, we introduce the Enhanced Bat Algorithm (ICBA) and elaborate on the factors that have been optimized. Section 4 is dedicated to presenting the outcomes of our simulations. Finally, Section 5 concludes this work.

2. Related Works

Metaheuristic algorithms have proven their efficacy across a wide range of applications. The challenge of feature selection, inherently a multi-objective optimization problem, strives to achieve a dual goal: maximizing classification accuracy while minimizing the number of selected features. To address this, numerous metaheuristic approaches have been employed, effectively tackling feature selection challenges, and a selection of these will be reviewed.
Furthermore, various heuristic algorithms, inspired by the mechanisms of biological and physical systems found in nature, have been introduced. These algorithms stand out as robust solutions for global optimization tasks, showcasing the innovative application of natural phenomena to solving complex computational problems.

Abdel-Basset, Mohamed, et al. [13] combined the Grey Wolf Optimizer algorithm with a two-phase mutation strategy to address the feature selection challenge. Ghanem, Waheed Ali HM, et al. [14] used the multi-objective BAT algorithm (MOBBAT) to develop an algorithm of a proficient wrapper approach-based feature selection. Anter, A. M., Azar, A. T., & Fouad, K. M. [15] suggested an intelligent hybrid technique that combines Chaos Theory, Rough Set Theory (RST), and the Binary Grey Wolf Optimization Algorithm (CBGWO). Li, An-Da, Bing Xue, and Mengjie Zhang. [16] proposed an improved version of the sticky binary PSO (ISBPSO) algorithm to boost evolutionary performance. The ISBPSO incorporates three innovative approaches built upon the sticky binary particle swarm optimization (SBPSO), a novel variation of the binary PSO that was recently introduced.

Jinghui Feng et al. [17] proposed an Enhanced Binary Bat Algorithm (EBBA) specifically tailored to tackle feature selection problems. To augment the capabilities of the Bat Algorithm (BA) and extend its suitability for feature selection challenges, they integrated a trio of sophisticated techniques into EBBA: a global search strategy based on Lévy flight, a technique to improve population diversity, and a chaos-based method to adjust. Eskandari, S., and M. Seifaddini [18] found that the minimal informative subsets are discovered using the binary bat algorithm (BBA). This allows for effectively considering many feature subsets during the redundancy analysis stage. Qasim, O. S., & Algamal, Z. Y. [4] suggested six different types of BA (BA-S and BA-V). A transfer function (TF) is utilized in each type to map the solutions from continuous space to discrete space. Anter, A. M., & Ali, M. [19]

suggested the hybrid crow search optimization algorithm(CFCSA)for feature selection in medical diagnostics, which combines chaos theory and fuzzy c-means. Tawhid, M. A., & Dsouza, K. B. [20] introduced a new hybrid binary version of the bat and an improved particle swarm optimization approach (HBBEPSO) To solve the issue of feature selection. Nakamura, Rodrigo YM, et al. [21] suggested a novel feature selection method, Bat Algorithm combined with Optimum-Path Forest classifier to find the best combination of features. Zhou, Xianjin, et al. [22] suggested integrating a new version of the BA algorithm with the ABC algorithm (IBA). Rauf, Hafiz Tayyab, et al. [23] suggested an improved version of BA to address the local minima issue and premature convergence of BA. The suggested variant used a Gaussian adaptive inertia weight to regulate each member of the swarm's velocity.

Remeseiro, Beatriz, and Veronica Bolon-Canedo. [24] reviewed the most recent feature selection techniques designed and used in medical problems. El-Kenawy, El-Sayed, and Marwa Eid. [25] introduced a hybrid methodology that merges the Gray Wolf Optimization (GWO) and Particle Swarm Optimization (PSO) algorithms. This innovative approach is designed to pinpoint essential features while discarding redundant ones and reducing complexity in the process. H. D. Praveena, et al. [26] introduced an improved artificial bee colony optimization algorithm for feature selection, followed by classification with a stacked autoencoder. A. Y. Mahdi and S. S. Yuhaniz [27] proposed an Improved Binary Sparrow Search Algorithm (IBSSA) for feature selection in clinical texts to enhance COVID-19 patient categorization. A. A. Hussein, et al. [28] suggested an algorithm combining a discrete grey wolf optimizer with Q-learning (DGWO-QL) for the green vehicle routing problem (GVRP), focusing on environmental impact and computational efficiency. A.N. N. Kumar, et al. [29] proposed an innovative approach that combines firefly and harmony search algorithms for energy-efficient and secure VM allocation in cloud data centers.

