Journal of Algorithms and Computations

journal homepage: http://jac.ut.ac.ir

Relevance-Based Sailfish Optimizer for Robust and Compact Breast Cancer Diagnosis Models

M. A. Shiri *[1] and N. Mansouri†[2]

[1] Department of Computer Science, Shahid Bahonar University of Kerman, Kerman, Iran

[2] Department of Computer Science, Shahid Bahonar University of Kerman, Kerman, Iran

ABSTRACT

Machine learning models for breast cancer diagnosis are often hindered by the high dimensionality of clinical datasets, where many features are redundant or irrelevant. To address this challenge, this paper proposes a novel hybrid feature selection method, the Relevance-Based Sailfish Optimizer Feature Selection (RBSOFS), designed to identify a minimal yet highly informative subset of features. The RBSOFS approach was implemented and evaluated on the Breast Cancer Wisconsin dataset, with the selected features being fed into five established classifiers: Naive Bayes (NB), Random Forest (RF), Support Vector Machine (SVM), Decision Tree (DT), and Logistic Regression (LR). Significantly, this algorithm was obtained using a subset of only 6-7 features, a drastic reduction that leads to simpler and more computationally efficient models compared to competing methods. The findings indicate that RBSOFS is a robust and effective framework for enhancing breast cancer diagnosis.

Keywords: Healthcare, Relevance computation, Sailfish Optimization, Accuracy, Filter method

ARTICLE INFO

Article history:
Research paper
Received 27 June 2025
Accepted 16 July 2025
Available Online 16 July 2025

AMS subject classification: 68T05, 90C59.

† Corresponding author: Najme Mansouri, Email: *n..mansouri@uk.ac.ir*.

Email: *mohammadansari7685@gmail.com*

Journal of Algorithms and Computation 57 issue 1, August 2025, PP. 157—184

1. Introduction

The vast amount of data generated in the healthcare domain presents a significant opportunity for creating intelligent systems capable of diagnosing and treating various diseases [1]. In particular, machine learning techniques have gained substantial interest for developing diagnostic systems for the early detection of life-threatening diseases such as breast cancer. Early and accurate diagnosis is paramount for improving patient outcomes and reducing mortality rates. However, a critical challenge in building such systems is the high-dimensional nature of medical datasets, which often contain noisy and redundant features. The inclusion of these features can lead to issues like overfitting and ultimately degrade the performance of predictive models [2, 3].

To address this curse of dimensionality, feature selection (FS) serves as a crucial preprocessing step [4]. The primary goal of FS is to identify and select a subset of the most informative features, thereby improving learning efficiency, reducing computational costs, and enhancing model generalization [5]. An effective FS method seeks to find a feature subset with two key properties: maximum relevance to the target class (i.e., the diagnosis) and minimum redundancy among the selected features themselves [6]. While numerous FS algorithms exist, traditional or exhaustive methods are often computationally infeasible for complex, high-dimensional data. Consequently, metaheuristic optimization algorithms have been widely adopted as a powerful and efficient approach for tackling this complex search problem [1].

This paper proposes a novel method, Relevance-Based Sailfish Optimizer Feature Selection (RBSOFS), to improve the accuracy of breast cancer diagnosis by identifying an optimal feature subset from the Breast Cancer Wisconsin (Diagnostic) dataset. The key contributions and benefits of this work are:

- *Demonstrating Superior Diagnostic Accuracy:* We show through extensive experiments that the proposed RBSOFS method achieves a higher and more robust classification accuracy compared to both a baseline model (using all features) and eight other state-of-the-art metaheuristic algorithms across five different classifiers.

- *Achieving a Better Trade-off between Accuracy and Complexity:* The primary benefit of our method is its ability to significantly reduce model complexity without sacrificing performance. RBSOFS identifies a minimal subset of highly relevant features (e.g., 6-7 features), outperforming methods that rely on more than double that number. This results in diagnostic models that are simpler, computationally faster, and more easily interpretable for clinical use.

- *Introducing a Novel Hybrid FS Framework:* We present a new two-stage hybrid feature selection model. This model first uses a relevance-based filter to efficiently discard irrelevant features and then leverages the exploration and exploitation capabilities of the Sailfish Optimizer to pinpoint the most effective feature combination from the remaining candidates.

The remainder of this paper is organized as follows. Section 2 provides the necessary theoretical background on feature selection and metaheuristic algorithms. Section 3 reviews the relevant literature on state-of-the-art methods and their applications in healthcare. Section 4 details the proposed RBSOFS methodology. Section 5 presents the experimental results, including the comparative analysis and discussion of the findings. Finally, Section 6 concludes the paper, summarizing the key achievements, limitations, and directions for future work.

2. Background

This section covers the fundamental concepts of feature selection, metaheuristic algorithms, and the learning algorithms used in this study.

2.1. Feature Selection (FS)

Feature selection is a crucial data preprocessing technique used to select an optimal subset of features from the original dataset without altering them. The general process, as illustrated in Figure 1, typically involves four main stages: generating a candidate feature subset [7], evaluating the subset's quality using a specific strategy [8], validating the subset against certain criteria, and applying a stopping condition to terminate the process.

