

Article

mDA: Evolutionary Machine Learning Algorithm for Feature Selection in Medical Domain

Ibrahim Aljarah ^{1,*}, Abdullah Alzaqebah ², Nailah Al-Madi ³, Ala' M. Al-Zoubi ⁴ and Amro Saleh ³¹ The Department of Artificial Intelligence, The University of Jordan, Amman 11942, Jordan² Department of Computer Science, The World Islamic Sciences and Education University, Amman 11947, Jordan³ Department of Computer Science, King Hussein School of Computing Sciences, Princess Sumaya University for Technology, Amman 11941, Jordan⁴ Department of Data Science and Artificial Intelligence, Faculty of Science and Information Technology, Al-Zaytoonah University of Jordan, Amman 11733, Jordan

* Correspondence: i.aljarah@ju.edu.jo

Abstract

The rapid expansion of medical data, characterized by its complex high-dimensional attributes, presents numerous promising opportunities and substantial challenges in health-care analytics. Adopting effective feature selection techniques is essential to take advantage of the potential of such data. This research presents a modified algorithm called (mDA), which is the hybrid algorithm between the Evolutionary Population Dynamics and the Dragonfly Algorithm. This method combines Evolutionary Population Dynamics's strength with the Dragonfly Algorithm's flexible capabilities, offering a robust evolutionary machine learning approach specifically designed for medical data analysis. By integrating the dynamic population modeling of Evolutionary Population Dynamics with the adaptive search techniques of Dragonfly Algorithm, the proposed mDA significantly improves accuracy, reduces the number of features, and obtains the minimum average of the fitness scores. Comparative experiments conducted on seven diverse medical datasets against other established algorithms confirm the superior performance of the proposed mDA, establishing it as a valuable approach in examining complex medical data.

Keywords: feature selection; dragonfly algorithm; optimization; medical data analytics; evolutionary algorithms; classification

Academic Editors: Shi Cheng,
Chaomin Luo and Shangce Gao

Received: 19 October 2025

Revised: 3 December 2025

Accepted: 10 December 2025

Published: 13 December 2025

Citation: Aljarah, I.; Alzaqebah, A.;
Al-Madi, N.; Al-Zoubi, A.M.; Saleh, A.mDA: Evolutionary Machine
Learning Algorithm for Feature
Selection in Medical Domain.*Computation* **2025**, *13*, 292.[https://doi.org/10.3390/
computation13120292](https://doi.org/10.3390/computation13120292)

Copyright: © 2025 by the authors.
Licensee MDPI, Basel, Switzerland.
This article is an open access article
distributed under the terms and
conditions of the Creative Commons
Attribution (CC BY) license
([https://creativecommons.org/
licenses/by/4.0/](https://creativecommons.org/licenses/by/4.0/)).

1. Introduction

The swift progress in the development of medical technologies and the widespread use of electronic health records have created enormous amounts of medical data. This high-dimensional data holds the potential to transform healthcare by facilitating more precise diagnoses, customized treatments, and predictive analytics. Nonetheless, this data's vast volume and complexity pose substantial challenges, particularly in extracting meaningful insights. A major challenge is the existence of irrelevant or redundant features that can hinder the effectiveness of machine learning models, causing overfitting and diminished generalization ability.

In recent decades, metaheuristic and evolutionary algorithms have proven to be highly effective in solving a variety of optimization problems [1–5]. The Dragonfly Algorithm (DA), a contemporary metaheuristic inspired by the behavior of dragonflies [6], stands out as a recently successful algorithm capable of surpassing other well-established optimizers

in the literature. It has been employed in diverse real-world applications, including economic emission dispatch in power systems [7,8], simulation building [9], wireless node localization in computer networks [10], and machine learning [11,12]. The DA has demonstrated excellent performance across numerous continuous, discrete, single-objective, and multi-objective optimization problems, outperforming several state-of-the-art metaheuristic and evolutionary algorithms such as Particle Swarm Optimization (PSO) and Differential Evolution (DE). Recent studies have also highlighted the growing role of intelligent and language-aware AI systems in healthcare and social media analytics, demonstrating effective applications of machine learning and natural language processing in Arabic and multilingual contexts [13–15].

Recently, the authors in [6] presented a binary version of the DA, known as BDA, which utilizes a transfer function (TF) to transform a continuous search space into a discrete one. An initial evaluation of BDA's effectiveness was performed on various feature selection challenges, with the findings indicating the method's satisfactory performance [16].

The persuasive advantages of the EPD operator prompted us to incorporate it into the newly developed Dragonfly Algorithm (DA) to evaluate its efficacy on FS problems. In this research, we adapted the DA by choosing the top three solutions along with one that is randomly generated to reposition a solution from the lower half of the population. This strategy allows solutions with lower fitness to influence the population's structure. Comprehensive results and extensive comparisons indicate that the EPD significantly boosts the DA's performance, enhancing the proposed method's ability to surpass other optimizers and achieve superior solutions with better convergence characteristics. This study presents an EPD-enhanced DA-based optimizer aimed at improving the basic DA's performance on FS tasks. Our main contributions in this research include:

- The notable advantages of the EPD operator encouraged us to utilize it with the recently introduced DA and examine its efficiency in FS problems in the medical domain.
- In the suggested method, a solution from the worst 50% of the population is repositioned by choosing one of the top three solutions and a randomly generated solution.
- The suggested method has been evaluated on seven medical datasets, each with unique configurations and attributes, to illustrate its effectiveness, solution quality, and efficiency in feature selection tasks.
- The EPD operator is combined with the modified DA (mDA) for the first time to address feature selection problems.

The paper is structured as follows: Section 2 covers related work. Section 3 highlighted the basics of DA, binary DA, and the EPD operator. Section 4 outlines the proposed methodology. Section 5 presents the results. Section 6 highlighted the clinical discussion of the findings. Lastly, Section 7 provides the conclusions and suggests avenues for future research.

2. Related Works

In our literature review, we employed a systematic research approach to locate and evaluate pertinent studies. We conducted searches in key scientific databases such as Elsevier, IEEE, Springer, and MDPI, utilizing specific keywords like metaheuristics, evolutionary computation, nature-inspired approaches, hybrid approaches, local search, Evolutionary Population Dynamics (EPD), and the Dragonfly Algorithm (DA). Our selection criteria were aimed at studies that proposed, examined, or implemented hybrid metaheuristic algorithms, with a special focus on methods integrating EPD and DA or similar evolutionary strategies. This approach guaranteed a thorough and focused review of existing literature, enabling us to evaluate the theoretical and practical contributions of current methods and to distinctly define the innovation of the proposed mDA algorithm.

Numerous studies have attempted to apply DA or enhance its effectiveness in addressing practical challenges such as photovoltaic systems [17], prolonging RFID network lifespan [18], 0-1 knapsack problems [19], and the economic emission dispatch problem [7]. In 2017, the researchers in [20] introduced a memory-based hybrid DA combined with PSO principles for global optimization problems. Moreover, the authors in [21] developed a modified DA with elite opposition learning for global optimization.

DA is applied within the healthcare care domain for the purpose of feature selection. The research in [22] aimed to categorize breast cancer tumors as either benign or malignant. Implementing the Dragonfly algorithm allows a curated selection of features to be identified, thereby augmenting the precision of classification models. The algorithm enhances the feature selection methodology by systematically identifying the most salient features and concurrently discarding redundancies. This methodological framework enhances diagnostic accuracy in the medical field, particularly in differentiating among various classifications of breast cancer tumors. DA was employed in [23] for the purpose of feature selection within the domain of machine learning; the investigation did apply this algorithm to a dataset pertaining to the classification of chronic kidney disease, thereby demonstrating considerable enhancements in classification precision. Although the primary emphasis was placed on improving classification results, the efficacy of the dragonfly algorithm in feature selection could yield advantages in medical contexts for applications such as disease diagnosis or prognosis.

Subsequent investigations could delve into the algorithm's potential in analyzing medical data to advance predictive models within healthcare environments. In [24], DA was utilized in medical image registration within the context of this investigation. It was evaluated against alternative bio-inspired algorithms, including particle swarm optimization and artificial bee colony methods. The outcomes of the simulations demonstrated that the dragonfly algorithm yielded superior quality in image registration results, though with a prolonged convergence time. This contradiction between the quality of registration and the computational duration is of principal importance when selecting an algorithm for medical applications, such as the monitoring of tumor progression. Consequently, DA has shown significant potential in the domain of medical image registration tasks, providing high-quality outcomes despite the associated increase in computational time. In the segmentation of thermographic images for early diagnosis of breast diseases, ref. [25] mimicking the swarming behaviors of dragonflies, the algorithm balances exploration and exploitation phases to compute optimal thresholds for image segmentation aims to provide a reliable method for clinicians to analyze thermography images effectively, assisting in the early detection of breast cancer.