3- Methodology

3.1 Bat Algorithm

The Bat Algorithm (BA) is a globally optimized swarm intelligence system inspired by bats' echolocation skills. Specifically, bats navigate while flying at a random velocity (V_i) towards a random point (Xi) while adjusting their loudness (A_0), wavelength (λ), and frequency (f_{min}). These adjustments in wavelength, frequency, and pulse emission rate (r), which vary within the range of [0, 1], are autonomously made by the bats based on the proximity to their target. A comprehensive description of the mathematical model underpinning the BA is provided here. The frequency (f_i) of the ith bat in the nth iteration is defined as follows.

$$f_i = f_{min} + (f_{max} - f_{min}) \times \beta_i \quad (1)$$

In the formula, β represents a random number within the range of [0, 1], and f_{min} and f_{max} denote the minimum and maximum bounds for the frequencies across all bats.
Additionally, the following model can be used to predict the ith bat V_i's velocity:

$$\mathrm{V_i^t = V_i^{t-1} + (X_i^{t-1} - X^*)f_I} \quad (2)$$

Where X^* is the bat in the swarm with the greatest fitness function value.
Furthermore, the ith bat's update method is displayed as follows:

$$\mathrm{X_i^t = X_i^{t-1} + V_i^t} \quad (3)$$

Where X_i^t represents the ith bat's position in the tth iteration.
Furthermore, the Bat Algorithm (BA) enhances its search efficacy through random walks that occur locally. Specifically, the algorithm directs bats with the highest ranking in the swarm to undertake a search in the local area based on predefined most probable estimates, detailed as follows:

$$X^N = X^* + \varepsilon \times A^t \tag{4}$$

In this context, ϵ is a random variable with values ranging from $[-1,1]$, representing the bats' loudness at iteration t, and signifies the position of the bat newly generated following the walk.
Moreover, as the iterations progress, both the loudness (A_i) and the pulse emission rate (r_i) of the bats are updated by the following guidelines:

$$A_i^{t+1} = \alpha A_i^t \quad , \quad r_i^{t+1} = r_i^0[1 - \exp(\gamma t)] \tag{5}$$

where α is a parameter ranging from 0 to1, and $\gamma > 0$.
Utilizing such mathematical models, the principal steps of the Bat Algorithm (BA) can be succinctly summarized as follows:

Input:
- Population size N (number of bats).
- Max number of iterations *max_iter*.
- Loudness A_i, Pulse rate r_i, Frequency range $[f_{min}, f_{max}]$ (defined by Equations (1) and (2)).

Output:
- Optimal solution X^* with best fitness.

Steps

1.**Initialize Parameters and Population**:
- Generate an initial random population (bat swarm) $P=[X_1,X_2,\ldots,X_N]$ with initial positions and velocities for each bat.
- Randomly assign values for loudness A_i, pulse emission rate r_i, and frequency f_i for each bat based on the specified ranges.
- Calculate the initial fitness values for all bats in the swarm.

2.**Update Positions and Velocities**:
- For each bat i in the population, update its frequency f_i using Equation (1)
- Calculate new velocity V_i and update the position X_i for each bat using Equations (2) and (3)

3.**Local Search with Random Walk**:
- Generate a random number N_{rand} between 0 and 1.
- If $N_{rand} > r_i$ (pulse rate threshold), perform a random walk to explore the local neighborhood near the best solution X^*, as per Equation (5).

4. **Evaluate and Accept New Solutions**:
- Generate another random number N_{rand} between 0 and 1.
- If $N_{rand} > A_i$ (loudness threshold) and the fitness of the new solution X_{new} is better than the previous solution $f(X_{new})<f(X_i)$, update X_i with X_{new}.

5. **Adjust Loudness and Pulse Rate**:
- Gradually decrease loudness A_i and increase pulse rate r_i over iterations to enhance exploitation capabilities.