FS algorithms are broadly classified into three main categories based on how they interact with the machine learning model [9-11]:

- Filter methods rank features based on their intrinsic statistical properties (e.g., correlation) without involving any learning algorithm.
- Wrapper methods use the predictive performance of a specific learning algorithm to evaluate the quality of a feature subset. While often achieving high accuracy, they are computationally expensive because the learning algorithm must be trained repeatedly.
- Embedded methods integrate the feature selection process directly into the construction of the learning model, offering a compromise between the filter and wrapper approaches [12, 13].

The methodology in this paper incorporates a hybrid approach, leveraging the strengths of these different strategies. Table 1 provides a summary of the advantages and disadvantages of each approach.

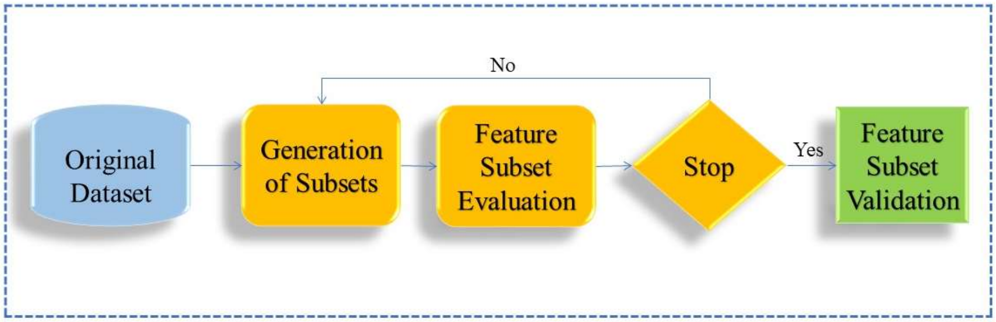

Figure 1. Feature selection process.

Table 1: Advantages and disadvantages of feature selection approaches.

Disadvantages	Advantages	Method
Lower Accuracy and Higher Error Probability	Class-Independence, Cost-Effective, Good Generalizability	Filter
Higher Computational Cost, Reduced Processing Speed, Risk of Overfitting	Higher Accuracy, Classifier Interaction, Ability to Identify Feature Dependencies	Wrapper
Feature Selection Is Integrated with the Classification Process	High Accuracy, Suitable Generalizability, Low Computational Cost	Embedded

2.2. Metaheuristic algorithms

Finding the optimal feature subset is an NP-hard problem, making exhaustive search methods computationally impractical for most real-world datasets [14, 15]. For this reason, metaheuristic algorithms have become a popular and effective strategy [16, 17]. These are high-level optimization frameworks, often inspired by natural or biological processes, designed to find a near-optimal solution in a reasonable time frame [18]. A key strength of metaheuristics is their ability to balance exploration (searching broadly across the solution space) and exploitation (focusing the search on promising areas), which is essential for navigating the complex FS landscape [19]. Their ability to handle non-analytical, black-box objective functions makes them a powerful choice for wrapper-based feature selection [20, 21].

2.3. Learning algorithms

To assess the quality of the feature subsets selected by different methods, this study uses five well-established classification algorithms:

- *Naive Bayes (NB):* A simple probabilistic classifier based on Bayes' theorem with strong independence assumptions [22, 23].
- *Support Vector Machine (SVM):* A powerful classifier that finds an optimal hyperplane to separate data points into different classes [24, 25].
- *Logistic Regression (LR):* A statistical model used to predict a binary outcome, such as the presence or absence of a disease [26, 27].
- *Decision Tree (DT):* A non-parametric model that uses a tree-like structure of decisions and their possible consequences [28, 29, 30].

- *Random Forest (RF):* An ensemble learning method that constructs a multitude of decision trees during training to improve predictive accuracy and control overfitting [31, 32].

3. Related Works

This section reviews the relevant literature from two key perspectives. First, it examines recent advancements in the development of metaheuristic algorithms for the feature selection problem. Second, it explores the application of these techniques within the healthcare domain, with a particular focus on cancer diagnosis, to contextualize the contributions of this study.

3.1. Feature Selection with Metaheuristic Algorithms

A robust learning model is built by identifying and eliminating irrelevant and redundant information. A feature selection method reduces computational and processing costs while improving model performance. Recent literature on metaheuristic-based feature selection demonstrates several key trends aimed at enhancing search efficiency and classification accuracy. One prominent approach involves improving or hybridizing well-established algorithms [33, 34]. For instance, researchers have enhanced the Whale Optimization Algorithm (WOA) by incorporating operators from Darwinian evolution to avoid local optima [35] and adapted the Firefly Algorithm (FA) for greater efficiency in practical applications [36]. Similarly, hybrid models have been proposed to leverage the strengths of multiple optimizers, such as combining the Grey Wolf Optimizer (GWO) with the Harris Hawks Optimizer (HHO) [37], or creating a hybrid method based on the Dynamic Butterfly Optimization Algorithm (DBOA) to specifically improve the balance between exploration and exploitation [38].

