STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **15**, January 2026, pp 443–459.
Published online in International Academic Press (www.IAPress.org)

A High-dimensional Feature Selection Based on Modified Energy Valley Optimizer and ReliefF Algorithm

Islam S. Fathi [1,*], Bajeszeyadaljunaeidia [1], Mohammed Tawfik [2]

[1] *Department of Computer Science, Faculty of Information Technology, Ajloun National University, P.O.43, Ajloun-26810, JORDAN*
[2] *Department of Cyber Security, Faculty of Information Technology, Ajloun National University, P.O.43, Ajloun-26810, JORDAN*

Abstract A high-dimensional feature selection represents a crucial preprocessing phase in data mining and machine learning applications, exerting substantial influence on the effectiveness of machine learning algorithms. The primary goal of FS involves removing irrelevant attributes, minimizing computational time and memory demands, and improving the overall efficacy of the associated learning algorithm. The Energy Valley Optimizer (EVO) constitutes an innovative metaheuristic approach grounded in sophisticated physics concepts, specifically those connected to particle equilibrium and decomposition patterns. This research introduces an improved binary variant of The Energy Valley Optimizer (IBEVO) designed to tackle a high-dimensional feature selection challenges. The base EVO algorithm has been augmented with significant enhancement to boost its comprehensive effectiveness. ReliefF algorithm represents an addition incorporated into the EVO's initialization phase to strengthen the algorithm's capacity to utilize its potential in addition to, it integrated into the base EVO, accelerating convergence rates. The findings from ten gene expression datasets characterized by high dimensionality and limited sample sizes show that the newly developed method enhances predictive performance while simultaneously decreasing feature count, achieving highly competitive outcomes when compared to other state-of-the-art feature selection approaches.

Keywords Energy Valley Optimizer, High-dimensional data, Feature Selection, Search Space Reduction, ReliefF algorithm

DOI: 10.19139/soic-2310-5070-2862

1. Introduction

The rapid advancement of computing and web technologies has resulted in the creation of enormous datasets containing numerous attributes. The strategic identification of pertinent and beneficial characteristics can significantly influence various domains, including machine learning [1], text analysis [2], Internet of things [3], Bioinformatics [4, 5], and industrial implementations [6, 7, 8].Bio-inspired optimization has shown effectiveness in IoT applications, including fog node placement [9] and intrusion detection [10], demonstrating superior performance in resource-constrained environments where traditional methods face challenges in balancing competing optimization objectives. For instance, within machine learning contexts, the existence of superfluous elements in multi-dimensional datasets diminishes classification precision and elevates processing expenses [11]. IoT applications commonly face difficulties in managing and analyzing vast quantities of sensor-generated information. An additional obstacle concerns the existence of unrelated and duplicated characteristics. A preliminary processing step, such as attribute selection, becomes necessary to address multi-dimensional data issues and remove unneeded or repetitive elements [12]. Attribute selection represents a fundamental component in data preprocessing that serves a crucial function in developing reliable models. This process encompasses identifying and choosing the most valuable characteristics from existing datasets.

*Correspondence to: Islam S. Fathi (Email: i.mohamed@anu.edu.jo).

ISSN 2310-5070 (online) ISSN 2311-004X (print)

An attribute selection framework consists of three core components: classification methods like support vector machines (SVMs) [13], k-nearest neighbor (KNN) [14], assessment metrics, and the search methodology employed to identify optimal characteristics. FS approaches can be broadly divided into two main categories: wrapper and filter methodologies. In wrapper techniques, feature subset assessment relies on classification algorithm effectiveness. Wrapper methods employ classification algorithms as standalone components, enabling quality evaluation of chosen subsets based on their performance [15]. Filter approaches operate independently of learning models, assessing attribute subsets by examining data properties without considering the specific model utilized. Filter techniques function autonomously from learning models and evaluate attribute subsets based on data characteristics, irrespective of the particular algorithm implemented. Nevertheless, it should be noted that filter methods may not consistently identify the most effective feature combination. Conversely, wrapper approaches generally deliver the most suitable attribute subset for specific classifiers regarding performance metrics [16].

An FS methodology seeks to discover the ideal attribute combination from all available subset possibilities. Two primary categories of search algorithms exist: precise search techniques and metaheuristic [17]. Precise search approaches examine attribute sets containing k characteristics; the search space magnitude correlates directly with 2^k. Therefore, this demands substantial computational resources. Metaheuristic algorithms demonstrate probabilistic characteristics by beginning optimization through random solution generation to efficiently navigate search spaces. Implementing metaheuristic for addressing attribute selection challenges relies on their capacity to produce near-optimal solutions within acceptable timeframes [18]. Metaheuristic demonstrate high adaptability to particular problem areas due to their straightforward nature and implementation ease. A notable characteristic of these algorithms involves preventing early convergence while maintaining equilibrium between exploration and exploitation—two essential elements. In this research, the energy valley optimizer (EVO) is improved (IBEVO) for enhanced attribute selection. Improvements encompass the Laplace crossover approach for superior exploitation and a random substitution technique for accelerated convergence.

2. Related Work

Metaheuristic methodologies are generally organized into four separate classifications, distinguished by their inspirational sources: human-inspired techniques [19], collective intelligence [20], evolutionary computation [21], and physics-derived approaches [22].

Agrawal and colleagues [23] introduced an innovative binary adaptation of the knowledge-based gaining sharing method (GSK) for tackling FS challenges, designated as FSNBGSK. The KNN classifier was utilized for assessment across 23 datasets from the UCI collection. The developed approach exhibited enhanced performance relative to alternative algorithms regarding classification precision. Several instances of human-inspired methodologies include imperial competition algorithms (ICA) [24], the cultural evolution algorithm (CEA) [25], the volleyball premier league (VPL) [26], and teaching-learning-based optimization (TLBO) [19]. Algorithm hybridization has become a favored strategy within FS research; this classification enables investigators to harness the distinctive advantages of various algorithms [22].

Collective intelligence methodologies draw motivation from group animal behaviors in colonies. The techniques examined include Binary Horse Herd Optimization (BinHOA) [27], Binary Cuckoo Search (BCS) [28], Binary Dragonfly algorithm (BDA) [29], and Binary Flower Pollination Algorithm (BFPA) [30]. The Particle Swarm Optimization (PSO) algorithm has attracted considerable researcher interest since its development. Xue and colleagues implemented novel initialization and updating strategies for PSO to reduce processing time, decrease attribute count, and enhance classification precision [31]. Furthermore, Al-Tashi, Qasem, and others [32] developed a binary hybridization approach based on WOA. In the initial framework, the simulated annealing (SA) algorithm integrates within the WOA structure, while the subsequent framework employs SA to enhance the optimal solution achieved after each iteration. Results indicate that the presented algorithm surpasses existing binary approaches in both precision and processing efficiency, with experiments conducted on 18 UCI standard datasets.

Evolutionary computation mimics natural selection principles, drawing from Darwinian concepts. The genetic algorithm (GA), a particular evolutionary methodology, has achieved recognition for its remarkable capability to

address attribute selection challenges effectively [33]. Implementation results of nested-GA showed substantial enhancement in classification precision. For instance, applying GA algorithms in document classification [34]. Additional evolutionary approaches include differential evolution algorithms (DE) [35], geography-based optimizer [36], and stochastic fractal search [37].

Physics-derived approaches originate from fundamental natural physics principles and laws. These metaheuristic algorithms have contributed significantly to resolving attribute selection problems. Among these algorithms are lightning search algorithm (LSA) [38], multi-verse optimizer (MVO) [39], electromagnetic field optimization (EFO) [40], henry gas solubility optimization (HGSO) [41], and gravitational search algorithm (GSA) [42]. Additionally, SA [43] draws inspiration from metallurgical procedures. The equilibrium optimizer (EO) has become a prominent and noteworthy contribution to physics-based methodologies [44]. Ahmed and colleagues [45] developed an enhanced equilibrium optimizer version, employing automata and U-shaped transfer functions, to address attribute selection problems. KNN evaluated the approach on 18 datasets, comparing it against eight established methods, including classical and hybrid meta-heuristic algorithms. D. A. Elmanakhly and others introduced a binary equilibrium optimizer variant, called BinEO. This version builds upon opposition-based learning methodology and incorporates local search algorithms [1]. KNN classifiers represented the most commonly employed wrapper approaches. Results confirm BinEO's effectiveness. Nebojsa and colleagues [46] presented a Novel Chaotic Firefly Algorithm for addressing Global Optimization Problems: Application for Dropout Regularization. Samir Malakar and others [47] introduced a GA-based hierarchical attribute selection method for handwritten word recognition. Luka Gajic and colleagues [48] presented Multi-layer Perceptron Training Using Hybridized Bat Algorithm, noting that neural networks differ from other machine learning algorithms by not relying on statistical and mathematical models for future predictions.