6. **Check Termination Condition**:
- Repeat Steps 2 to 5 until the termination criterion is met (e.g., reaching *max_iter* or achieving a satisfactory fitness level).

7. **Output the Best Solution**:
- Return the optimal solution X^*, which is the best solution found by the swarm.

3.2 Chaotic map

A mathematical model that displays chaotic behavior is called a chaotic map. A field of mathematics known as chaos theory examines dynamic, complicated systems that are extremely sensitive to their starting points. Chaotic maps are frequently employed to depict how a system changes over time.

The logistic map is a well-known chaotic map representing a basic population model, defined by its recurrence relation.

$$\mathrm{X_{n+1}} = \mathrm{r.X_n}(1 - \mathrm{X_n}) \tag{6}$$

Here, r is a parameter controlling the system's behavior, and Xn represents the population at time n. As the parameter r grows, the chaotic map exhibits intriguing and disorderly behavior. The system changes from ordered to chaotic behavior at specific levels of r.

Chaotic maps are utilized in various disciplines like computer science, physics, biology, and economics to understand dynamic system behavior and generate pseudorandom numbers in cryptography.

Chaotic behavior suggests a deterministic system highly sensitive to initial conditions, with even slight changes over time potentially causing radically different outcomes.

The power behind integrating chaotic maps into BA lies in exploiting the deterministic unpredictability of chaos to enhance the algorithm's adaptability, convergence speed, and overall accuracy in high-dimensional optimization tasks.

3.3 Proposed ICBA

In this section, the integrated chaotic maps into the Bat Algorithm (BA), namely, (ICBA) for feature selection problems. A hybrid proposed based on

Bat algorithms (BA) starts with an initial random population to select features, then a fitness function and a Random Forests classifier was used for training instances and clarifies how it was used for features that are picked.

The chaotic maps adjust the frequency of the bats' pulses, making them emit more frequently as the number of iterations increase, which is stated as follows:

$$C = r * a(1 - a) \tag{7}$$

$$f = 1 - \exp\,(-r * \mathrm{C}) \tag{8}$$

where a is a parameter ranging from 0 to 1, C a random number generated by Chaotic maps, and r represent pulse rates.

The accuracy rate is a common metric used to evaluate the performance of a classification algorithm. It is calculated as the ratio of correctly classified instances to a dataset's total number of instances. The equation is:

$$Accuracy\ Rate = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions} \times 100 \tag{9}$$

The steps of ICBA are summarized in Algorithm 1 as follows:

Algorithm 1:ICBA

Inputs:
- ***Max_iter*: Maximum number of iterations**
- **N_{dim} : Number of dimensions or features in the dataset**
- ***num_bats*: Population size (number of bats)**
- ***A*: Initial loudness**
- ***r*: Initial pulse rate**
- **Objective function *f(X)***: Fitness function to evaluate feature subset performance based on classification accuracy

Outputs:
- *best_solution*: Optimal feature subset
- *best_fitness*: Fitness value of the best solution

Solution Representation and Encoding:
- Each bat's position (solution) is represented as a binary vector of length N_{dim} , where each bit indicates whether a feature is selected (1) or not selected (0).
- Each cycle generates a new binary solution vector for each bat, where each feature has a probability of 0.5 to be selected or deselected, based on chaotic random numbers.

Steps

1. **Initialize Parameters and Population**:
- Set parameters *Max_iter*, N_{dim} , *num_bats*, *A*, and *r*.
- Initialize the bat population, where each bat has:
  - A binary position vector of size N_{dim} (randomly initialized).
  - Initial loudness, pulse rate, and frequency.

2. **Main Loop (Iterative Process)**:
- **For each iteration** from 1 to *Max_iter*:
  - **For each bat** in the population:

1. **Generate Chaotic Random Number**:
   - Use Equation (7) to generate a chaotic random number *X*, guiding random movement.
2. **Generate New Solution (Feature Subset)**:
   - For each feature, generate a new binary vector *new_solution*, setting each feature to 1 if $X < 0.5$ or 0 otherwise.
3. **Calculate Fitness of the New Solution**:
   - Calculate the fitness *new_fitness* of *new_solution* using the objective function *f(X),* based on the classifier's accuracy with this subset.
4. **Update Bat's Position if Solution is Better**:
   - If *new_fitness* is better than the bat's current fitness and $X< A$ (loudness threshold):
   - Update the bat's position to *new_solution*.
   - Set *bat['fitness']= new_fitness*.
5. **Update Bat Frequency**:
   - Adjust the frequency using Equation (8), which modifies the bat's movement pattern for the next iteration.