Building upon these earlier medical applications of the DA algorithm, recent literature has demonstrated a clear transition toward hybrid evolutionary–deep learning paradigms for FS and transformer-based architectures with intrinsic interpretability mechanisms. A comprehensive literature survey [26] distills the current status of evolutionary feature selection, focusing on integration approaches for attention, adaptive population modeling, and multi-objective optimization. These approaches have a significant impact on structuring the proposed mDA algorithm.

Similarly, parallel literature on deep attention networks and vision transformer models [27,28] has been observed for integration purposes in healthcare data, including imaging and electronic health records. These models utilize attention pooling, hierarchical fusion, and representation mechanisms that inherently produce salient features that meet the criteria for feature selection. The interpretability naturally incorporated by these transformer models makes it easier to validate them for clinical use.

Methodologically, recent breakthroughs have been made in domain-aware transformer frameworks [29] that incorporate priors informed by physics and biology into their optimization approaches. These models provide enhanced semantic coherence and can be conceptualized alongside traditional wrapper approaches to FS that rely on domain information to inform and guide the process of identifying salient features. This continues to strengthen the role of jointly evolving search approaches with knowledge-informed architectures for robustness and interpretability.

From a visualization and explainability perspective, recent work continues to proliferate in post-hoc attribution methods such as SHAP and LIME [30] to effectively integrate these attribution approaches with FS algorithms, guaranteeing clinical plausibility and interpretability for these models. Empirical analyses [31] have verified that clinician-centered explanations can outperform standard SHAP explanations in both clinician trust and diagnostic accuracy, underscoring the need for interpretability and user-centered design in medical AI. Complementary analyses have concurrently argued for the use of attention maps, gradient attribution, and perturbation-based validations to ensure the relevance of selected features in clinical decision-support systems.

Taken together, these recent developments represent a clear paradigm shift toward interpretable, domain-grounded, and hybridized feature selection frameworks. They have established a coherent research stream integrating evolutionary search, deep representational learning, and explainable AI, a trajectory that is directly reflected in the conception and design of our proposed mDA model.

3. Preliminaries

3.1. Dragonfly Algorithm (DA)

The Dragonfly Algorithm is a newly introduced swarm intelligence algorithm [6]. This algorithm imitates the predation and migration behaviors of conceptualized dragonflies. The predation behavior, termed a static swarm (feeding), involves dragonflies flying in small clusters within a confined area to locate food sources. On the other hand, the migration behavior, called dynamic swarm (migratory), involves dragonflies flying in larger groups in a single direction to facilitate the swarm's migration.

Like other algorithms inspired by nature, DA operates in two phases: the exploration phase, which is based on static swarming behavior, and the exploitation phase, which is based on dynamic swarming behavior.

Five individual behaviors are used to simulate the swarming actions of dragonflies. In the equations below, X denotes the current search agent's position, X_j indicates the j -th neighbor of the X search agent, and N represents the size of the neighborhood [32]:

- *Separation* is a strategy used by a search agent to maintain distance from other nearby search agents. This action is represented mathematically as Equation (1):

$$S_i = - \sum_{j=1}^N X - X_i \quad (1)$$

- *Alignment* describes how a single entity adjusts its speed to align with the speeds of other nearby entities. This action is mathematically represented by Equation (2):

$$A_i = \frac{\sum_{j=1}^N V_j}{N} \quad (2)$$

where V_j denotes the speed of the j -th neighboring entity.

- *Cohesion* indicates the propensity of individuals to move towards the nearby center of mass, as represented mathematically by Equation (3):

$$C_i = \frac{\sum_{j=1}^N x_j}{N} - X \quad (3)$$

- *Attraction* describes the inclination of individuals to move toward the food source. The mathematical representation of the attraction between the food source and the i^{th} solution is given by Equation (4):

$$F_i = F_{loc} - X \quad (4)$$

where F_{loc} denotes the location of the food source.

- *Distraction* This describes the natural inclination of individuals to flee from a threat. The separation between the adversary and the i^{th} solution is represented mathematically by Equation (5):

$$E_i = E_{loc} + X \quad (5)$$

where E_{loc} represents the enemy's position.

Within the DA, the fitness and position of the food source are intended to be revised using the top-performing candidate (search agent) up to that point. Moreover, the fitness and position of the adversary should be adjusted based on the least successful candidate. This leads to a convergence toward favorable regions and a divergence from less favorable areas within the search space.

According to the PSO algorithm framework, the DA updates a dragonfly's position using two vectors: the step vector (ΔX), akin to the velocity vector in PSO, and the position vector. The step vector indicates the direction in which the dragonflies move. The step vector is modeled as Equation (6):

$$\Delta X_{t+1} = (sS_i + aA_i + cC_i + fF_i + eE_i) + wX_t \quad (6)$$

In this context, s , w , a , c , f , and e denote the weights for the separation S_i , alignment A_i , cohesion C_i , attraction towards the food source F_i , and distraction from the enemy E_i for each i -th individual, respectively. These weights allow the DA to exhibit varying degrees of exploration and intensification during the optimization process. A comprehensive study of how these parameters affect the DA and their specific values is available in [6].

The position of an individual is updated as in Equation (7):

$$X_{t+1} = X_t + \Delta X_{t+1} \quad (7)$$

where t is the current iteration.

The pseudocode for the DA is presented in Algorithm 1. The process begins with generating a random initial population, where the dragonflies' positions and step vectors are arbitrarily assigned. During each iteration, the algorithm performs the following actions repeatedly until a stopping condition is met. Firstly, each population member is assessed via a fitness function. Secondly, the primary coefficients are revised. Thirdly, using Equations (1)–(5), the separation (S), alignment (A), cohesion (C), food source (F), and enemy (E) are updated. Lastly, the step vectors and positions are adjusted based on Equations (6) and (7), correspondingly.

Finally, the best solution found so far is returned.

Algorithm 1 Pseudocode of the DA

```

Initialize the population  $X_i (i = 1, 2, \dots, n)$ 
Set  $\Delta X_i (i = 1, 2, \dots, n)$ 
while (termination condition is not met) do
    Assess each dragonfly
    Update (F) and (E)
    Modify the primary coefficients ( $i, e, w, s, a, c, f$ , and  $e$ )
    Compute  $S, A, C, F$ , and  $E$  by applying Equations (1)–(5)
    Revise step vectors ( $\Delta X_{t+1}$ ) via Equation (6)
    Revise  $X_{t+1}$  using Equation (7)
Return the best solution

```

3.2. Binary Dragonfly Algorithm (BDA)

In binary optimization problems, the search space is modeled as a hypercube, and an individual's position can be altered by flipping one or more bits in its position vector $x = \{x_1, x_2, \dots, x_d\}$. Since the original DA was intended for continuous optimization problems, it updates the individual's position by adding the step vector to the current position vector. However, this method does not apply to binary optimization problems like feature selection. As indicated in a previous study [33], employing a transfer function is an efficient and practical method to adapt a continuous algorithm to a binary context. This paper utilizes both S-shaped and V-shaped transfer functions.

In general, transfer functions are utilized to compute the likelihood of altering the elements of a position to either 0 or 1, depending on the value of the step vector (velocity) of the search agent indexed by i in the dimension indexed by d during the current iteration (t) as an input parameter. In earlier research [6], the transfer function of Equation (8) was employed to determine the likelihood of converting continuous positions to binary.

$$T(v_d^i(t)) = |(v_d^i(t)) / \sqrt{1 + (v_d^i(t))^2}| \quad (8)$$

The result $T(v_k^i(t))$, obtained from Equation (8) is subsequently employed to transform the i -th component of the position vector into either 0 or 1 as per Equation (9)

$$X(t+1) = \begin{cases} \neg X_t & r < T(v_k^i(t)) \\ X_t & r \geq T(v_k^i(t)) \end{cases} \quad (9)$$

where r is a random number in the $[0,1]$ interval.

The step vector represents the current individual's dynamism and determines the movement's extent. A smaller step vector value implies that the individual is nearing the optimal solution and should make minor adjustments (exploitation). Conversely, a higher step vector value indicates the search agent is distant from the optimal solution and necessitates significant changes (exploration) [34]. In a binary algorithm utilizing the step vector to determine the probability of position changes, transfer functions immensely influence the equilibrium between exploration and exploitation. The probability calculation will stay constant throughout the optimization process if the transfer function remains unchanged. Modifying the transfer function can enhance the influence of the step vector on position alterations for both exploration and exploitation of the search space.