EVO [49] represents a recently developed metaheuristic algorithm inspired by advanced physics concepts regarding particle equilibrium and decomposition patterns. The EVO concept originates from fundamental physics laws governing particle decay through various matter types. EVO also incorporates complexity analysis of test functions and achieves outstanding results. As previously mentioned, metaheuristics have demonstrated beneficial effects on attribute selection problems recently. Additional optimization strategies remain necessary to accomplish further enhanced outcomes. Despite comprehensive research, numerous metaheuristics continue facing various challenges requiring resolution, such as local optima trapping and excessive time consumption. EVO possesses particular benefits including parameter-free operation and rapid convergence characteristics. This motivated establishing an improved binary EVO version as a binary optimization approach for attribute selection problems. Our contributions summarize as follows:

- IBEVO: A binary modified EVO algorithm version addresses attribute selection challenges.
- Laplace crossover methodology increases EVO population diversity during initial phases.
- Enhanced EVO exploitation capacity through random replacement strategy integration at each EVO iteration conclusion to prevent local optima occurrence.
- Algorithm effectiveness assessment through experiments on 20 widely recognized standard datasets.

This manuscript follows this structure: Section 2 briefly examines Energy valley optimizer, Section 3 presents the proposed IBEVO algorithm, Section 4 displays experiments, and Section 5 provides conclusions.

3. Energy Valley Optimizer

EVO represents a recently developed metaheuristic algorithm that has been applied to engineering optimization challenges. It belongs to the physics-derived methodology category. EVO draws inspiration from core physics principles governing particle decomposition through various material types. The concept of "physical reaction" refers to the mechanism of causing two particles or external subatomic elements to collide, leading to the creation of new particles. It is believed that while most particles exhibit instability, certain ones remain stable and maintain this condition indefinitely. The unstable elements release energy during their breakdown process, which is alternatively termed decomposition. Each particle type possesses a distinct total decomposition rate. A particle experiences

energy reduction, with the excess energy being discharged. The Energy Valley framework entails examining particle equilibrium through binding energy analysis and investigating how these particles influence one another.

3.1. Mathematical Formula

The initial stage encompasses implementing the setup process, wherein the potential solutions (P_i) are represented as particles possessing different levels of equilibrium within the exploration domain.

$$C = \begin{bmatrix} C_1 \\ C_2 \\ \vdots \\ C_i \\ \vdots \\ C_n \end{bmatrix} = \begin{bmatrix} c_1^1 c_1^2 \cdots c_1^j \cdots c_1^d \\ c_2^1 c_2^2 \cdots c_2^j \cdots c_2^d \\ \cdots \ddots \\ \cdots \\ c_i^1 c_i^2 \cdots c_i^j \cdots c_i^d \\ \cdots \vdots \\ c_n^1 c_n^2 \cdots c_n^j \cdots c_n^d \end{bmatrix}, \begin{cases} i = 1, 2, \ldots, n. \\ j = 1, 2, \ldots, dim. \end{cases} \tag{1}$$

$$c_i^j = c_{i,\min}^j + rand \cdot \left(c_{i,\max}^j - c_{i,\min}^j \right), \begin{cases} i = 1, 2, \ldots, n. \\ j = 1, 2, \ldots, dim. \end{cases} \tag{2}$$

The parameter "n" denotes the quantity of particles present in the exploration domain. The parameter "dim" signifies the dimensionality of the problem being examined. c_i^j represents the j-th decision parameter utilized for determining the initial location of the i-th solution candidate. $c_{i,\max}^j$ and $c_{i,\min}^j$ constitute the maximum and minimum boundaries of the j-th parameter for the i-th solution candidate. The parameter "rand" signifies a randomly generated value that adheres to a uniform probability distribution within the range [0, 1].

The subsequent stage of the approach entails establishing the Enrichment Boundary (EB) for the particles. This factor is utilized to account for differences between particles containing excess neutrons and particles experiencing neutron shortage. To achieve this objective, the assessment function is evaluated for each particle, resulting in the calculation of the Neutron Enrichment Level (NEL) of these particles. The previously mentioned components are expressed mathematically as follows:

$$EB = \frac{\sum_{i=1}^{n} NEL_i}{n}, \quad i = 1, 2, \ldots, n. \tag{3}$$

The parameter NEL_i represents the neutron enrichment degree for the i-th particle, whereas EB signifies the enrichment boundary of the particles.

During the third stage, the calculation of particle equilibrium values is performed through objective function assessment.

$$SL_i = \frac{NEL_i - BS}{WS - BS}, \quad i = 1, 2, \ldots, n. \tag{4}$$

The equilibrium value for the i-th particle is SL_i, which is established according to the optimal (BS) and poorest (WS) equilibrium levels. These levels match the minimum and maximum objective function values identified to date. Within EVO's primary exploration cycle, when a particle's neutron enrichment degree exceeds the enrichment threshold (EB), it indicates that the particle possesses an elevated neutron-to-proton (N/Z) proportion. Additionally, alpha and gamma decompositions are expected to transpire if the particle's equilibrium level surpasses the equilibrium boundary (SB). This anticipation originates from the probability of such decompositions in larger particles with heightened equilibrium levels. In alpha decomposition physics. The decision parameters present in the solution candidate are replaced by the emissions within the particle or candidate demonstrating the greatest equilibrium level, designated as P_{BS}. The mathematical expression of these is as follows:

$$C_i^{\text{New1}} = C_i \left(C_{BS} \left(C_i^j \right) \right), \quad \begin{cases} i = 1, 2, \ldots, n. \\ j = \text{Alpha Index II}. \end{cases} \tag{5}$$

Where C_i^{New1} represents a recently generated particle within the exploration domain, C_i is the present location vector of the i-th particle in the exploration domain. C_{BS} is the particle's location vector that possesses the greatest equilibrium level, is the j-th decision parameter.

In this framework, the calculation of the overall distance between the examined particle and remaining particles is conducted through a methodology, and the closest particle is chosen for this objective.

$$D_i^k = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}, \begin{cases} i = 1, 2, \ldots, n. \\ k = 1, 2, \ldots, n-1. \end{cases} \tag{6}$$

The parameter D_i^k represents the total distance between the i-th particle and the k-th adjacent particle. The locations of the particles are indicated by (x1, y1) and (x2, y2). The process of modifying the location to create the second solution candidate during this stage is executed through the following operations:

$$C_i^{\text{New2}} = C_i \left(C_{Ng} \left(C_i^j \right) \right), \quad \begin{cases} i = 1, 2, \ldots, n. \\ j = \text{Gamma Index II}. \end{cases} \tag{7}$$

Beta decomposition occurs in particles possessing reduced equilibrium levels, demonstrating diminished stability. Based on physics principles governing beta decomposition, particles emit β radiation to improve their equilibrium. Due to the instability of these particles, a substantial movement within the exploration domain becomes essential. Under these circumstances, a method is utilized to modify particle locations, incorporating directed displacement toward the particle or alternative with optimal equilibrium level (C_{BS}) and the particles' centroid (C_{CP}). This component of the algorithm replicates the particles' tendency to move closer to the equilibrium zone. Particles are positioned adjacent to this zone, with most of them displaying enhanced stability. These concepts are formulated as follows:

$$C_{CP} = \frac{\sum_{i=1}^{n} C_i}{n}, \quad i = 1, 2, \ldots, n. \tag{8}$$

$$C_i^{New1} = P_i + \frac{(r_1 \times C_{BS} - r_2 \times C_{CP})}{SL_i}, \quad i = 1, 2, \ldots, n. \tag{9}$$

The factors r_1 and r_2 signify two randomly produced integers within the interval of [0, 1]. To enhance the algorithm's exploitation and exploration capabilities, an alternative method is employed to modify the locations of particles utilizing beta decomposition. This approach involves directing the particles systematically toward the particle with superior equilibrium level (C_{BS}) and an adjacent particle or candidate (C_{Ng}), ensuring that the displacement remains unaffected by particle equilibrium level. These components can be mathematically expressed as follows:

$$C_i^{\text{New2}} = C_i + (r_3 \times C_{BS} - r_4 \times C_{Ng}), \quad i = 1, 2, \ldots, n. \tag{10}$$

The factors r_3 and r_4 signify two randomly produced integers within the interval of [0, 1]. When the neutron enrichment degree (NEL_i) of a particle falls beneath the enrichment threshold (EB), it is hypothesized that the particle exhibits a diminished neutron-to-proton (N/Z) proportion. To reach the equilibrium zone, the particle demonstrates a propensity to either absorb electrons or release positrons. In this framework, a probabilistic displacement within the exploration domain is defined to accommodate these movement types, formulated as follows:

$$C_i^{\text{New}} = C_i + r, \quad i = 1, 2, \ldots, n. \tag{11}$$

where C_i^{New} and P_i are the subsequent and present location vectors of the i-th particles. The factor represents a randomly produced integer within the interval of [0, 1]. The pseudo-code of the EVO is displayed in Fig. 1.