- **Update Best Solution if Necessary**:

1. If *new_fitness* is better than *best_fitness*:
   - **Update** *best_solution* **to** *new_solution* **and** *best_fitness* **to** *new_fitness*.

3. **Decrease Loudness and Increase Pulse Rate**:
- Decrease *A* (loudness) and increase *r* (pulse rate) gradually across all bats to shift from exploration to exploitation as iterations progress.

4. **Check Termination Condition**:
- Repeat Steps 2 and 3 until reaching *Max_iter* or achieving satisfactory *best_fitness*.

5. **Output the Best Solution**:
- Return *best_solution* (optimal feature subset) and *best_fitness* (its fitness score).

4. Simulations

In this section, we conduct simulations to assess the effectiveness of the proposed ICBA. Initially, we present the datasets and configurations used for the evaluation. Following this, we compare the performance of ICBA against a range of other algorithms. Lastly, we demonstrate the effectiveness of the enhancements integrated into ICBA.

4.1. Datasets and Setups

This work introduces five datasets sourced from the UC Irvine Machine Learning Repository [30], with Table 1 offering key details about these datasets.

Table 1: Datasets [30].

	Dataset	*No. of Features*	*No. of Instance*
1	Breastcancer	10	699
2	BreastEW	30	569
3	Congress	16	435
4	SpectEW	22	267
5	tic-tac-toe	9	958

In addition, a 6th Gen Intel (R) CoreTM i3-6006U @ 2.00 GHz CPU and 4 GB of RAM were utilized. We implemented the simulation codes using Python, and used a Forests classifier. Moreover, this paper benchmarks the proposed ICBA against several notable algorithms, including binary PSO (BPSO) [31], BGWO[32] , BDA [33], BBA [21], , and EBBA [17] are introduced as benchmarks. It's important to note that the ICBA and the benchmark algorithms share a uniform configuration of a population size of 24 and a total of 100 iterations. Additionally, a wrapper approach to feature selection is employed throughout this paper.

In our study, we selected the Random Forest classifier due to its simplicity, ease of application, high accuracy with complex datasets, and capability to mitigate overfitting. By integrating a wrapper method with this straightforward yet cost-effective classification algorithm, we can secure a robust feature subset well-suited for intricate classification challenges. Conversely, if a more sophisticated classification technique were employed for wrapper-based feature selection, the resultant feature subset might not perform as well with simpler classification models. This discrepancy arises because advanced classification algorithms tend to influence the wrapper approach's learning algorithm, such as the proposed ICBA, to adapt to the nuances of the classification technique itself rather than discerning the intrinsic relationships among various features.

4.2. Simulation Results

The accuracy rate optimization results using Equation (9) for several techniques are displayed in Table 2. Take note that the values that rank highest across all comparison methods are bolded. It is evident that the proposed ICBA gets the highest accuracy rate out of the five datasets. It indicates that the proposed ICBA performs better than all benchmark algorithms, making it a better choice for handling feature selection issues. (see fig 1, fig 2, fig 3, fig 4 and fig 5).

Table 2. The optimization outcomes, in terms of classification accuracy, obtained from several algorithms are presented, with the highest-performing values distinctly marked in bold. This notation emphasizes the algorithms that achieved the most favorable results in comparison to their counterparts. [6]

	Breastcancer	*BreastEW*	*Congress*	*SpectEW*	*tic-tac-toe*
BBA	0.9786	0.9613	0.9793	0.7379	0.8493
BDA	0.9767	0.9589	0.9743	0.7185	0.8099
BGWO	0.9767	0.9532	0.9750	0.7219	0.8465
BPSO	0.9786	0.9612	0.9785	0.7335	0.8521
EBBA	0.9786	0.9614	0.9793	0.7407	0.8521
ICBA	**0.9941**	**0.9857**	**0.9893**	**0.8691**	**0.9688**