3.3. Evolutionary Population Dynamics (EPD)

Evolutionary algorithms (EAs) are stochastic search methods where a group of potential solutions (population) is initialized and then gradually enhanced to meet the specified objectives better. Certain EAs employ mutation mechanisms to modify selected solutions,

while others use crossover operators. These operators aim to evolve the chosen solutions, which are often the most optimal ones. Evolutionary Population Dynamics (EPD) involves eliminating the least optimal solutions in a population and repositioning them near the best ones. This approach is fundamentally based on the theory of self-organized criticality (SOC) [35], which suggests that a local alteration in the population can influence the entire population, leading to delicate balances without an external organizing force [36]. Genetic algorithms merge the best solutions using evolutionary operators such as crossover and mutation. Conversely, EPD excludes the least optimal solutions from the current population. Two metaheuristic methods inspired by the SOC concept are evolutionary programming using self-organizing criticality (EPSO) [37] and extremal optimization (EO) [38]. EPD is a straightforward and effective mechanism that can be integrated into various optimizers. It begins by removing the least optimal solutions from the swarm and subsequently repositioning these removed solutions around the top search agents.

4. Methodology

Feature selection is presented as a binary optimization challenge, limiting solutions to binary outcomes. Therefore, the binary version of the DA can be utilized to tackle this challenge. In this research, a vector consisting of zeroes and ones represents a solution to a FS problem, with a zero indicating that the associated feature is not selected and a one indicating that the feature is selected. The length of the solution vector corresponds to the number of features present in the original dataset. This study introduces eight wrapper feature selection techniques that leverage the BDA. Each technique uses a transfer function to convert a continuous value into a binary form. The KNN classifier [39] is employed to evaluate the selected feature subsets. The fitness function takes into account both classification accuracy and the number of features selected, in line with the understanding that feature selection is a multi-objective task. The objective function is shown in Equation (10):

$$Fitness = \alpha \gamma_R(D) + \beta \frac{|C|}{|N|} \quad (10)$$

where $\gamma_R(D)$ denotes the classification error rate, $|C|$ signifies the count of selected features, and $|N|$ represents the total number of features in the initial dataset. The parameters α and β correspond to the significance of classification accuracy and subset length, respectively. α ranges within the interval $[0,1]$, and β is defined as $(1 - \alpha)$, adapted from [40].

4.1. Applying the EPD Strategy to BDA

As previously discussed, the EPD approach discards the least efficient solutions from the population and replaces them by creating new solutions in the vicinity of the more effective ones. This EPD strategy serves as a simple yet efficient operator for methods based on populations [36], and hence it is incorporated into the traditional DA as it is also a stochastic population-based optimizer. EPD improves the exploitation capability of BDA by removing the poorest solutions from the group and creating new nearby solutions around the superior ones.

To incorporate the EPD technique within the BDA algorithm, the swarm of dragonflies is split into two groups after sorting by their fitness scores. The less fit group is removed and reinitialized using four different strategies derived from the better half of the population.

In this study, we integrated the EPD scheme with the binary DA. The hybridization model utilizes a random selection operator. Specifically, one of the top three dragonflies in the population is chosen along with a randomly selected dragonfly. Subsequently, the 'poor' solution's leader is chosen randomly. To execute this concept, a random selection

mechanism is used to pick the solutions. Additionally, this method incorporates a basic mutation operator.

In this method, the top three individuals are chosen, and a fourth solution is created randomly. Each of the worst half solutions is repositioned around one of these four solutions based on a generated random number. The procedure is simple: a random number X_r is generated in each iteration, and one of four choices is applied to reposition the suboptimal solution: if $X_r \in [0, 0.25]$, the best solution is used; if $X_r \in [0.25, 0.5]$, the second-best solution is used; if $X_r \in [0.5, 0.75]$, the third-best solution is chosen; and if $X_r \in [0.75, 1]$, a random solution is used.

The selected solution will be used as a starting point to reposition the poor solution. Repositioning the poor solutions around the best solutions aims to heighten the median of the swarm in each step. However, this process may cause a premature convergence of the algorithm. As a remedy, a randomly generated solution is used in the first rule to promote exploration and prevent trapping in local optima.

The overall pseudo code of the mDA algorithm is described in Algorithm 2.

Algorithm 2 Pseudocode of the mDA

```

Initialize the population  $X_i (i = 1, 2, \dots, n)$ 
Initialize  $\Delta X_i (i = 1, 2, \dots, n)$ 
while (end condition is not satisfied) do
    Evaluate each dragonfly
    Update  $(F)$  and  $(E)$ 
    Update the main coefficients ( $i, e, w, s, a, c, f$ , and  $e$ )
    Calculate  $S, A, C, F$ , and  $E$  (using Equations (1)–(5))
    Update step vectors ( $\Delta X_{t+1}$ ) using Equations (8) and (9)
    for  $i = (n/2) + 1$  to  $n$  do
        Update the position of  $i$ -th dragonfly using EPD approach
    Return the best solution using Equation (10)

```

It should be noted that the computational complexity of the proposed modified Differential Algorithm (mDA) is not substantially different from that of the original Differential Algorithm (DA). The DA has a computational complexity of $O(t \times d \times n^2)$, where t represents the number of iterations, d stands for the number of variables, and n denotes the number of solutions. The introduction of binary operators does not alter this complexity, as they are incorporated into the original DA's position updating method. However, to reinitialize 50% of the solutions, an additional complexity of $O(n/2)$ is introduced, making the overall computational complexity of the proposed mDA $O(t \times d \times n^2 + n/2)$. It is important to note that because half of the solutions need to be re-evaluated for their objective value, the mDA requires $n/2$ more function evaluations than the DA.

4.2. Experiments

We conducted our experiments using seven publicly accessible medical datasets: Breastcancer, BreastEW, Colon, HeartEW, Leukemia, Lymphography, and PenglungEW. The comparative algorithms included Traditional PSO, Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Grey Wolf Optimizer (GWO), Salp Swarm Algorithm (SSA), Grasshopper Algorithm (GOA), and Harris Hawks Optimization (HHO). The evaluation metrics comprised Accuracy, mean number of selected features, and mean best fitness.

4.3. Dataset Description

We assess the performance of the proposed nDA model compared to the binary DA version and other methods using seven renowned medical datasets obtained from the UCI benchmark repository [41,42].

Table 1 summarizes the datasets employed in our experiments. It enumerates seven unique datasets, each selected for its significance to particular medical and biological issues. The table describes the number of features, cases, and categories for each dataset, which are essential measures for assessing the data's complexity and breadth. Detailed information of each dataset is given below:

- **Breastcancer:** This dataset contains 9 features with 699 instances, divided into 2 classes, making it suitable for binary classification tasks related to breast cancer detection.
- **BreastEW:** Comprising 30 features and 569 instances, this dataset also targets breast cancer but with a different feature set, reflecting varied experimental conditions or data collection methodologies.
- **Colon:** A relatively small dataset with only 62 instances but a high dimensionality of 2000 features, primarily used for studying gene expression profiles in colon cancer.
- **HeartEW:** Contains 270 instances and 13 features used in analyzing heart disease with two outcome classes.
- **Leukemia:** The most feature-rich dataset in the collection, with 7129 features across 72 instances, indicative of high-throughput genetic profiling in leukemia studies.
- **Lymphography:** Consists of 148 instances and 18 features, used in diagnosing lymph node tumors with binary class outcomes.
- **PenglungEW:** Distinct from the others, this dataset has 73 instances and 325 features but expands the classification challenge to 7 classes, possibly indicating different stages or types of lung diseases.

Table 1. List of the Datasets Used in the Experiments.

No.	Dataset	No. of Features	No. of Instances	No. of Classes
1.	Breastcancer	9	699	2
2.	BreastEW	30	569	2
3.	Colon	2000	62	2
4.	HeartEW	13	270	2
5.	Leukemia	7129	72	2
6.	Lymphography	18	148	2
7.	PenglungEW	325	73	7

4.4. Evaluation and Experimental Settings

The suggested approach is implemented using the Matlab R2019a tool on an Intel(R) Core i7 processor running at 2.00 GHz with 16 GB of RAM. The suggested and contrasted alternatives are implemented using the same platform and programming language to provide fair comparisons.

In this work, the performance of the algorithms on the findings obtained was validated using the cross-validation method, which randomly splits each data set into 80% training and 20% testing parts [43,44]. Furthermore, each technique was run 20 times to ensure robustness and reliability of the results.

The classification accuracy in Equation (11) is used to evaluate the proposed approach, an important measure to evaluate the classification problems; more accuracy means a better solution. On the other hand, the FS algorithms aim to reduce the dimensionality of the dataset by selecting the minimum number of features concerning the classification accuracy. The accuracy and the number of the selected features are included as objectives of the fitness function; therefore, the objective is to minimize the number of features and increase the accuracy of the classification. So, the accuracy is converted to a minimization problem by taking the error rate instead of the accuracy (1-accuracy), as shown in Equation (12).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (11)$$

$$Fitness = \alpha \times (ErrorRate) + \beta \times \frac{|R|}{|N|} \quad (12)$$

where α and β are parameters between 0 and 1 to represent the importance weight of each objective ($\beta = 1 - \alpha$), *ErrorRate* indicates the classification error rate, *R* represents the number of selected features and the total number of features is denoted as *N*, based on the literature [43–47]; α is set to 0.99 and β equal to 0.01.