4. The Proposed Algorithm (IBEVO)

This segment provides a thorough description of the developed IBEVO, which represents a wrapper-based methodology created to address Feature Selection challenges. The primary phases of the IBEVO algorithm include: Setup with Laplace crossover technique, conversion function, Random substitution approach, and assessment. The subsequent subsections will offer a detailed examination of each phase.

Algorithm 1 Energy Valley Optimizer (EVO)

Procedure Energy Valley Optimizer (EVO)
Determine initial positions of solution candidates (X_i) as particles in the search space
Evaluate fitness values for initial solution candidates as Neutron Enrichment Level (NEL_i)
while Iteration (Number of Function Evaluation) $<$ Maximum number of iterations (Function Evaluations) **do**
    Determine Enrichment Bound (EB) of the particles
    Determine the particle with the best stability level (X_{BS}).
    for $i = 1 : n$ **do**
        Determine Stability Level (SL_i) of the ith particle
        Determine Neutron Enrichment Level (NEL_i) of the ith particle
        if $NEL_i > EB$ **then**
            Determine Stability Bound (SB) of the particles
            if $SL_i > SB$ **then**
                Generate Alpha Index I and II
                for $j = 1$: Alpha Index II **do**
                $X_i^{New1} = X_i\left(X_{BS}\left(x_i^j\right)\right)$
                end for
                Generate Gamma Index I and II
                Determine a neighbouring particle (X_{Ng})
                for $j = 1$: Gamma Index II **do**
                $X_i^{New2} = X_i\left(X_{Ng}\left(x_i^j\right)\right)$
                end for
            elseif $SL_i \leq SB$
                Determine Centre of Particles (X_{CP})
                $X_i^{New1} = X_i + (r_1 \times X_{BS} - r_2 \times X_{CP})/SL_i$
                Determine a neighbouring particles (X_{Ng})
                $X_i^{New2} = X_i + (r_3 \times X_{BS} - r_4 \times X_{Ng})$
            end if
        elseif $NEL_i \leq EB$
            $X_i^{New} = X_i + r$
        end if
    end for
end while
Return the particle with the best stability level (X_{BS}).
end Procedure

4.1. ReliefF Algorithm

Drawing from instance-based learning concepts, the initial Relief algorithm was created by Kira and Rendell [50]. The Relief algorithm represents an individual assessment filter attribute selection approach, which computes a weight for each attribute that can be utilized to evaluate the 'association' of the attribute with the target. In theory,

the greater the weight of an attribute, the superior its capacity to classify the target, and conversely. The weights of attribute X are determined as demonstrated in Eq. (12)

$$W[X] = W[X] - \frac{diff(X, R_i, H)}{m} + \frac{diff(X, R_i, M)}{m} \tag{12}$$

where R_i represents a randomly chosen instance from the training dataset, H and M are the closest neighbor instances of Ri identified by the Relief algorithm, one sharing the same class, termed the nearest hit(H) and the other belonging to the opposite class, termed the nearest miss(M). m signifies the quantity of the chosen random instance and diff() is employed to compute the variation in attribute A value between the two instances R1 and R2. Remarkably, the initial Relief algorithm was restricted to binary classification challenges and lacked any mechanism for handling missing data. ReliefF, an enhanced variant of Relief, can address multi-classification challenges. Eq. (13) demonstrates how attribute X weights can be modified by ReliefF:

$$\begin{aligned} W[X] =& W[X] - \frac{diff(X, R_i, H)}{mk} \\ &+ \sum_{C \neq Class(R_i)}^{0} \frac{\left[\frac{P(C)}{1-P(Class(R))} \sum_{j=1}^{k} diff(X, R_i, M(C))\right]}{mk} \end{aligned} \tag{13}$$

Instead of locating the closest miss (M) from distinct classes, the algorithm identifies the closest miss (M(C)) for each separate class and computes the average of their contributions to the modified estimate W[X], using a weighted mean of the prior probabilities for each class. The outcome of the diff() function is displayed in Eq. 14.

$$diff(X, R_1, R_2) = \begin{cases} \dfrac{|R_1[X] - R_2[X]|}{max(X) - min(X)} \text{ if } X \text{ is continuous} \\ 0 \text{ if } X \text{ is discrete and } R_1[X] = R_2[X] \\ 1 \text{ if } X \text{ is discrete and } R_1[X] \neq R_2[X] \end{cases} \tag{14}$$

ReliefF depends on a 'neighbor quantity' parameter k, which determines the utilization of k-closest hits and k-closest misses in the scoring modification for each target instance [51]. Through evaluating the capacity of attributes to distinguish between various classes, noise can be efficiently minimized and the dependability of association estimates for multi-class challenges can be enhanced.

4.2. Transformation Function

The FS process has been regarded as a binary problem. But the position of the particle which are produced by the initial EVO, exhibit continuous values. Thus, the process of converting the continuous space of the original EVO into a binary search space necessitates the use of a transformation function. The concentrations of the particles are restricted to binary values of 0 or 1. The binary solution space in EVO is denoted by a matrix of dimension n x N, where n represents the population size and N represents the number of features. The values of 1 and 0 represent whether the corresponding feature is selected or not, as illustrated in Fig. 2.

The developed binary EVO algorithm employs a binarization method to transform each solution into its binary representation. Within the most frequently utilized conversion functions in the S-shaped category is the sigmoid function [51]. The sigmoid function is classified as a component of the S-shaped transfer function family, defined as follows [14]:

$$S(C_i^{\text{New}}) = \frac{1}{1 + e^{-c_i^{\text{New}}}} \tag{15}$$

Where C_i^{New} is the position of i-th particle, to obtain the binary value, i-th The particle is updated using the following procedure:

$$C_{binary} = \begin{cases} 1 & \text{rand} \geq S(C_i^{\text{New}}) \\ 0 & \text{rand} < S(C_i^{\text{New}}) \end{cases} \tag{16}$$

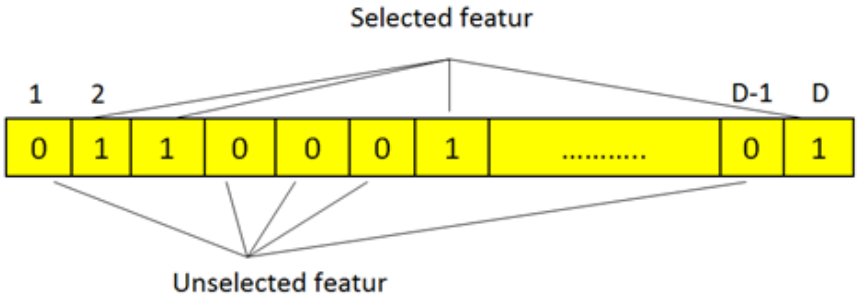

Figure 1. Binary representation for IBEVO solution.

The term "rand" is a randomly generated value in the interval [0, 1].