4.3. Evaluation of Classifier Performance Across Various Algorithms

In this detailed examination, we investigate the intricacies of two notable classification techniques: the K-Nearest Neighbors (KNN) and the Decision Tree algorithms. To be more precise, the decision tree can be easily overfitted but it is simple to comprehend and visualize with minimal data preparation. Similarly, KNN may not be the most efficient choice for very large datasets, especially if computational resources are limited. The accuracy rate of the simulation results generated by different classification algorithms is displayed in Table 3. Note that the values that are bolded rank best when compared using all techniques of comparison. The random forest gets the highest accuracy rate. It suggests that it outperforms alternative classification algorithms, which makes it a superior option for dealing with problems involving feature selection. (see fig 6)

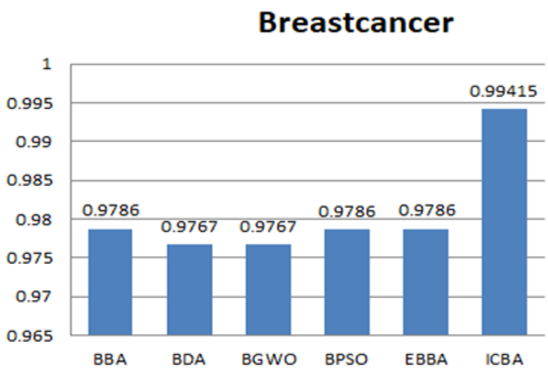


Figure 1: illustrates the comparative accuracy rates attained by diverse algorithms when applied to the Breast Cancer dataset

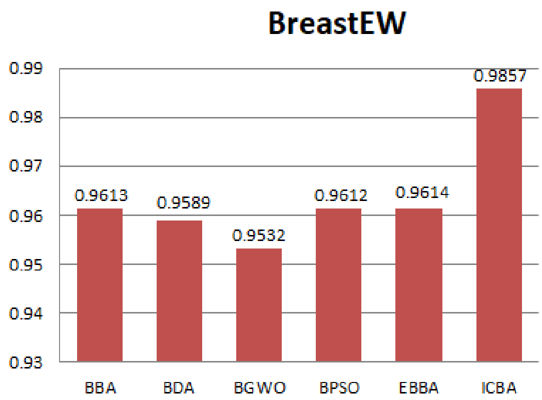
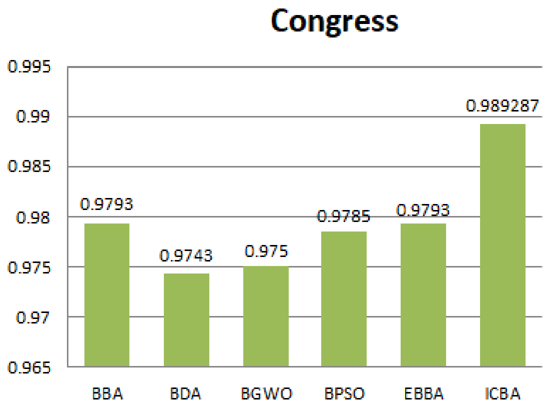


Figure 2: illustrates the comparative accuracy rates attained by diverse algorithms when applied to the BreastEW dataset



.

Figure 3: illustrates the comparative accuracy rates attained by diverse algorithms when applied to the Congress dataset

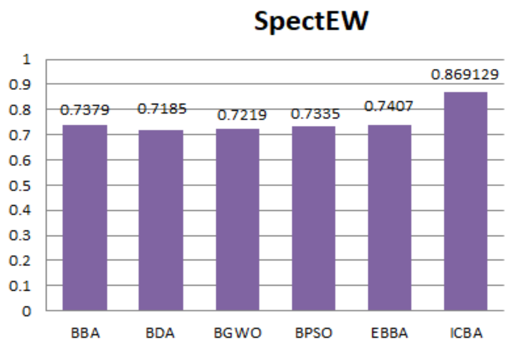


Figure 4: illustrates the comparative accuracy rates attained by diverse algorithms when applied to the Congress dataset