A sensitivity analysis was utilized to choose the experiment settings properly. The population size was set up in the analysis using various alternative values: 10, 20, 30, 50, and 100 search agents. While the maximum number of iterations was tested on three values: 50, 100, and 150. As clearly shown in Table 2, the population size with a value equal to 30 performed better. In addition, the maximum number of iterations is set to 100, based on the sensitivity results and according to previous studies [45–47]. On the other hand, the KNN is the most often used classifier with the different datasets available in the UCI repository. And according to [43,44], the value of *k* is set to 5.

Table 2. A sensitivity analysis of the classification accuracy for mDA algorithm with various numbers of iterations and population size.

Benchmark	Iterations	Population Size				
		10	20	30	50	100
Breastcancer	50	0.9594	0.9611	0.9581	0.9571	0.9606
	100	0.9588	0.9576	0.9678	0.9597	0.9581
	150	0.9611	0.9579	0.9573	0.9603	0.9606
BreastEW	50	0.9298	0.9322	0.9293	0.9301	0.9287
	100	0.9288	0.9368	0.9382	0.9354	0.9307
	150	0.9356	0.9344	0.9353	0.9362	0.9337
Colon	50	0.6265	0.6306	0.6143	0.6643	0.6143
	100	0.6163	0.6235	0.7174	0.6143	0.5847
	150	0.6204	0.5959	0.6357	0.6357	0.6051
HeartEW	50	0.7921	0.7907	0.7979	0.7900	0.7894
	100	0.8021	0.7968	0.8227	0.8079	0.7898
	150	0.7852	0.7921	0.7984	0.7813	0.7928
Leukemia	50	0.7553	0.8009	0.7491	0.7684	0.7719
	100	0.7728	0.7640	0.8211	0.7386	0.7447
	150	0.7307	0.7763	0.7149	0.7526	0.7763
Lymphography	50	0.7389	0.7369	0.7477	0.7406	0.7466
	100	0.7243	0.7390	0.7627	0.7513	0.7411
	150	0.7390	0.7165	0.7311	0.7245	0.7441
PenglungEW	50	0.5560	0.5138	0.5379	0.5319	0.4888
	100	0.4966	0.5310	0.6035	0.4966	0.5241
	150	0.5284	0.5724	0.4845	0.5500	0.5345

The sensitivity analysis of the classification accuracy for the proposed algorithm (mDA), showing how it performs across a variety of medical datasets with differing numbers of iterations and population sizes. The results reveal that the Breastcancer dataset consistently exhibits high classification accuracy, which marginally improves as the number of iterations increases, indicating the algorithm’s stability across different population sizes. In contrast, the Colon dataset shows notable variability in accuracy, especially at larger

population sizes, which might suggest a sensitivity to the dimensionality of the feature space or a propensity for overfitting. The Leukemia and HeartEW datasets display a positive trend in accuracy with increasing iterations, underscoring that more iterations aid in achieving better generalization of the model. However, the PenglungEW dataset, which is categorized into multiple classes, tends to have lower overall accuracy, pointing to the inherent challenges somehow of multi-class classification. This analysis provides essential insights into the robustness and efficiency of the mDA algorithm, highlighting its potential advantages and limitations when applied to different types of medical datasets.

Table 3 enumerates the parameters utilized in the experiments outlined in the study, providing a clear framework for the setup and execution of the classification models tested. The population size for the experiments was set at 30 based on the sensitivity analysis experiment, with a maximum of 100 iterations per run to allow the algorithms sufficient opportunity to converge toward optimal solutions. The K-nearest neighbors algorithm (KNN) used a value of K equal to 5, optimizing the balance between bias and variance in the classification. Two parameters, α , and β , were set at 0.99 and 0.01 respectively in the fitness function, indicating a strong preference for one aspect of the model's performance over another, likely focusing on maximizing accuracy while controlling for overfitting or complexity in the model. This structured approach in parameter selection underscores the methodological rigor of the experiments, aiming to achieve both precise and generalizable outcomes.

Table 3. List of the used Parameters in the Experiments.

No.	Parameter	Value
1.	Training Data	80%
2.	Testing Data	20%
3.	Population Size	30
4.	Max Number of iterations	100
5.	K (Knn)	5
6.	Number of runs for each technique	20
7.	α in fitness function	0.99
8.	β in fitness function	0.01

5. Results

Table 4 provides a comparative analysis of classification accuracy between the proposed algorithm (mDA) and other well-known optimization algorithms across various medical datasets. The table includes algorithms such as Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Grey Wolf Optimizer (GWO), Salp Swarm Algorithm (SSA), Grasshopper Optimisation Algorithm (GOA), Harris Hawks Optimization (HHO), and standard (DA).

The mDA algorithm generally shows superior or competitive performance compared to other algorithms. For instance, in the Breastcancer dataset, mDA achieves a classification accuracy of 0.9678, slightly outperforming the nearest competitor, GOA, which scored 0.9673. Similarly, in the BreastEW dataset, mDA's accuracy of 0.9382 is among the highest, surpassed only by GA's 0.9581. However, in the Colon and PenglungEW datasets, while mDA does not lead, it still maintains a respectable performance, indicating its robustness across different types of data challenges. This comparative analysis highlights the efficacy of mDA in achieving high classification accuracy and showcases its potential as a reliable tool in medical data analysis.

Table 4. A comparative analysis of classification accuracy between the proposed mDA and other algorithms.

Benchmark	GA	PSO	GWO	SSA	GOA	HHO	DA	mDA
Breastcancer	95.81 ± 0.21	96.42 ± 0.78	96.14 ± 0.81	95.91 ± 1	96.73 ± 2.6	94.99 ± 1.8	96.24 ± 0.01	96.78 ± 1.18
BreastEW	91.87 ± 2.71	93.31 ± 2.51	94.52 ± 5.32	93.00 ± 1.34	94.22 ± 1.55	93.96 ± 0.43	94.26 ± 1.75	93.82 ± 0.85
Colon	64.59 ± 0.85	67.25 ± 1.7	63.98 ± 1.91	61.22 ± 0.12	59.29 ± 1.16	66.63 ± 0.11	69.29 ± 3.12	71.74 ± 1.53
HeartEW	81.90 ± 0.01	79.93 ± 0.22	79.28 ± 1.19	79.61 ± 3.42	74.31 ± 2.3	78.54 ± 4.48	79.47 ± 1.24	82.27 ± 0.01
Leukemia	72.11 ± 0.5	79.65 ± 1.55	81.14 ± 3.69	62.11 ± 3	64.83 ± 1.52	78.42 ± 2.99	81.84 ± 0.84	82.11 ± 2.47
Lymphography	75.98 ± 1.01	75.42 ± 2.26	72.42 ± 1.64	74.75 ± 2.24	75.00 ± 0.16	71.91 ± 0.92	68.14 ± 5.25	76.27 ± 4.24
PenglungEW	53.79 ± 1.74	47.24 ± 0.81	49.74 ± 0.01	61.55 ± 1.14	45.00 ± 1.77	40.00 ± 4.86	63.02 ± 1.77	60.35 ± 3.27
Ranking (Wins)	0	0	1	0	0	0	1	5
Ranking (Ties)	0	0	0	0	0	0	0	0
Ranking (Losses)	7	7	6	7	7	7	6	2
Ranks (F-test)	4.8571	4.0000	4.5714	5.5714	5.5714	6.1429	3.4286	1.8571

Table 5 presents a comparative analysis of the average number of selected features across various algorithms when applied to different medical datasets. The table highlights the efficiency of each algorithm in feature selection, which is critical for reducing model complexity while maintaining or enhancing prediction accuracy. For instance, in the Breast-cancer dataset, mDA and HHO select the fewest features, demonstrating their capability to achieve efficient feature reduction. In contrast, the Colon dataset shows significant variation in the number of features selected, with mDA choosing 923.20 features on average, which is lower than GWO and SSA but higher than DA and HHO.

Table 5. A comparison results in terms of average number of selected features.