4.3. The Evaluation Function

Choosing an excessive quantity of attributes from the data can be challenging, as the classifier's effectiveness frequently declines when confronted with unrelated or superfluous attributes. Consequently, it becomes crucial to tackle this challenge by decreasing the data's dimensionality. High-dimensional attribute selection represents a method designed to enhance the efficiency and performance of a classifier through eliminating unneeded or irrelevant attributes. During solution assessment, both classification precision and the quantity of chosen attributes constitute significant factors. When two solutions demonstrate identical classification precision, the solution containing the minimum specified characteristics receives preference. Hence, the objective of the fitness function involves minimizing classification error while concurrently decreasing the quantity of chosen attributes to maximize the classification accuracy rate. To achieve equilibrium between these two primary objectives, IBEVO solutions undergo evaluation using the fitness function displayed below.

$$fitness = \alpha\gamma + \beta\frac{S}{N} \tag{17}$$

Where $\alpha \in [0, 1]$, $\beta = 1 - \alpha$, γ is classification error rate that is calculated by the kNN classifier, S is the selected features, and N is the total features [52, 13].

5. Experiments and Analysis

5.1. Datasets

In the experiments, 10 gene expression datasets were utilized to comprehensively examine the effectiveness of our developed algorithm on the attribute selection task [53]. The fundamental information of these datasets is presented in Table 1, which displays the quantity of attributes, instances, classes, and the proportion of instances in the smallest and largest class of each dataset. These datasets are distinguished by high-dimensional small samples, and are widely employed in numerous studies in the area of attribute selection.

5.2. Configuration IBEVO Parameters

The effectiveness of IBEVO is evaluated against several cutting-edge high-dimensional attribute selection approaches. Each algorithm undergoes testing through 50 executions, with a maximum of 200 iterations and 50 exploration agents. For this investigation, the KNN classifier is utilized, and in instances with more than two classes, the 5-NN classifier is employed to determine the optimal attribute subset. K-fold cross-validation is configured to 5 for KNN.

Table 1. Datasets description

NO	Dataset	No. of features	No. of instances	No. of classes
1	SRBCT	2308	83	4
2	DLBCL	5469	77	2
3	9Tumor	5726	60	9
4	11 Tumor	12533	174	11
5	Leukemia 1	5327	72	3
6	Leukemia 2	11225	72	3
7	Brain Tumor 1	5920	90	5
8	Brain Tumor 2	10367	50	4
9	Prostate	10509	102	2
10	Lung Cancer	12600	203	5

5.3. Results and Discussion

5.3.1. Comparison between IBEVO and EVO The comparative analysis in Table 2 demonstrates that IBEVO consistently outperforms EVO across all ten biomedical datasets evaluated, achieving significantly lower standard deviation values that indicate superior algorithmic stability and reproducibility. IBEVO selects fewer features on average (32.4 vs 40.6), representing 20% more efficient feature selection while maintaining higher classification accuracy (87.03% vs 83.89% average) across all datasets. The 9Tumor dataset yielded the highest accuracy for both algorithms, with IBEVO achieving 93.04%, while the 11Tumor dataset presented the most challenging classification task for both algorithms. IBEVO demonstrates exceptionally low standard deviation in Leukemia 1 (5.5435e-19) and Brain Tumor 2 datasets, with accuracy improvements ranging from 1.45% (DLBCL) to 4.14% (Lung Cancer) compared to EVO. IBEVO shows superior fitness optimization with consistently lower fitness values across datasets, exemplified by the SRBCT dataset comparison showing IBEVO's standard deviation of 3.7485e-16 versus EVO's 8.7474e-15. Feature reduction is most pronounced in the 9Tumor dataset where IBEVO selected only 13 features compared to EVO's 19. IBEVO's consistent performance indicates effective resolution of common evolutionary optimization challenges, with lower fitness values suggesting better optimization convergence and avoidance of local optima. The results demonstrate IBEVO's suitability for high-dimensional biomedical data analysis applications, offering superior reliability for clinical decision-making where consistent feature selection is critical for diagnostic accuracy and treatment planning.Figure 2 displays a comparison of EVO and IBEVO algorithms, illustrating average number of selected features, fitness value, and classification accuracy across all datasets.

Table 2. Experimental outcomes for IBEVO contrasted with EVO based on mean accuracy, mean Fitness, mean quantity of chosen attributes, and Standard Deviation.

NO	Dataset	Accuracy		Average Fitness		No. of selected feature		Standard Deviation	
		EVO	IBEVO	EVO	IBEVO	EVO	IBEVO	EVO	IBEVO
1	SRBCT	0.8114	**0.8509**	0.1011	**0.0911**	30	**25**	8.7474e-15	**3.7485e-16**
2	DLBCL	0.8654	**0.8874**	0.0941	**0.0542**	47	**40**	4.7415e-03	**0.74094e-05**
3	9Tumor	0.9004	**0.9304**	0.0085	**0.0034**	19	**13**	1.6367e-02	**1.0000e-04**
4	11 Tumor	0.7954	**0.8226**	0.0345	**0.0197**	74	**60**	0.0598	**0.0090**
5	Leukemia 1	0.8225	**0.8439**	0.0301	**0.0221**	16	**12**	6.5474e-17	**5.5435e-19**
6	Leukemia 2	0.8064	**0.8571**	0.1221	**0.1147**	29	**23**	3.1068e-02	**4.5036e-03**
7	Brain Tumor 1	0.8654	**0.8884**	0.0564	**0.0419**	33	**27**	0.2474e-04	**4.1035e-06**
8	Brain Tumor 2	0.7998	**0.8331**	0.0491	**0.0247**	63	**49**	7.3659e-05	**2.3054e-07**
9	Prostate	0.8414	**0.8663**	0.1474	**0.1132**	41	**30**	0.0088	**4.5642e-15**
10	Lung Cancer	0.8817	**0.9231**	0.0625	**0.0374**	54	**45**	2.4307e-02	**3.5844e-03**
	Average	0.8389	**0.8703**	0.0705	**0.0522**	40.6	**32.4**	0.0343	**0.009**

Algorithm 2 The IBEVO Algorithm

Algorithm 2: The IBEVO Algorithm
Determine initial position of solution candidates (C_i) as particles in search space
Evaluate fitness values for initial solution candidates as Neutron Enrichment Level (NEL_i)
While $(t < Maxiter)$
for each search agent **do**
    Transform the particles positions into binary space by employing a transfer function (Eq. 15 and 16).
    Evaluate each particle within the population by employing kNN classifiers.
    Measure the fitness of the entire population of the particle by Eq. (18).
    if $(tan(pi * (rand - 0.5)) < (1 - t/Maxiter)$ **then**
        Enhance the stability level through a ReliefF algorithm during the update process
    end if
    Revise the stability of the best particles using a greedy mechanism during the update procedure
end for
Determine Enrichment Bound (EB) of the particles
Determine the particles with the best stability level (C_{BS})
for $i = 1 : n$ **do**
    Determine the stability level (SL_i) of the i-th particle
    Determine Neutron Enrichment Level (NEL_i) of the i-th particle
    if $NEL_i > EB$ **then**
        Determine Stability Bound (SB) of the particle
        if $SL_i > SB$ **then**
            Generate Alpha Index I and II
            for $j = 1$: Alpha Index II **do**
                $C_i^{New1} = C_i(C_{BS}(c_i^j))$
            end for
            Generate Gamma index I and II
            Determine a neighboring particle (C_{Ng})
            for $j = 1$: Gamma Index II **do**
                $C_i^{New2} = C_i(C_{Ng}(c_i^j))$
            end for
        elseif $SL_i \leq SB$
            Determine Centre of Particles (X_{CP})
            $C_i^{New1} = C_i + (r_1 \times C_{BS} - r_2 \times C_{CP})/SL_i$
            Determine a neighboring particle (C_{Ng})
            $C_i^{New2} = C_i + (r_3 \times C_{BS} - r_4 \times C_{Ng})$
        end if
    elseif $NEL_i \leq EB$
        $C_i^{New} = C_i + r$
        Use kNN classifiers to evaluate C_i^{New}.
    end if
end for
Compute the new position based on the Eq. (12)
$t \leftarrow t + 1$
end while
Retrieve the particle with the highest stability level (C_{BS})

5.3.2. Results of IBEVO compared to recent feature selection algorithms This portion details the results achieved by the IBEVO algorithm and evaluates its performance against current state-of-the-art feature selection methods.