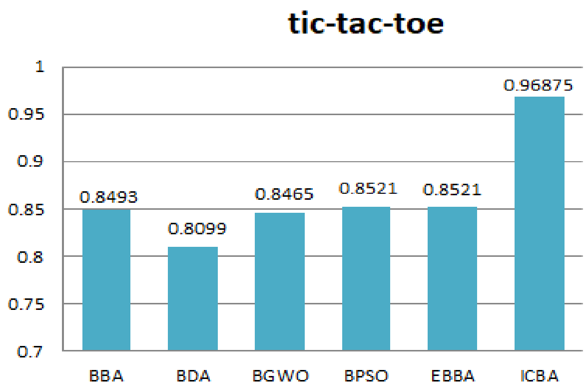


Figure 5: illustrates the comparative accuracy rates attained by diverse algorithms when applied to the tic-tac-toe dataset

Table 5: The outcomes of optimization derived through various classification algorithms, with the most superior results emphasized in bold.

	Breastcancer	*BreastEW*	*Congress*	*SpectEW*	*tic-tac-toe*
KNN	0.9809	0.9707	0.9847	0.8765	0.8646
Decision Tree(DT)	0.9824	0.9714	0.9847	0.8518	0.9201
Random forest	**0.99415**	**0.9857**	**0.9893**	**0.8765**	**0.9688**

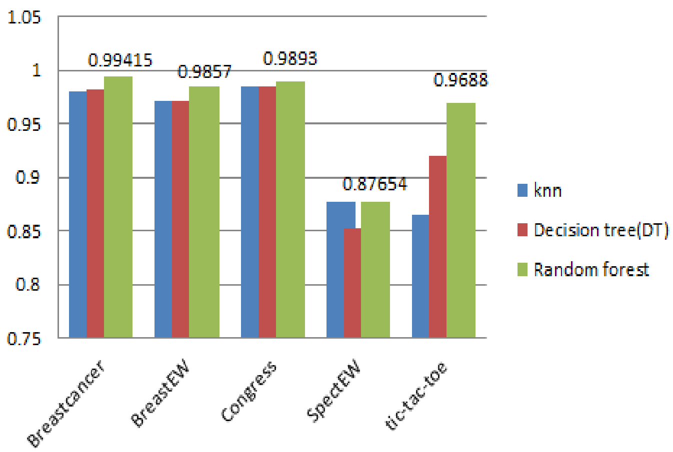


Figure 6: accuracy rates achieved by various classification algorithms.

Utilizing diverse classification algorithms, the ICBA methodology surpasses other comparative approaches, demonstrating its effectiveness across varying factors regardless of the classification technique employed. This leads to consistent robust performance in scenarios prone to overfitting and underfitting, highlighting the adaptability and strength of the ICBA strategy.

In the ICBA algorithm, these advantages are reflected in the experimental results. For example, ICBA outperforms other algorithms by consistently achieving higher accuracy across datasets, such as a 0.9941 accuracy on the (Breastcancer) dataset compared to 0.9786 with the next-best algorithm. This indicates that ICBA, through the chaotic map's influence, effectively navigates the feature selection space to select optimal feature subsets with higher classification accuracy.

5. Conclusions

This paper examines feature selection issues that have the potential to improve classification and decrease data dimension. We initially conceptualize the feature selection dilemma and reformulate it into a fitness function. Next, we propose an ICBA to address the issue of feature selection. In ICBA, we improved the BA and expanded its applicability to feature selection issues by integrating chaotic maps into it. Ultimately, ICBA is tested using simulations, and the outcomes show that it performs better than alternative comparative benchmarks. We plan to assess the approached ICBA using more realistic datasets in the future.

Conflicts of Interest

The authors declare that there is no conflict of interest that occurred when conducting this research paper.

References

[1] H. Liu, Y. Liu, R. Zhang, D. Liu, and Z. Zhang, "Feature selection embedded cluster distribution position for characteristic analysis of multi-dimension poverty-stricken households in China," *Journal of Applied Science and Engineering (Taiwan)*, vol. 24, no. 3, pp. 289–297, 2021, doi: 10.6180/jase.202106_24(3).0003.