Benchmark	GA	PSO	GWO	SSA	GOA	HHO	DA	mDA
Breastcancer	4.45	4.00	5.55	5.05	5.00	4.00	5.00	5.00
BreastEW	14.00	15.90	23.60	18.05	15.75	19.85	14.80	13.60
Colon	697.40	943.50	1150.95	1164.40	654.10	973.80	873.30	923.20
HeartEW	10.35	7.80	11.35	10.85	7.60	8.15	8.85	8.40
Leukemia	2952.00	3443.35	3785.75	3929.25	1320.05	3585.65	3088.25	3332.30
Lymphography	10.85	12.40	14.60	12.20	9.25	11.25	11.35	10.30
PenglungEW	72.20	138.70	182.70	192.20	85.10	160.20	123.70	127.50
Ranking (Wins)	1	0	0	0	4	0	0	1
Ranking (Ties)	0	1	0	0	0	1	0	0
Ranking (Losses)	6	6	7	7	3	6	7	6
Ranks (F-test)	2.7143	4.3571	7.5714	7.1429	2.1429	4.7857	3.8571	3.4286

In more complex datasets like Leukemia, which initially have thousands of features, the variation in selected features is stark, with mDA selecting 3332.30 features, substantially fewer than GWO and SSA but more than GOA and DA. This indicates that while mDA generally selects more features compared to some algorithms, it may balance feature reduction and maintaining classification performance.

Overall, this table effectively demonstrates how different optimization algorithms manage the trade-off between reducing the number of features and retaining sufficient information for accurate classification, providing insights into their applicability in various scenarios.

Table 6 summarizes the feature selection ratio and dimensionality reduction efficiency obtained by the proposed mDA across all benchmark datasets. The results show that mDA consistently achieves significant feature-space reduction, eliminating on average nearly half of the original features ($\approx 49\%$) while maintaining strong classification performance. In particular, for medium-dimensional datasets such as BreastEW and PenglungEW, mDA attains up to 60% reduction efficiency, highlighting its robustness in identifying highly relevant features. Although the reduction rate is moderate in high-dimensional datasets such as Colon and Leukemia ($\approx 53\%$), this behavior reflects the algorithm's tendency to preserve discriminative information essential for accurate classification. Overall, the results confirm that mDA provides an effective balance between compactness and accuracy, outperforming several baseline algorithms.

Table 6. Feature Selection Ratio and Dimensionality Reduction Efficiency using mDA Algorithm.

Dataset	Total-Features	Selected Features (mDA)	Feature-Selection Ratio (%)	Dimensionality Reduction Efficiency (%)
Breastcancer	9	5.00	55.56	44.44
BreastEW	30	13.60	45.33	54.67
Colon	2000	923.20	46.16	53.84
HeartEW	13	8.40	64.62	35.38
Leukemia	7129	3332.30	46.76	53.24
Lymphography	18	10.30	57.22	42.78
PenglungEW	325	127.50	39.23	60.77

Table 7 offers a comparative analysis of average best fitness values achieved by various algorithms when applied to different datasets in the research. This table shows the effectiveness of each algorithm based on their fitness values, a measure reflecting how well each algorithm has optimized a predefined objective function. For example, the Breastcancer dataset shows that mDA achieves an impressive fitness value of 0.0233, indicating its superior performance in optimizing the classification task compared to all other algorithms. Similarly, in the Leukemia dataset, mDA reports the lowest fitness value of 0.0064, suggesting a significant optimization capability over the alternatives.

However, in the Colon dataset, GA displays lower fitness values, indicating potential areas where mDA could be less effective or require adjustments to enhance its optimization performance. The trend across different datasets highlights the varying effectiveness of these algorithms in specific scenarios, offering insights into their potential applications and limitations.

Overall, this comparative analysis provides a clear perspective on the optimization capabilities of each algorithm, with mDA generally demonstrating robust performance across most datasets, underscoring its utility in solving complex classification problems effectively.

Furthermore, we used the F-test to statistically validate the difference in the performance obtained. The results of the F-test are displayed under each table. However, the comparative performance is summarized in each table using the Wins–Ties–Losses (WTL) criterion. More wins are an indication of superior overall performance consistency.

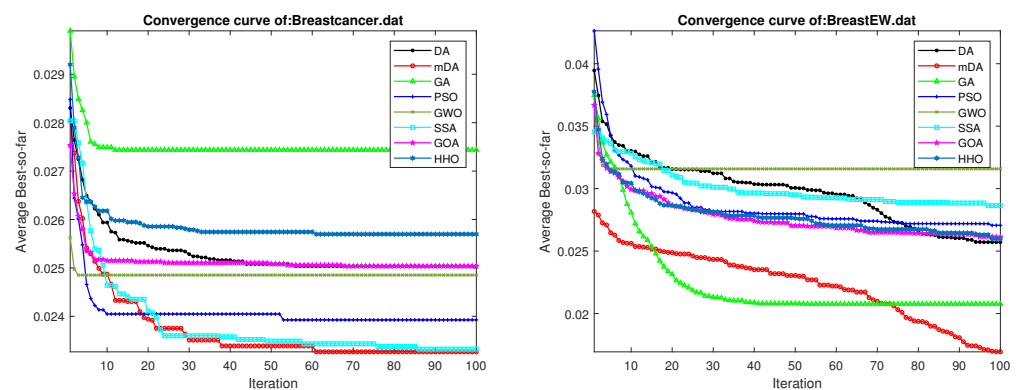
Figure 1 depicts the convergence curves for the standard (DA) and the modified DA (mDA) over a series of iterations applied to the given dataset. The x-axis represents the number of iterations, ranging from 0 to 100, while the y-axis measures the average best-so-far fitness values, which indicate the optimization performance of each algorithm at each iteration.

Table 7. Comparison of the average best fitness values between the proposed mDA and other methods.

Benchmark	GA	PSO	GWO	SSA	GOA	HHO	DA	mDA
Breastcancer	0.0274 ± 0.0009	0.0239 ± 0.0029	0.0249 ± 0.005	0.0233 ± 0.0115	0.0250 ± 0.0085	0.0257 ± 0.0063	0.0250 ± 0.0001	0.0233 ± 0.0017
BreastEW	0.0208 ± 0.0016	0.0271 ± 0.0035	0.0316 ± 0.0086	0.0286 ± 0.0074	0.0261 ± 0.0018	0.0260 ± 0.0027	0.0257 ± 0.0054	0.0169 ± 0.002
Colon	0.0459 ± 0.0086	0.1229 ± 0.0094	0.1341 ± 0.0159	0.1068 ± 0.0015	0.0699 ± 0.0006	0.0726 ± 0.0072	0.0872 ± 0.0066	0.0763 ± 0.0122
HeartEW	0.1077 ± 0.0004	0.0963 ± 0.0001	0.1142 ± 0.0015	0.1046 ± 0.0067	0.0920 ± 0.0038	0.0979 ± 0.0136	0.1038 ± 0.0131	0.1023 ± 0.0001
Leukemia	0.0154 ± 0.0019	0.0187 ± 0.0072	0.0939 ± 0.004	0.0559 ± 0.0076	0.0392 ± 0.0096	0.0441 ± 0.0001	0.0278 ± 0.002	0.0064 ± 0.0027
Lymphography	0.0652 ± 0.0026	0.0765 ± 0.0053	0.0878 ± 0.0145	0.0601 ± 0.0003	0.0781 ± 0.0021	0.0709 ± 0.0019	0.0650 ± 0.0082	0.0686 ± 0.0052
PenglungEW	0.0526 ± 0.0083	0.0469 ± 0.013	0.1038 ± 0.0001	0.0409 ± 0.0045	0.0376 ± 0.0061	0.0459 ± 0.0088	0.0388 ± 0.0054	0.0381 ± 0.0003
Ranking (Wins)	1	0	0	1	2	0	0	3
Ranking (Ties)	0	0	0	0	0	0	0	0
Ranking (Losses)	6	7	7	6	5	7	7	4
Ranks (F-test)	4.2857	4.7143	7.4286	4.6429	3.7857	4.7143	3.9286	2.5000

From the graph, it is evident that mDA (shown in red) starts with higher fitness values than traditional DA (shown in black), and maintains a consistent improvement, demonstrating a steeper and more continuous decline in the fitness values as the iterations progress. This suggests that mDA converges more effectively towards the optimum solution than DA in most cases, showing a more gradual decline in fitness values. By iteration 100, mDA achieves significantly lower fitness values than DA, indicating better optimization performance for all datasets.

The convergence curve for mDA is relatively smoother and steeper, which underscores its enhanced capability to quickly and efficiently refine solutions, likely due to improved algorithmic modifications that enhance its search and optimization processes within the feature space of the datasets. These graphs effectively illustrate the comparative advantage of mDA in achieving lower fitness values faster, which is indicative of better overall performance in optimizing the classification task for this particular dataset.