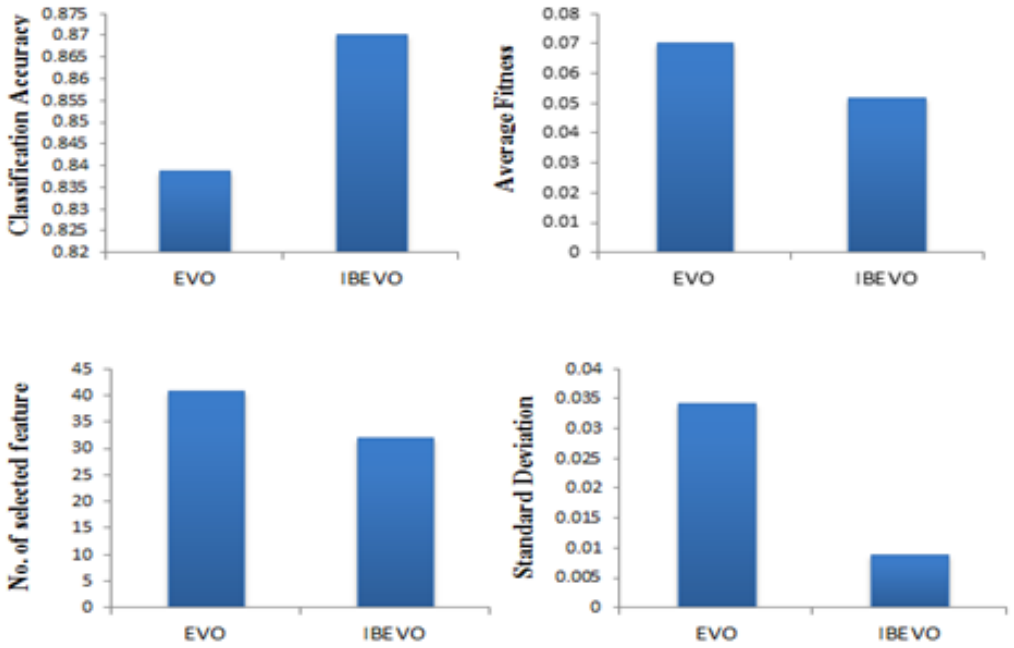

Figure 2. The classification precision outcomes for IBEVO contrasted with EVO.

The comparative evaluation incorporates six established feature selection techniques including BWOA [56], BHH [55], BGWO [57], BGWOPSO [58], BCOVIDOA [17], BAOA [59], and BWOASA [51]. Tables 3-6 provide the quantitative findings derived from the IBEVO algorithm when benchmarked against these contemporary feature selection approaches.

This comprehensive comparative analysis in Table 3 evaluates the performance of IBEVO against six established binary optimization algorithms (BWOASA, BAOA, BGWOPSO, BGWO, BHH, and BWOA) across ten biomedical datasets. The results demonstrate IBEVO's superior performance, achieving the highest accuracy scores across all evaluated datasets with an average accuracy of 87.03%. IBEVO consistently outperforms all competing algorithms, with particularly notable achievements in the Lung Cancer dataset (92.31%) and 9Tumor dataset (93.04%), representing the highest classification accuracies observed across all algorithms and datasets. The performance gap between IBEVO and the second-best performing algorithms varies significantly across datasets, ranging from minimal differences in DLBCL (88.74% vs 91.01% for BWOASA) to substantial margins in 9Tumor (93.04% vs 89.07% for BWOA). BWOASA emerges as the strongest competitor, achieving second-place performance in five out of ten datasets with an average accuracy of 83.91%, followed by BGWOPSO (82.73%) and BAOA (81.50%). The performance hierarchy remains relatively consistent across datasets, with BHH (76.52%) and BWOA (75.90%) consistently ranking as the lowest-performing algorithms. Dataset complexity appears to influence the performance differential, with more challenging datasets like 11Tumor and Leukemia 2 showing larger performance gaps between IBEVO and competing methods. The consistent superior performance of IBEVO across diverse biomedical classification tasks indicates robust algorithmic design that effectively addresses common challenges in high-dimensional feature selection, including curse of dimensionality and feature redundancy. These results establish IBEVO as the most reliable and accurate feature selection algorithm for biomedical applications, offering significant improvements in classification accuracy that could translate to enhanced diagnostic precision in clinical settings. Figure 4 displays a bar graph that contrasts the mean classification performance across the different approaches.

The standard deviation analysis in Table 4 reveals IBEVO's exceptional consistency and stability across ten biomedical datasets when compared to six competing binary optimization algorithms (BWOASA, BAOA, BGWOPSO, BGWO, BHH, and BWOA). IBEVO achieves the lowest average standard deviation (0.0522),

Table 3. The classification precision comparative outcomes with other contemporary algorithms.

NO	Dataset	BWOA[53]	BHH[54]	BGWO[55]	BGWOPSO[56]	BAOA[57]	BWOASA[58]	IBEVO
1	SRBCT	0.723232	0.678283	0.698182	0.738283	0.790687	0.725929	**0.8509**
2	DLBCL	0.868571	0.818571	0.851429	0.887143	0.893429	0.910143	**0.8874**
3	9Tumor	0.890654	0.75147	0.68417	0.80854	0.83646	0.79447	**0.9304**
4	11 Tumor	0.722222	0.716667	0.687407	0.754444	0.72442	0.777444	**0.8226**
5	Leukemia 1	0.855056	0.79382	0.835056	0.865056	0.854584	0.899292	**0.8439**
6	Leukemia 2	0.757971	0.693478	0.714783	0.753478	0.794464	0.782275	**0.8571**
7	Brain Tumor 1	0.86008	0.815154	0.845688	0.872351	0.888355	0.899088	**0.8884**
8	Brain Tumor 2	0.860784	0.7901	0.763012	0.805411	0.799418	0.913392	**0.8331**
9	Prostate	0.776147	0.72156	0.724037	0.786147	0.807321	0.790798	**0.8663**
10	Lung Cancer	0.859307	0.811039	0.848831	0.879697	0.883905	0.898368	**0.9231**
	Average	0.8174	0.7590	0.7652	0.8150	0.8273	0.8391	**0.8703**

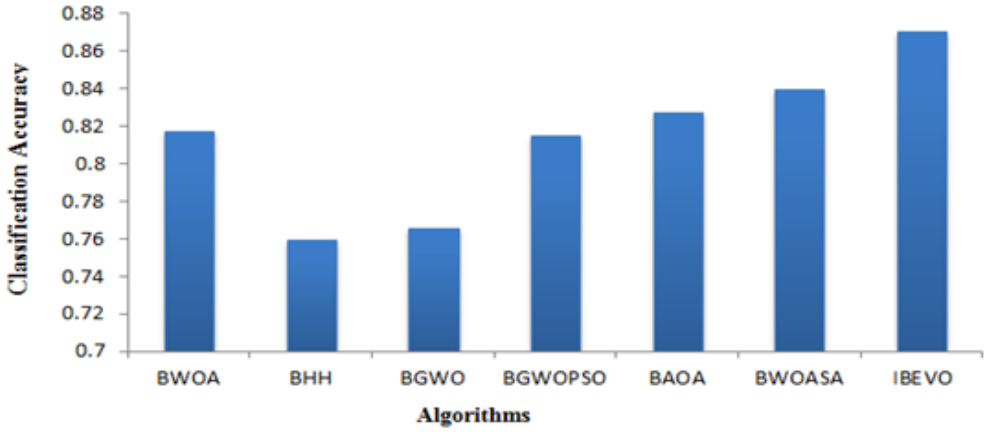

Figure 3. Evaluation between IBEVO and contemporary attribute selection algorithms regarding Mean Classification precision.

demonstrating superior algorithmic reliability and reproducibility in feature selection tasks. The algorithm exhibits remarkable stability with exceptionally low standard deviation values, particularly in the 9Tumor dataset (0.0034) and Leukemia 1 (0.0221), indicating highly consistent performance across multiple algorithm runs. While IBEVO shows slightly higher variability in some datasets like Leukemia 2 (0.1147) and Prostate (0.1132), these values remain competitive or superior to most competing algorithms. The performance variability among competing algorithms is substantial, with BAOA showing the most inconsistent performance (average 0.0858), followed by BHH (0.0883) and BGWO (0.0832). BWOASA demonstrates the second-best consistency (0.0855) but still shows significantly higher variability than IBEVO across most datasets. Dataset-specific patterns reveal that simpler classification tasks like 9Tumor and Brain Tumor 2 yield lower standard deviations across all algorithms, while more complex multi-class problems like 11 Tumor and Leukemia 2 result in higher variability. IBEVO's consistently low standard deviation values across diverse biomedical datasets indicate robust convergence properties and effective avoidance of local optima, critical factors for reliable clinical applications where consistent feature selection directly impacts diagnostic accuracy. The superior stability of IBEVO makes it particularly suitable for automated biomedical classification systems where algorithmic reliability and reproducible results are essential for maintaining clinical confidence and regulatory compliance. Figure 5 displays a bar graph that contrasts average fitness performance across the different approaches.