[2] N. H. Ali, M. E. Abdulmunem, and A. E. Ali, "Learning evolution: A survey," *Iraqi Journal of Science*, vol. 62, no. 12, pp. 4978–4987, 2021.

[3] M. L. Bermingham *et al.*, "Application of high-dimensional feature selection: evaluation for genomic prediction in man," *Scientific reports*, vol. 5, no. 1, p. 10312, 2015.

[4] O. S. Qasim and Z. Y. Algamal, "Feature selection using different transfer functions for binary bat," *International Journal of Mathematical, Engineering and Management Sciences*, vol. 5, no. 4, pp. 697–706, 2020, doi: 10.33889/IJMEMS.2020.5.4.056.

[5] A. M. Al-Juboori *et al.*, "A hybrid cracked tiers detection system based on adaptive correlation features selection and deep belief neural networks," *Symmetry (Basel)*, vol. 15, no. 2, p. 358, 2023.

[6] A. S. Khalil and R. D. Al-Dabbagh, "A Comparative Study on Meta-Heuristic Algorithms for Solving the RNP Problem," *Iraqi Journal of Science*, vol. 60, no. 7, pp. 1639–1648, 2019.

[7] M. R. Ramli, Z. A. Abas, M. I. Desa, Z. Z. Abidin, and M. B. Alazzam, "Enhanced convergence of Bat Algorithm based on dimensional and inertia weight factor," *Journal of King Saud University-Computer and Information Sciences*, vol. 31, no. 4, pp. 452–458, 2019.

[8] X.-S. Yang, "Bat algorithm for multi-objective optimisation," *International Journal of Bio-Inspired Computation*, vol. 3, no. 5, pp. 267–274, 2011.

[9] S. Mirjalili, S. M. Mirjalili, and X.-S. Yang, "Binary bat algorithm," *Neural Computing and Applications*, vol. 25, pp. 663–681, 2014.