**Figure 1.** Cont.

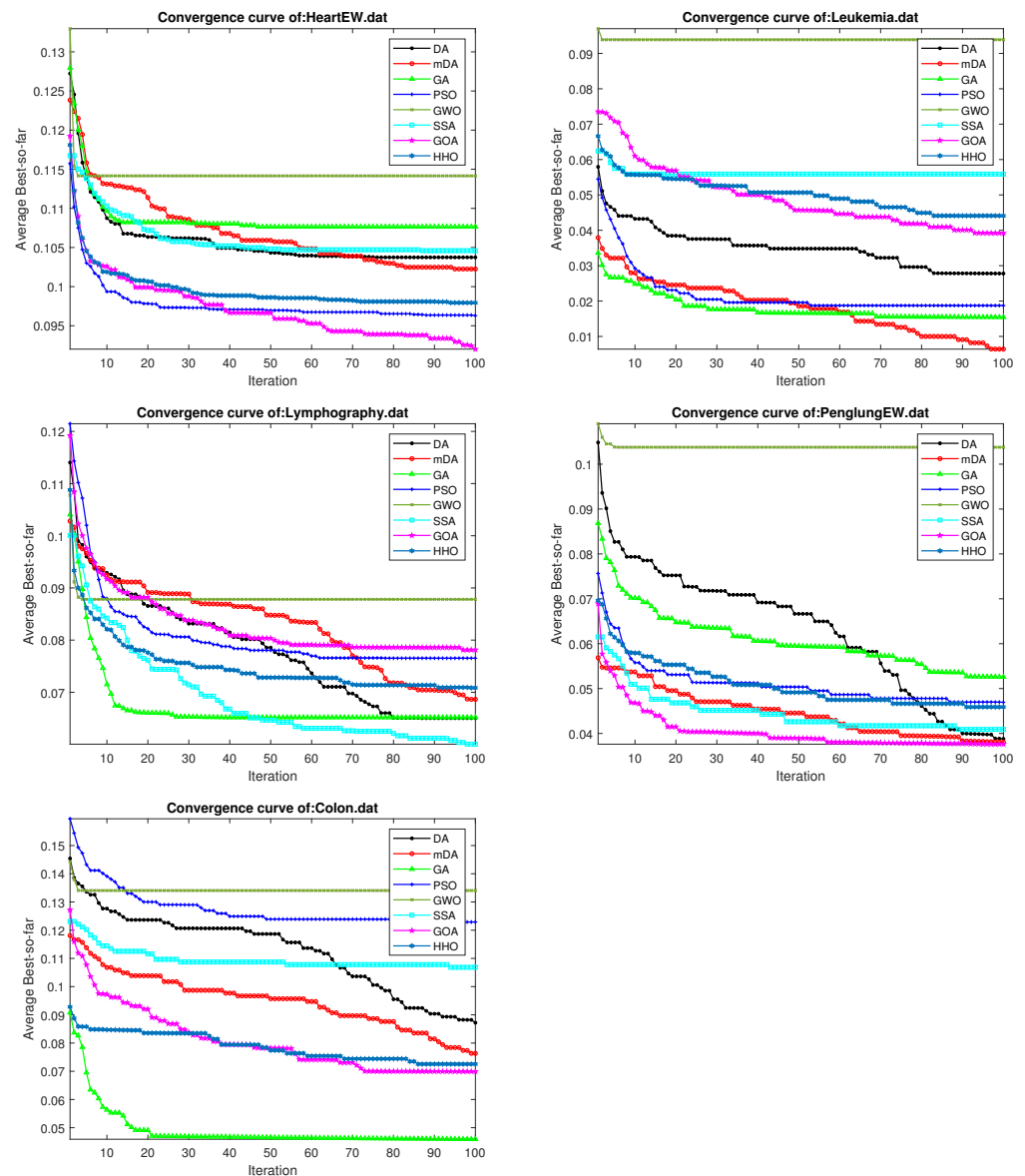


Figure 1. Convergence curves for the tested algorithms over the tested datasets.

To statistically assess the difference in performance between the algorithms compared, we employed a range of non-parametric tests that can be applied to multiple algorithms and datasets. The first step was to test the hypothesis that the performances of all algorithms are equivalent using the Friedman test, which evaluates the null hypothesis that all algorithms achieve equivalent performance across datasets. This test is appropriate here because it is rank-based and does not rely on normality assumptions. As shown in Table 8, the Friedman statistic was $\chi^2 = 201.465$ with 7 degrees of freedom, resulting in an extremely small p -value (5.618×10^{-40}). This strongly rejects the null hypothesis and confirms that there are significant performance differences between the algorithms.

Table 8. Friedman test for differences among algorithms across all datasets and runs.

Test	Value	df
χ^2 statistic	201.465	7
p -value	5.618×10^{-40}	—

Then we computed the average rank of each algorithm across all datasets, presented in Table 9. The proposed mDA algorithm achieved the lowest average rank (2.304), indicating the most consistent superior performance relative to the other baseline methods.

Table 9. Average ranks of the algorithms across all datasets and runs (lower is better).

Algorithm	Average Rank
mDA	2.304
BDA	3.732
BPSO	4.196
bGWO	4.511
BGA	4.896
BSSA	5.321
BGOA	5.421
HHO	5.618

To identify which performance differences were statistically significant at the pairwise level, a Nemenyi post-hoc test was performed. The results in Table 10 show that mDA is significantly better than all other algorithms at the $\alpha = 0.05$ significance level, with all p -values being extremely small. This confirms that the superiority of mDA is not due to random fluctuations but reflects genuine performance advantages across benchmarks.

Table 10. Nemenyi post-hoc test results comparing the proposed mDA to other algorithms (significance at $\alpha = 0.05$).

Algorithm	p -Value	Significant ($\alpha = 0.05$)
BGA	1.110223×10^{-16}	Yes
BPSO	2.827673×10^{-9}	Yes
bGWO	1.327716×10^{-12}	Yes
BSSA	0	Yes
BGOA	0	Yes
HHO	0	Yes
BDA	2.917804×10^{-5}	Yes

The Critical Difference (CD) diagram in Figure 2 provides a visual representation of these statistical findings. Algorithms whose rank differences exceed the CD threshold (0.97) are considered significantly different. As depicted, mDA is clearly separated from all other methods and lies well outside the CD interval of competing approaches, reinforcing the conclusion that its performance advantage is statistically significant.

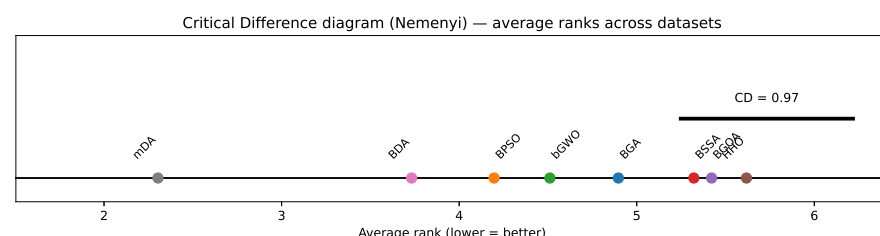


Figure 2. Critical Difference (Nemenyi) diagram showing the average ranks of the algorithms.

Taken all together, the Friedman test result, the result of the post-hoc test using the Nemenyi method, and the CD diagram provide a strong indication that the new mDA algorithm performs better than all of the baselines. The above findings provide strong evidence for the reliability of the advantages of the mDA algorithm.

Moreover, a Filter Feature Selection (FFS-) technique was performed for further analysis. Table 11 compares the proposed approach and various classification methods combined with filter feature selection. The results illustrate the proposed approach's superior performance on most datasets. For instance, the mDA algorithm achieved the best results on five out of seven datasets: breast cancer, BreastEW, Heart, and PenglungEW, with accuracies of 0.9678, 0.9382, 0.8227, 0.7627, and 0.6035, respectively. As for the remaining datasets, Colon and Leukaemia achieved the highest accuracies of 0.9838 and 0.8611 using J48 and AdaBoost, respectively. This discrepancy arises because the Colon and Leukaemia datasets may possess specific characteristics that do not align well with the strengths of the mDA algorithm. These characteristics could include different distributions of features, higher levels of noise, or imbalanced class distributions, which the mDA algorithm might not handle as effectively as other algorithms like J48 or AdaBoost.

Table 11. Classification accuracy results comparing the proposed mDA algorithm with other algorithms using filter feature selection.

	FFS-J48	FFS-RF	FFS-k-NN	FFS-AdaBoost	FFS-Bagging	mDA
Breast-cancer	0.7428	0.6974	0.7284	0.7162	0.6841	0.9678
BreastEW	0.6296	0.6301	0.7305	0.6269	0.6465	0.9382
Colon	0.9838	0.9516	0.9193	0.9677	0.8870	0.7174
Heart	0.7814	0.7962	0.7614	0.8159	0.8155	0.8227
Leukaemia	0.8198	0.7297	0.6527	0.8611	0.6527	0.8211
Lymphography	0.7432	0.6486	0.7364	0.7498	0.7229	0.7627
PenglungEW	0.4505	0.5616	0.5890	0.4931	0.4109	0.6035

6. Clinical Discussion

Beyond the quantitative improvements achieved by the proposed mDA algorithm across the seven medical benchmarks, we further examined the clinical relevance of the most frequently selected features to ensure that the selected subsets align with established biomedical understanding.