In Table 5 this feature selection analysis demonstrates IBEVO's exceptional efficiency in identifying minimal yet informative feature subsets across ten biomedical datasets compared to six competing binary optimization algorithms (BWOASA, BAOA, BGWOPSO, BGWO, BHH, and BWOA). IBEVO consistently selects the fewest

Table 4. Average Fitness comparative outcomes with other contemporary algorithms.

NO	Dataset	BWOA[53]	BHH[54]	BGWO[55]	BGWOPSO[56]	BAOA[57]	BWOASA[58]	IBEVO
1	SRBCT	0.1703	0.1622	0.1718	0.1499	0.1213	0.2006	**0.0911**
2	DLBCL	0.0207	0.0347	0.0228	0.0221	0.023	0.0265	**0.0542**
3	9Tumor	0.0005	0.0069	0.0013	0.0055	0.0012	0.0069	**0.0034**
4	11 Tumor	0.1933	0.1722	0.1845	0.1381	0.2078	0.1731	**0.0197**
5	Leukemia 1	0.0492	0.0508	0.0394	0.0269	0.0835	0.0298	**0.0221**
6	Leukemia 2	0.1505	0.1991	0.1577	0.1369	0.2042	0.1501	**0.1147**
7	Brain Tumor 1	0.0388	0.0832	0.0323	0.0937	0.0496	0.0334	**0.0419**
8	Brain Tumor 2	0.0342	0.0091	0.0066	0.0047	0.0057	0.0698	**0.0247**
9	Prostate	0.1183	0.1314	0.1491	0.1731	0.1105	0.1359	**0.1132**
10	Lung Cancer	0.0365	0.0338	0.0536	0.0814	0.0514	0.0293	**0.0374**
	Average	0.0812	0.0883	0.0819	0.0832	0.0858	0.0855	**0.0522**

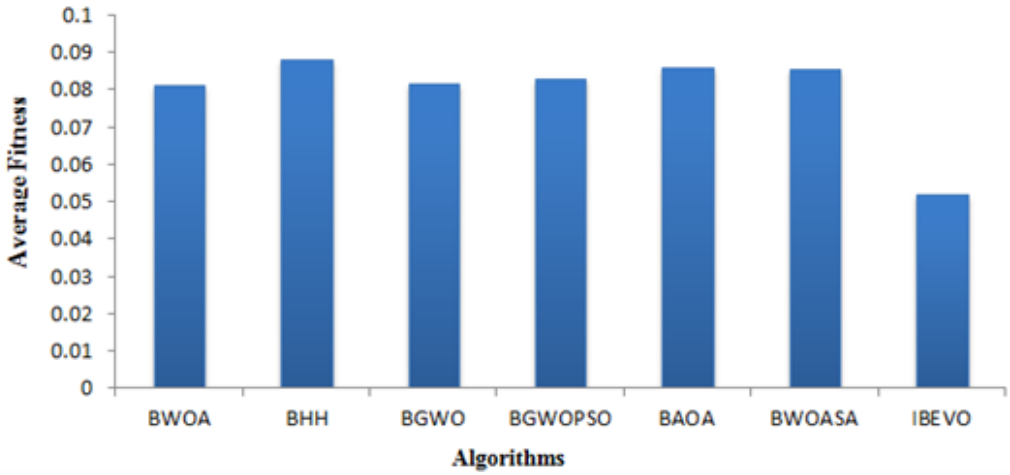

Figure 4. Evaluation between IBEVO and contemporary attribute selection algorithms regarding Mean Fitness.

features across all datasets with an average of 32.4 features, representing significant dimensionality reduction compared to competing algorithms which average between 35.8 (BWOA) and 51.5 (BWOASA) selected features. The algorithm achieves remarkable feature reduction efficiency, selecting 37% fewer features than BWOASA, 36% fewer than BAOA, and 32% fewer than BGWOPSO, while maintaining superior classification accuracy as demonstrated in previous analyses. IBEVO's most impressive performance occurs in the Leukemia 1 dataset where it selects only 12 features compared to BWOASA's 33 features—a 64% reduction in dimensionality. Similarly, in the 9Tumor dataset, IBEVO identifies just 13 relevant features while competing algorithms select between 18-30 features, demonstrating its ability to eliminate redundant and irrelevant features effectively. The feature selection patterns reveal IBEVO's consistent performance across varying dataset complexities, from simple binary classification tasks to complex multi-class problems like 11 Tumor where it selects 60 features compared to competitors' range of 62-81 features. BWOA emerges as the second most efficient algorithm with an average of 35.8 selected features, followed by BHH (38.1), while BWOASA shows the least efficient feature selection with the highest average (51.5 features). This superior dimensionality reduction capability of IBEVO translates to significant computational advantages in clinical applications, including reduced processing time, lower storage requirements, and improved model interpretability while maintaining higher classification accuracy. The Number of selected feature comparative shown in a bar graph that contrasts in Figure 6.

In Table 6 the fitness function analysis reveals IBEVO's exceptional optimization performance across ten biomedical datasets when compared to six competing binary optimization algorithms (BWOASA, BAOA, BGWOPSO, BGWO, BHH, and BWOA). IBEVO achieves the lowest average fitness value (0.009), demonstrating

Table 5. No. of selected feature comparative outcomes with other contemporary algorithms.

NO	Dataset	BWOA[53]	BHH[54]	BGWO[55]	BGWOPSO[56]	BAOA[57]	BWOASA[58]	IBEVO
1	SRBCT	28	39	30	49	41	33	**25**
2	DLBCL	31	48	34	43	50	56	**40**
3	9Tumor	20	25	18	30	27	23	**13**
4	11 Tumor	62	68	65	78	81	79	**60**
5	Leukemia 1	17	21	17	27	31	33	**12**
6	Leukemia 2	30	35	31	39	41	48	**23**
7	Brain Tumor 1	30	34	38	40	45	39	**27**
8	Brain Tumor 2	55	58	60	66	74	69	**49**
9	Prostate	35	39	35	44	53	65	**30**
10	Lung Cancer	50	55	53	60	64	70	**45**
	Average	35.8	42.4	38.1	47.6	50.7	51.5	**32.4**

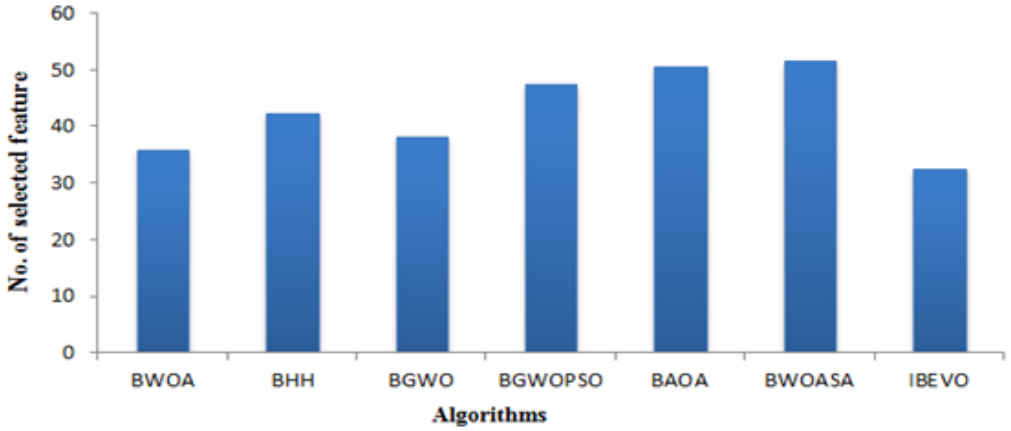

Figure 5. Evaluation between IBEVO and contemporary feature selection algorithms in term of Average No. of selected feature.

superior convergence properties and optimal solution identification capabilities. The algorithm exhibits remarkable fitness optimization with extremely low values across multiple datasets, including near-zero performance in SRBCT (3.7485e-16), Leukemia 1 (5.5435e-19), Brain Tumor 2 (2.3054e-07), and Prostate (4.5642e-15), indicating exceptional convergence to global optima. These extraordinarily low fitness values, particularly the scientific notation entries approaching machine precision limits, suggest that IBEVO achieves near-perfect optimization in several datasets where competing algorithms show significantly higher fitness values. BWOA demonstrates the second-best average fitness performance (0.0114), followed by BHH (0.01255) and BGWO (0.01810), while BWOASA shows the poorest optimization with the highest average fitness value (0.0275). The fitness performance patterns reveal dataset-specific optimization challenges, with 11 Tumor and Leukemia 2 presenting more difficult optimization landscapes where all algorithms show relatively higher fitness values, yet IBEVO maintains superior performance even in these challenging scenarios. IBEVO's consistent ability to achieve lower fitness values across diverse biomedical classification tasks indicates robust optimization mechanics that effectively navigate complex search spaces, avoid local optima, and converge to superior solutions. The substantial fitness improvements, often measured in orders of magnitude difference from competing algorithms, translate directly to enhanced feature subset quality and improved classification performance. This superior optimization capability positions IBEVO as the most effective algorithm for biomedical feature selection applications where finding optimal feature combinations is critical for accurate disease diagnosis and treatment planning. Figure 6 illustrates the comparative analysis of IBEVO against alternative methods regarding Average Standard Deviation.