[10] N. F. Hassan, A. Al-Adhami, and M. S. Mahdi, "Digital speech files encryption based on Hénon and gingerbread chaotic maps," *Iraqi Journal of Science*, vol 63, no 2, pp. 830–842, 2022.
[11] O. Altay and E. V. Altay, "A novel chaotic transient search optimization algorithm for global optimization, real-world engineering problems and feature selection," *PeerJ Computer Science*, vol. 9, p. e1526, 2023.
[12] M. R. Feizi-Derakhsh and E. A. Kadhim, "An Improved Binary Cuckoo Search Algorithm for Feature Selection Using Filter Method and Chaotic Map," *Journal of Applied Science and Engineering (Taiwan)*, vol. 26, no. 6, pp. 897–903, 2023, doi: 10.6180/jase.202306_26(6).0015.
[13] M. Abdel-Basset, D. El-Shahat, I. El-henawy, V. H. C. de Albuquerque, and S. Mirjalili, "A new fusion of grey wolf optimizer algorithm with a two-phase mutation for feature selection," *Expert Systems with Applications*, vol. 139, Jan. 2020, doi: 10.1016/j.eswa.2019.112824.
[14] W. A. H. M. Ghanem *et al.*, "Cyber intrusion detection system based on a multiobjective binary bat algorithm for feature selection and enhanced bat algorithm for parameter optimization in neural networks," *IEEE Access*, vol. 10, pp. 76318–76339, 2022.
[15] A. M. Anter, A. T. Azar, and K. M. Fouad, "Intelligent hybrid approach for feature selection," in *The International Conference on Advanced Machine Learning Technologies and Applications (AMLTA2019) 4*, Springer, 2020, pp. 71–79.
[16] A. Da Li, B. Xue, and M. Zhang, "Improved binary particle swarm optimization for feature selection with new initialization and search space reduction strategies," *Applied Soft Computing*, vol. 106, Jul. 2021, doi: 10.1016/j.asoc.2021.107302.
[17] J. Feng, H. Kuang, and L. Zhang, "EBBA: An Enhanced Binary Bat Algorithm Integrated with Chaos Theory and Lévy Flight for Feature Selection," *Future Internet*, vol. 14, no. 6, Jun. 2022, doi: 10.3390/fi14060178.
[18] S. Eskandari and M. Seifaddini, "Online and offline streaming feature selection methods with bat algorithm for redundancy analysis," *Pattern Recognition*, vol. 133, p. 109007, 2023.
[19] A. M. Anter and M. Ali, "Feature selection strategy based on hybrid crow search optimization algorithm integrated with chaos theory and fuzzy c-means algorithm for medical diagnosis problems," *Soft Computing*, vol. 24, no. 3, pp. 1565–1584, Feb. 2020, doi: 10.1007/s00500-019-03988-3.
[20] M. A. Tawhid and K. B. Dsouza, "Hybrid binary bat enhanced particle swarm optimization algorithm for solving feature selection problems," *Applied Computing and Informatics*, vol. 16, no. 1–2, pp. 117–136, Apr. 2018, doi: 10.1016/j.aci.2018.04.001.
[21] R. Y. M. Nakamura, L. A. M. Pereira, K. A. Costa, D. Rodrigues, J. P. Papa, and X.-S. Yang, "BBA: a binary bat algorithm for feature selection," in *2012 25th SIBGRAPI conference on graphics, patterns and images*, IEEE, 2012, pp. 291–297.
[22] X. Zhou, F. Gao, X. Fang, and Z. Lan, "Improved bat algorithm for UAV path planning in three-dimensional space," *IEEE Access*, vol. 9, pp. 20100–20116, 2021.
[23] H. T. Rauf, J. Gao, A. Almadhor, M. Arif, and M. T. Nafis, "Enhanced bat algorithm for COVID-19 short-term forecasting using optimized LSTM," *Soft Computing*, vol. 25, no. 20, pp. 12989–12999, Oct. 2021, doi: 10.1007/s00500-021-06075-8.
[24] B. Remeseiro and V. Bolon-Canedo, "A review of feature selection methods in medical applications," *Computers in Biology and Medicine*, vol. 112. Elsevier Ltd, Sep. 01, 2019. doi: 10.1016/j.compbiomed.2019.103375.
[25] E. S. El-Kenawy and M. Eid, "Hybrid gray wolf and particle swarm optimization for feature selection," *International Journal of Innovative Computing, Information and Control*, vol. 16, no. 3, pp. 831–844, 2020, doi: 10.24507/ijicic.16.03.831.
[26] H. D. Praveena, C. Subhas, and K. R. Naidu, "Improved Artificial Bee Colony Based Feature Selection for Epileptic Seizure Detection.," *International Journal of Intelligent Engineering & Systems*, vol. 16, no. 3, pp. 332–344, 2023.
[27] A. Y. Mahdi and S. S. Yuhaniz, "An Intelligent Feature Selection Approach Based on a Novel Improve Binary Sparrow Search Algorithm for COVID-19 Classification.," *International Journal of Intelligent Engineering & Systems*, vol. 16, no. 4, p39, 2023.
[28] A. A. Hussein, E. T. Yassen, and A. N. Rashid, "Grey Wolf Optimizer for Green Vehicle Routing Problem.," *International Journal of Intelligent Engineering & Systems*, vol. 16, no. 5, pp. 627–636,2023.

[29] A. N. N. Kumar, M. Mallegowda, A. V. K. Mohan, K. N. Shreenath, and C. K. Raju, "Movement Mode Harmony Search Based Multi-Objective Firefly Algorithm Feature Selection for Detecting the Security Threats in Virtual Machine.," *International Journal of Intelligent Engineering & Systems*, vol. 17, no. 1, p159, 2024.

[30] A. Frank, "UCI machine learning repository," *http://archive. ics. uci. edu/ml*, 2010.

[31] R. K. Huda and H. Banka, "A group evaluation based binary PSO algorithm for feature selection in high dimensional data," *Evolutionary Intelligence*, vol. 14, no. 4, pp. 1949–1963, 2021.

[32] E. Emary, H. M. Zawbaa, and A. E. Hassanien, "Binary grey wolf optimization approaches for feature selection," *Neurocomputing*, vol. 172, pp. 371–381, 2016.

[33] M. M. Mafarja, D. Eleyan, I. Jaber, A. Hammouri, and S. Mirjalili, "Binary dragonfly algorithm for feature selection," in *2017 International conference on new trends in computing sciences (ICTCS)*, IEEE, 2017, pp. 12–17.