For the **Breastcancer** and **BreastEW** datasets, the selected attributes consistently highlighted morphological and cytological descriptors such as cell uniformity, epithelial cell size, bare nuclei, and chromatin texture. These indicators have a correspondence with histopathologic criteria used by pathologists for distinguishing between benign and malignant lesions and for estimating tumor aggressiveness, thus verifying that the algorithm can identify salient features for diagnosis.

In the **HeartEW** dataset, the retained variables primarily encompassed cardiorespiratory and electrocardiographic markers—including chest pain type, ST-segment slope, and exercise induced responses, that are well-established predictors of ischemia and cardiovascular risk in coronary artery disease (CAD). The resulting can mirror the features commonly employed in clinical scoring systems for non-invasive cardiac diagnosis.

For the **ColonEW** and **LeukemiaEW** gene-expression benchmarks, mDA identified compact and biologically interpretable gene signatures associated with pathways in cell-cycle control, apoptosis regulation, and hematopoietic proliferation. These subsets capture known genomic signals implicated in oncogenic progression, reflecting the model's capacity to balance sparsity with biological coherence.

In the **Lymphography** dataset, the selected variables predominantly involved patterns of lymph node enlargement, capsularity, and structural irregularity, which correspond to the clinical staging criteria used in lymphoma assessment. Such features directly relate to tumor dissemination and disease extent, indicating that mDA emphasizes anatomically meaningful discriminants.

Finally, for the multi-class **PenglungEW** dataset, the retained subset covered heterogeneous cytological and morphological indicators that capture inter-subtype variation across lung cancer classes. The inclusion of a slightly larger feature set here reflects the need to preserve separability across diverse histopathological subtypes. Together, these findings demonstrate that the mDA optimizer systematically gravitates toward clinically coherent and biologically plausible feature sets rather than arbitrary statistical artifacts. This not only enhances interpretability but also reinforces the practical translational value of the proposed framework in medical diagnostic contexts.

7. Conclusions and Future Works

This research introduces an enhanced hybrid DA optimizer incorporating EPD, aimed at boosting the performance of the standard DA in handling FS tasks. The mDA methodology was extensively applied to seven medical datasets. Detailed comparisons of the mDA's overall classification accuracy, selected features, fitness, and convergence characteristics were made against several well-known metaheuristic-based techniques. The exhaustive comparative results and analysis demonstrated the superior effectiveness of the proposed algorithm for various FS tasks within the medical field.

The conducted experiments confirm the proposed mDA algorithm's efficacy for feature selection within medical fields. Its capability to attain higher accuracy with a reduced number of features highlights its potential as a valuable tool for medical data analysis, providing a balance between clarity and predictive performance.

Future research could explore applying the EPD approach to various other population-based optimization algorithms. The effectiveness of the proposed binary DA and EPD-based techniques could also be utilized to address additional data-mining challenges. In future work, we plan to compare the proposed mDA with various FS methods within the domain. Furthermore, the proposed binary Dragonfly Algorithm (DA) and EPD-based techniques can be effectively utilized to tackle other complex data-mining challenges, especially those dealing with high-dimensional, unstructured, or heterogeneous datasets. As an illustration, future research could investigate areas such as LASSO and ridge regression, image-based feature selection, multimodal data analysis, and other intricate fields to demonstrate the scalability and practical applicability of the proposed methods. Future work should not only focus on technical advancements but also address significant socio-technical and ethical open research questions (ORQs) highlighted by this study and related literature. These encompass concerns such as transparency, fairness, responsible usage, computational sustainability, and the impact of human–AI interaction in the deployment of EPD- and DA-based algorithms.

Author Contributions: I.A.: Conceptualization, Methodology, Supervision, Project administration; I.A. and A.A.: Data curation, Formal analysis, Software; I.A., A.A., N.A.-M. and A.M.A.-Z.: Writing—review and editing, validation; A.S.: Software, Resources, Writing—review & editing. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Acknowledgments: This research has been done during Ibrahim Aljarah's sabbatical leave from the University of Jordan for the academic year 2023–2024, and we express our gratitude to the University of Jordan for its support in enabling this research.

Conflicts of Interest: The authors declare no conflicts of interest.

References

- Eshtay, M.; Faris, H.; Obeid, N. Improving Extreme Learning Machine by Competitive Swarm Optimization and its application for medical diagnosis problems. *Expert Syst. Appl.* **2018**, *104*, 134–152. [\[CrossRef\]](#)
- Khalilpourazari, S.; Pasandideh, S.H.R.; Niaki, S.T.A. Optimization of multi-product economic production quantity model with partial backordering and physical constraints: SQP, SFS, SA, and WCA. *Appl. Soft Comput.* **2016**, *49*, 770–791. [\[CrossRef\]](#)
- Zelinka, I. A survey on evolutionary algorithms dynamics and its complexity–Mutual relations, past, present and future. *Swarm Evol. Comput.* **2015**, *25*, 2–14. [\[CrossRef\]](#)
- Heidari, A.A.; Abbaspour, R.A.; Jordehi, A.R. Gaussian bare-bones water cycle algorithm for optimal reactive power dispatch in electrical power systems. *Appl. Soft Comput.* **2017**, *57*, 657–671. [\[CrossRef\]](#)
- Heidari, A.A.; Abbaspour, R.A.; Jordehi, A.R. An efficient chaotic water cycle algorithm for optimization tasks. *Neural Comput. Appl.* **2017**, *28*, 57–85. [\[CrossRef\]](#)
- Mirjalili, S. Dragonfly algorithm: A new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. *Neural Comput. Appl.* **2016**, *27*, 1053–1073. [\[CrossRef\]](#)
- Bhesdadiya, R.H.; Pandya, M.H.; Trivedi, I.N.; Jangir, N.; Jangir, P.; Kumar, A. Price penalty factors based approach for combined economic emission dispatch problem solution using Dragonfly Algorithm. In *Proceedings of the 2016 International Conference on Energy Efficient Technologies for Sustainability (ICEETS), Nagercoil, India, 7–8 April 2016*; IEEE: New York, NY, USA, 2016; pp. 436–441. [\[CrossRef\]](#)
- Suresh, V.; Sreejith, S. Generation dispatch of combined solar thermal systems using dragonfly algorithm. *Computing* **2017**, *99*, 59–80. [\[CrossRef\]](#)
- Hamdy, M.; Nguyen, A.T.; Hensen, J.L.M. A performance comparison of multi-objective optimization algorithms for solving nearly-zero-energy-building design problems. *Energy Build.* **2016**, *121*, 57–71. [\[CrossRef\]](#)
- Daely, P.T.; Shin, S.Y. Range based wireless node localization using Dragonfly Algorithm. In *Proceedings of the 2016 Eighth International Conference on Ubiquitous and Future Networks (ICUFN), Vienna, Austria, 5–8 July 2016*; IEEE: New York, NY, USA, 2016; pp. 1012–1015. [\[CrossRef\]](#)
- Elhariri, E.; El-Bendary, N.; Hassanien, A.E. Bio-inspired optimization for feature set dimensionality reduction. In *Proceedings of the 2016 3rd International Conference on Advances in Computational Tools for Engineering Applications (ACTEA), Zouk Mosbeh, Lebanon, 13–15 July 2016*; IEEE: New York, NY, USA, 2016; pp. 184–189. [\[CrossRef\]](#)
- Salam, M.A.; Zawbaa, H.M.; Emary, E.; Ghany, K.K.A.; Parv, B. A hybrid dragonfly algorithm with extreme learning machine for prediction. In *Proceedings of the 2016 International Symposium on INnovations in Intelligent SysTems and Applications (INISTA), Sinaia, Romania, 2–5 August 2016*; IEEE: New York, NY, USA, 2016; pp. 1–6. [\[CrossRef\]](#)
- Kanan, T.; AbedAlghafer, A.; AlZu'bi, S.; Hawashin, B.; Mughaid, A.; Kanaan, G.; Kamruzzaman, M. An Intelligent Health Care System for Detecting Drug Abuse in Social Media Platforms Based on Low Resource Language. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2023**, *32*, 691–703. [\[CrossRef\]](#)
- Kanan, T.; Hawashin, B.; Alzubi, S.; Almaita, E.; Alkhatib, A.; Maria, K.A.; Elbes, M. Improving arabic text classification using p-stemmer. *Recent Adv. Comput. Sci. Commun. (Former. Recent Patents Comput. Sci.)* **2022**, *15*, 404–411. [\[CrossRef\]](#)
- Mansour, A.M.; Obeidat, M.A.; Hawashin, B. A novel multi agent recommender system for user interests extraction. *Clust. Comput.* **2023**, *26*, 1353–1362. [\[CrossRef\]](#)
- Mafarja, M.M.; Eleyan, D.; Jaber, I.; Hammouri, A.; Mirjalili, S. Binary dragonfly algorithm for feature selection. In *Proceedings of the 2017 International Conference on New Trends in Computing Sciences (ICTCS), Amman, Jordan, 11–13 October 2017*; IEEE: New York, NY, USA, 2017; pp. 12–17. [\[CrossRef\]](#)
- Raman, G.; Raman, G.; Manickam, C.; Ganesan, S.I. Dragonfly algorithm based global maximum power point tracker for photovoltaic systems. In *Proceedings of the International Conference in Swarm Intelligence, Brussels, Belgium, 7–9 September 2016*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 211–219. [\[CrossRef\]](#)
- Hema, C.; Sankar, S.; Sandhya. Energy efficient cluster based protocol to extend the RFID network lifetime using dragonfly algorithm. In *Proceedings of the 2016 International Conference on Communication and Signal Processing (ICCSP), Melmaruvathur, India, 6–8 April 2016*; IEEE: New York, NY, USA, 2016; pp. 530–534. [\[CrossRef\]](#)
- Abdel-Basset, M.; Luo, Q.; Miao, F.; Zhou, Y. Solving 0–1 knapsack problems by binary dragonfly algorithm. In *Proceedings of the International Conference on Intelligent Computing, Liverpool, UK, 7–10 August 2017*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 491–502. [\[CrossRef\]](#)
- KS, S.R.; Murugan, S. Memory based hybrid dragonfly algorithm for numerical optimization problems. *Expert Syst. Appl.* **2017**, *83*, 63–78. [\[CrossRef\]](#)
- Song, J.; Li, S. Elite opposition learning and exponential function steps-based dragonfly algorithm for global optimization. In *Proceedings of the 2017 IEEE International Conference on Information and Automation (ICIA), Macao, China, 18–20 July 2017*; IEEE: New York, NY, USA, 2017; pp. 1178–1183. [\[CrossRef\]](#)