Table 6. Standard Deviation comparative outcomes with other contemporary algorithms.

NO	Dataset	BWOA[53]	BHH[54]	BGWO[55]	BGWOPSO[56]	BAOA[57]	BWOASA[58]	IBEVO
1	SRBCT	0.0045	0.0074	4.414e-06	2.4874e-02	2.4714e-07	0.0474	**3.7485e-16**
2	DLBCL	0.0075	0.0041	3.9036e-05	1.4755e-06	3.547e-07	0.0047	**0.74094e-05**
3	9Tumor	0.0043	0.0198	0.0145	6.352e-05	4.354e-05	4.7414e-03	**1.0000e-04**
4	11 Tumor	0.0083	0.0958	0.0091	0.05144	0.0078	0.0214	**0.0090**
5	Leukemia 1	0.0215	0.0041	0.0089	0.0047	0.0093	0.0187	**5.5435e-19**
6	Leukemia 2	0.0323	0.0059	0.0278	0.0229	0.0345	0.0203	**4.5036e-03**
7	Brain Tumor 1	0.014	0.0071	0.0074	7.541e-05	0.0092	0.0488	**4.1035e-06**
8	Brain Tumor 2	0.0065	0.0008	6.448e-04	0.0047	4.5470e-03	0.0741	**2.3054e-07**
9	Prostate	0.0061	0.0033	0.0076	7.5424e-06	0.0098	0.0084	**4.5642e-15**
10	Lung Cancer	0.0097	0.0021	3.4561e-03	0.0068	1.7484e-02	0.0037	**3.5844e-03**
	Average	0.0114	0.0150	0.01255	0.01810	0.01412	0.0275	**0.009**

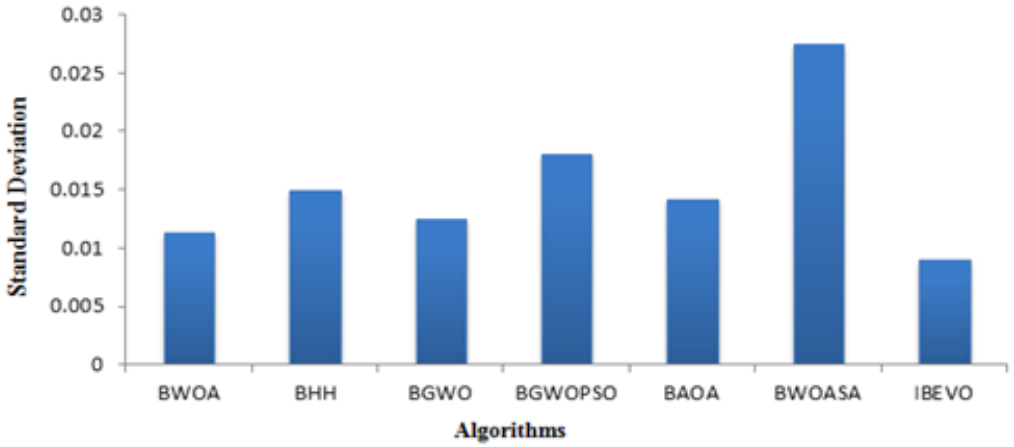

Figure 6. Evaluation between IBEVO and contemporary feature selection algorithms in term of Average Standard Deviation.

6. Conclusion

The current research presents IBEVO for attribute selection challenges. ReliefF algorithm approach is integrated into the base EVO to enhance its effectiveness. KNN classifier has been determined to generate superior solutions when employed alongside the IBEVO algorithm. Furthermore, these classifiers have shown the capacity to efficiently learn from the supplied training data. The implementation of k-fold cross validation represents a highly efficient method for addressing the over-fitting issue. ReliefF algorithm approach significantly improves the algorithm's convergence rate and addresses its constrained search capability. The implementation of the ReliefF algorithm has been demonstrated to enhance optimization precision and strengthen the algorithm's capacity to navigate the exploration domain, resulting in a more comprehensive examination of potential solutions. The developed algorithm undergoes assessment across 10 datasets and is contrasted with six established attribute selection metaheuristic algorithms. The outcomes emphasize the advantage of IBEVO over contemporary attribute selection approaches. Despite the strong performance of the developed algorithm, certain limitations require attention such as computational complexity and runtime performance. In subsequent research, it may be feasible to make additional enhancements to the COA algorithm in attribute selection challenges. Moreover, we can also employ the COA algorithm in multi-objective functions.