22. Majeed, N.M.; Ramo, F.M. Implementation of Features Selection Based on Dragonfly Optimization Algorithm. *Technium* **2022**, *4*, 44–52. [\[CrossRef\]](#)
23. Raouf, Z.T.; Abd, D.H. Feature Selection for Binary Dataset using Dragonfly Algorithm. In *Proceedings of the 2023 16th International Conference on Developments in eSystems Engineering (DeSE), Istanbul, Türkiye, 18–20 December 2023*; IEEE: New York, NY, USA, 2023; pp. 480–485. [\[CrossRef\]](#)
24. Sarvamangala, D.; Kulkarni, R.V. A comparative study of bio-inspired algorithms for medical image registration. In *Advances in Intelligent Computing*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 27–44. [\[CrossRef\]](#)
25. Díaz-Cortés, M.A.; Ortega-Sánchez, N.; Hinojosa, S.; Oliva, D.; Cuevas, E.; Rojas, R.; Demin, A. A multi-level thresholding method for breast thermograms analysis using Dragonfly algorithm. *Infrared Phys. Technol.* **2018**, *93*, 346–361. [\[CrossRef\]](#)
26. Song, X.; Zhang, Y.; Zhang, W. Evolutionary computation for feature selection in classification: A comprehensive survey of solutions, applications and challenges. *Swarm Evol. Comput.* **2024**, *90*, 101661. [\[CrossRef\]](#)
27. Zhang, Y.; Wang, J.; Gorriz, J.M.; Wang, S. Deep Learning and Vision Transformer for Medical Image Analysis. *J. Imaging* **2023**, *9*, 147. [\[CrossRef\]](#)
28. Chaddad, A.; Peng, J.; Xu, J.; Bouridane, A. Survey of Explainable AI Techniques in Healthcare. *Sensors* **2023**, *23*, 634. [\[CrossRef\]](#)
29. Feng, J.; Zhou, H.; Xu, W.; Liu, Q. Sliding-attention transformer neural architecture for predicting T cell receptor–antigen–human leucocyte antigen binding. *Nat. Mach. Intell.* **2024**, *6*, 981–994. [\[CrossRef\]](#)
30. Kababulut, F.Y.; Gürkan Kuntalp, D.; Düzyel, O.; Özcan, N.; Kuntalp, M. A New Shapley-Based Feature Selection Method in a Clinical Decision Support System for the Identification of Lung Diseases. *Diagnostics* **2023**, *13*, 3558. [\[CrossRef\]](#)
31. Hur, S.; Lee, Y.; Park, J.; Jeon, Y.J.; Cho, J.H.; Cho, D.; Lim, D.; Hwang, W.; Cha, W.C.; Yoo, J. Comparison of SHAP and clinician friendly explanations reveals effects on clinical decision behaviour. *npj Digit. Med.* **2025**, *8*, 578. [\[CrossRef\]](#)
32. Reynolds, C.W. Flocks, herds and schools: A distributed behavioral model. *ACM Siggraph Comput. Graph.* **1987**, *21*, 25–34. [\[CrossRef\]](#)
33. Mirjalili, S.; Lewis, A. S-shaped versus V-shaped transfer functions for binary Particle Swarm Optimization. *Swarm Evol. Comput.* **2013**, *9*, 1–14. [\[CrossRef\]](#)
34. Mirjalili, S.; Wang, G.G.; Coelho, L.d.S. Binary optimization using hybrid particle swarm optimization and gravitational search algorithm. *Neural Comput. Appl.* **2014**, *25*, 1423–1435. [\[CrossRef\]](#)
35. Bak, P.; Tang, C.; Wiesenfeld, K. Self-organized criticality: An explanation of the 1/f noise. *Phys. Rev. Lett.* **1987**, *59*, 381. [\[CrossRef\]](#)
36. Lewis, A.; Mostaghim, S.; Randall, M. Evolutionary population dynamics and multi-objective optimisation problems. In *Multi-Objective Optimization in Computational Intelligence: Theory and Practice*; IGI Global: Hershey, PA, USA, 2008; pp. 185–206. [\[CrossRef\]](#)
37. Lewis, A.; Abramson, D.; Peachey, T. An evolutionary programming algorithm for automatic engineering design. In *Proceedings of the International Conference on Parallel Processing and Applied Mathematics, Czestochowa, Poland, 7–10 September 2003*; Springer: Berlin/Heidelberg, Germany, 2003; pp. 586–594. [\[CrossRef\]](#)
38. Boettcher, S.; Percus, A. Extremal optimization: Methods derived from co-evolution. *arXiv* **1999**, arXiv:math/9904056. [\[CrossRef\]](#)
39. Altman, N.S. An introduction to kernel and nearest-neighbor nonparametric regression. *Am. Stat.* **1992**, *46*, 175–185. [\[CrossRef\]](#)
40. Emary, E.; Zawbaa, H.M.; Hassanien, A.E. Binary antlion approaches for feature selection. *Neurocomputing* **2016**, *213*, 54–65. [\[CrossRef\]](#)
41. Zawbaa, H.M.; Emary, E.; Parv, B.; Sharawi, M. Feature selection approach based on moth-flame optimization algorithm. In *Proceedings of the 2016 IEEE Congress on Evolutionary Computation (CEC), Vancouver, BC, Canada, 24–29 July 2016*; IEEE: New York, NY, USA, 2016; pp. 4612–4617. [\[CrossRef\]](#)
42. Alzaqebah, A.; Al-Kadi, O.; Aljarah, I. An enhanced harris hawk optimizer based on extreme learning machine for feature selection. *Prog. Artif. Intell.* **2023**, *12*, 77–97. [\[CrossRef\]](#)
43. Mafarja, M.; Aljarah, I.; Heidari, A.A.; Faris, H.; Fournier-Viger, P.; Li, X.; Mirjalili, S. Binary dragonfly optimization for feature selection using time-varying transfer functions. *Knowl.-Based Syst.* **2018**, *161*, 185–204. [\[CrossRef\]](#)
44. Hammouri, A.I.; Awadallah, M.A.; Braik, M.S.; Al-Betar, M.A.; Beseiso, M. Improved Dwarf Mongoose Optimization Algorithm for Feature Selection: Application in Software Fault Prediction Datasets. *J. Bionic Eng.* **2024**, *21*, 2000–2033. [\[CrossRef\]](#)
45. Mafarja, M.; Mirjalili, S. Whale optimization approaches for wrapper feature selection. *Appl. Soft Comput.* **2018**, *62*, 441–453. [\[CrossRef\]](#)
46. Mafarja, M.; Aljarah, I.; Faris, H.; Hammouri, A.I.; Ala'M, A.Z.; Mirjalili, S. Binary grasshopper optimisation algorithm approaches for feature selection problems. *Expert Syst. Appl.* **2019**, *117*, 267–286. [\[CrossRef\]](#)
47. Alzaqebah, A.; Aljarah, I.; Al-Kadi, O. A hierarchical intrusion detection system based on extreme learning machine and nature-inspired optimization. *Comput. Secur.* **2023**, *124*, 102957. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.