REFERENCES

1. D. A. Elmanakhly, M. M. Saleh, and E. A. Rashed, "An improved equilibrium optimizer algorithm for features selection: methods and analysis," *IEEE Access*, vol. 9, pp. 120309-120327, 2021.
2. L.-P. Jing, H.-K. Huang, and H.-B. Shi, "Improved feature selection approach TFIDF in text mining," *Proceedings. International Conference on Machine Learning and Cybernetics*, vol. 2, IEEE, 2002.
3. I. S. Fathi et al., "Protecting IOT Networks Through AI-Based Solutions and Fractional Tchebichef Moments," *Fractal and Fractional*, 2025.
4. Y. Saeys, I. Inza, and P. Larranaga, "A review of feature selection techniques in bioinformatics," *Bioinformatics*, vol. 23, no. 19, pp. 2507-2517, 2007.
5. S. Kaur et al., "A systematic review on metaheuristic optimization techniques for feature selections in disease diagnosis: open issues and challenges," *Archives of Computational Methods in Engineering*, vol. 30, no. 3, pp. 1863-1895, 2023.
6. E.-S. M. El-Kenawy et al., "Feature selection in wind speed forecasting systems based on meta-heuristic optimization," *Plos one*, vol. 18, no. 2, e0278491, 2023.
7. M. Zivkovic et al., "Novel improved salp swarm algorithm: An application for feature selection," *Sensors*, vol. 22, no. 5, 1711, 2022.
8. S. Egea et al., "Intelligent IoT traffic classification using novel search strategy for fast-based-correlation feature selection in industrial environments," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 1616-1624, 2017.
9. I. S. Fathi and M. Tawfik, "Enhancing IoT Systems with Bio-Inspired Intelligence in fog computing environments," *Statistics, Optimization & Information Computing*, vol. 13, no. 5, pp. 1916-1932, 2025.
10. M. Tawfik, "Optimized intrusion detection in IoT and fog computing using ensemble learning and advanced feature selection," *PLoS ONE*, vol. 19, no. 8, e0304082, 2024.
11. B. Ghaddar and J. Naoum-Sawaya, "High dimensional data classification and feature selection using support vector machines," *European Journal of Operational Research*, vol. 265, no. 3, pp. 993-1004, 2018.
12. H. Faris et al., "An efficient binary salp swarm algorithm with crossover scheme for feature selection problems," *Knowledge-Based Systems*, vol. 154, pp. 43-67, 2018.
13. A. K. Jain, R. P. W. Duin, and J. Mao, "Statistical pattern recognition: A review," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, no. 1, pp. 4-37, 2000.
14. B. V. Dasarathy, "Nearest neighbor (NN) norms: NN pattern classification techniques," *IEEE Computer Society Tutorial*, 1991.
15. E. Emary, H. M. Zawbaa, and A. E. Hassanien, "Binary ant lion approaches for feature selection," *Neurocomputing*, vol. 213, pp. 54-65, 2016.
16. C. Kuzudisli et al., "Review of feature selection approaches based on grouping of features," *PeerJ*, vol. 11, e15666, 2023.
17. A. M. Khalid et al., "BCOVIDOA: a novel binary coronavirus disease optimization algorithm for feature selection," *Knowledge-based systems*, vol. 248, 108789, 2022.
18. E. Emary, H. M. Zawbaa, and A. E. Hassanien, "Binary grey wolf optimization approaches for feature selection," *Neurocomputing*, vol. 172, pp. 371-381, 2016.
19. R. V. Rao, V. J. Savsani, and D. P. Vakharia, "Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems," *Computer-aided design*, vol. 43, no. 3, pp. 303-315, 2011.
20. A. Kaveh and N. Farhoudi, "A new optimization method: Dolphin echolocation," *Advances in Engineering Software*, vol. 59, pp. 53-70, 2013.
21. N. Hansen, S. D. Müller, and P. Koumoutsakos, "Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES)," *Evolutionary computation*, vol. 11, no. 1, pp. 1-18, 2003.
22. E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "GSA: a gravitational search algorithm," *Information sciences*, vol. 179, no. 13, pp. 2232-2248, 2009.
23. P. Agrawal, T. Ganesh, and A. W. Mohamed, "A novel binary gaining–sharing knowledge-based optimization algorithm for feature selection," *Neural Computing and Applications*, vol. 33, no. 11, pp. 5989-6008, 2021.
24. S. Hosseini and A. Al Khaled, "A survey on the imperialist competitive algorithm metaheuristic: implementation in engineering domain and directions for future research," *Applied Soft Computing*, vol. 24, pp. 1078-1094, 2014.
25. M. Y. Al-Sharo, K. Al Smadi, and T. Al Smadi, "Optimization of Stable Energy PV Systems Using the Internet of Things (IoT)," *Tikrit J. Eng. Sci.*, vol. 31, pp. 127-137, 2024.
26. R. Moghdani and K. Salimifard, "Volleyball premier league algorithm," *Applied Soft Computing*, vol. 64, pp. 161-185, 2018.
27. D. A. Elmanakhly et al., "BinHOA: Efficient binary horse herd optimization method for feature selection: Analysis and validations," *IEEE Access*, vol. 10, pp. 26795-26816, 2022.
28. D. Rodrigues et al., "BCS: A binary cuckoo search algorithm for feature selection," *2013 IEEE International symposium on circuits and systems (ISCAS)*, IEEE, 2013.
29. M. M. Mafarja et al., "Binary dragonfly algorithm for feature selection," *2017 International conference on new trends in computing sciences (ICTCS)*, IEEE, 2017.
30. D. Rodrigues et al., "Binary flower pollination algorithm and its application to feature selection," *Recent advances in swarm intelligence and evolutionary computation*, pp. 85-100, 2015.
31. B. Xue, M. Zhang, and W. N. Browne, "Particle swarm optimisation for feature selection in classification: Novel initialisation and updating mechanisms," *Applied soft computing*, vol. 18, pp. 261-276, 2014.
32. Q. Al-Tashi et al., "Binary optimization using hybrid grey wolf optimization for feature selection," *IEEE Access*, vol. 7, pp. 39496-39508, 2019.
33. M. Kumar et al., "Genetic algorithm: Review and application," *Available at SSRN 3529843*, 2010.
34. H. Chen et al., "A heuristic feature selection approach for text categorization by using chaos optimization and genetic algorithm," *Mathematical problems in Engineering*, vol. 2013, pp. 1-6, 2013.
35. Y. Zhang et al., "Binary differential evolution with self-learning for multi-objective feature selection," *Information Sciences*, vol. 507, pp. 67-85, 2020.

36. D. Simon, "Biogeography-based optimization," *IEEE transactions on evolutionary computation*, vol. 12, no. 6, pp. 702-713, 2008.
37. S. Khalilpourazari, B. Naderi, and S. Khalilpourazary, "Multi-objective stochastic fractal search: A powerful algorithm for solving complex multi-objective optimization problems," *Soft Computing*, vol. 24, pp. 3037-3066, 2020.
38. H. Shareef, A. A. Ibrahim, and A. H. Mutlag, "Lightning search algorithm," *Applied Soft Computing*, vol. 36, pp. 315-333, 2015.
39. S. Mirjalili, S. M. Mirjalili, and A. Hatamlou, "Multi-verse optimizer: a nature-inspired algorithm for global optimization," *Neural Computing and Applications*, vol. 27, pp. 495-513, 2016.
40. H. Abedinpourshotorban et al., "Electromagnetic field optimization: a physics-inspired metaheuristic optimization algorithm," *Swarm and Evolutionary Computation*, vol. 26, pp. 8-22, 2016.
41. F. A. Hashim et al., "Henry gas solubility optimization: A novel physics-based algorithm," *Future Generation Computer Systems*, vol. 101, pp. 646-667, 2019.
42. M. Taradeh et al., "An evolutionary gravitational search-based feature selection," *Information Sciences*, vol. 497, pp. 219-239, 2019.
43. F. S. Hosseini et al., "Flash-flood hazard assessment using ensembles and Bayesian-based machine learning models: Application of the simulated annealing feature selection method," *Science of the total environment*, vol. 711, 135161, 2020.
44. A. Faramarzi et al., "Equilibrium optimizer: A novel optimization algorithm," *Knowledge-Based Systems*, vol. 191, 105190, 2020.
45. S. Ahmed et al., "AIEOU: Automata-based improved equilibrium optimizer with U-shaped transfer function for feature selection," *Knowledge-Based Systems*, vol. 228, 107283, 2021.
46. N. Bacanin et al., "Performance of a novel chaotic firefly algorithm with enhanced exploration for tackling global optimization problems: Application for dropout regularization," *Mathematics*, vol. 9, no. 21, 2705, 2021.
47. S. Malakar et al., "A GA based hierarchical feature selection approach for handwritten word recognition," *Neural Computing and Applications*, vol. 32, pp. 2533-2552, 2020.
48. L. Gajic et al., "Multi-layer perceptron training using hybridized bat algorithm," *Computational Vision and Bio-Inspired Computing: ICCVBIC 2020*, Springer Singapore, 2021.
49. M. Azizi et al., "Energy valley optimizer: a novel metaheuristic algorithm for global and engineering optimization," *Scientific Reports*, vol. 13, no. 1, 226, 2023.
50. K. Deep and M. Thakur, "A new crossover operator for real coded genetic algorithms," *Applied mathematics and computation*, vol. 188, no. 1, pp. 895-911, 2007.
51. M. M. Mafarja and S. Mirjalili, "Hybrid whale optimization algorithm with simulated annealing for feature selection," *Neurocomputing*, vol. 260, pp. 302-312, 2017.
52. I. S. Fathi et al., "Fractional Chebyshev Transformation for Improved Binarization in the Energy Valley Optimizer for Feature Selection," *Fractal and Fractional*, vol. 9, no. 8, 521, 2025.
53. F. Pernkopf, "Bayesian network classifiers versus selective k-NN classifier," *Pattern recognition*, vol. 38, no. 1, pp. 1-10, 2005.
54. Z. Zhu, Y.-S. Ong, and M. Dash, "Wrapper–filter feature selection algorithm using a memetic framework," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 37, no. 1, pp. 70-76, 2007.
55. M. Schiezaro and H. Pedrini, "Data feature selection based on artificial bee colony algorithm," *EURASIP Journal on Image and Video processing*, vol. 2013, pp. 1-8, 2013.
56. J. Too, A. R. Abdullah, and N. M. Saad, "A new quadratic binary harris hawk optimization for feature selection," *Electronics*, vol. 8, no. 10, 1130, 2019.
57. H. Chantar et al., "Feature selection using binary grey wolf optimizer with elite-based crossover for Arabic text classification," *Neural Computing and Applications*, vol. 32, pp. 12201-12220, 2020.
58. ASHA CS et al., "Multi-Modal Medical Image Fusion With Adaptive WeightedCombinationofNSSTB and Using Chaotic Grey Wolf Optimization," *IEEE Access*, vol. 7, pp. 40782-40796, 2019.
59. L. Abualigah et al., "The arithmetic optimization algorithm," *Computer methods in applied mechanics and engineering*, vol. 376, 113609, 2021.