Another line of research focuses on developing novel algorithmic variants or applying them to specific FS contexts. This includes designing problem-specific genetic algorithms like PS-NSGA to handle multiple objectives effectively [39], creating multi-population versions of Particle Swarm Optimization (PSO) to improve solution diversity and initialization [40], and utilizing WOA within a multi-objective framework that simultaneously optimizes both filter and wrapper criteria [41].

A summary of these representative works, highlighting their core components and reported disadvantages, is presented in Table 2.

Table 2: Related works on feature selection with metaheuristic algorithms.

Disadvantage(s)	**Dataset used**	**Learning algorithm**	**Objective function(s)**	**Compared methods**	**Algorithm(s)**	**Ref.**
Limited Performance	4	SVM	Accuracy, Number of Selected Features	WOA	WOA	Tubishat et al. (2019) [35]

High Convergence Speed Compared to FA	22	KNN	Accuracy, Number of Selected Features	FA	FA	Bacanin et al. (2023) [36]
Irrational Numbers in The Best Fitness Metric	18	KNN	Accuracy, Number of Selected Features, and Computational Time	GWO HHO	HBGWOHHO	R. Al-Wajih et al. (2021) [37]
Nondeterministic, Lack of Generalization, Dependence on Dataset Characteristics	20	SVM NB DT	Accuracy, Number of Selected Features	DBOA	IFS-DBOIM	Tiwari and Chaturvedi (2022) [38]
High complexity	15	KNN NB	Accuracy, Number of Selected Features	GA	PS-NSGA	(Y. Zhou et al., 2021) [39]
Less Efficient for Low-Dimensional Datasets	29	KNN	Accuracy, Number of Selected Features	PSO	MPPSO	Kılıç et al. (2021) [40]
Higher Running Time Due to Crowding Distance Calculation, Increased Runtime Caused by the Applied Filter Function	12	KNN	Accuracy, Number of Selected Features	WOA	GPAWOA	Got et al. (2021) [41]

3.2. Feature Selection in Healthcare

The high dimensionality of medical data means that combining all available features with powerful classifiers can lead to overfitting and poor generalization. In a clinical context, using the wrong models can be disastrous, leading to incorrect diagnoses. Therefore, feature selection is a critical step in building reliable healthcare diagnostic systems.

In the specific context of breast cancer (BC), various studies have utilized feature selection to improve diagnostic accuracy. These efforts range from applying general Knowledge Data Discovery (KDD) frameworks to identify key features for early BC identification [42], to proposing novel wrapper-based models using algorithms like the Grasshopper Optimization Algorithm (GOA) to reduce the number of features while maintaining high accuracy [43]. Other works have focused on developing Computer-Aided Diagnosis (CAD) systems for analyzing mammogram images, employing unique algorithms such as the intelligent water drop (IWD) to extract the most critical features from image data [44].

The application of these techniques extends to other areas of oncology as well. For example, feature selection has been crucial in improving the classification of dermoscopic images for early melanoma detection, where binary variants of the Harris Hawk Optimization (HHO) algorithm were used to select significant visual features [45]. Furthermore, in the domain of cancer classification using high-dimensional microarray data, the combination of Particle Swarm Optimization (PSO) with ensemble learning methods has proven to be an effective strategy [46].

A review of the literature reveals that while significant progress has been made, several challenges remain. Some of the proposed methods suffer from high computational complexity [37] or a lack of generalization, making their performance dependent on the specific characteristics of the dataset [40]. A more critical gap, however, lies in the trade-off between classification accuracy and model simplicity. Many existing algorithms select a relatively large number of features to achieve high accuracy, which can result in models that are complex, slow, and difficult to interpret in a clinical setting. Therefore, a clear need exists for an efficient feature selection framework that can identify a minimal yet powerful subset of features to build a highly accurate and parsimonious diagnostic model. This study addresses this gap by proposing the RBSOFS method, specifically designed to maximize predictive accuracy while using the smallest possible number of features.

4. Proposed Method

This section details the components of our proposed methodology, including the relevance computation, the Sailfish Optimizer, the dataset used, and the overall framework.

4.1. Relevance computation

A feature can be relevant to a class label (*C*) either individually or in combination with other variables. Feature relevance is typically described as strongly relevant, weakly relevant, or irrelevant. A strongly relevant feature contains information that cannot be replaced by any other feature without a loss of predictive power. Weakly relevant features provide useful information but can be substituted by other features. Irrelevant features provide no useful information, and their removal can improve data quality without information loss [47-49]. The formal conditions for these relevance levels for a given feature *fi* are summarized in Table 3.

4.2. Sailfish Optimizer (SO)

Group hunting by arthropods, fish, birds, and mammals is a good example of social behavior. Group hunting requires less effort from predators than hunting alone. It is simplest to have predators attack prey without any coordination, whereas it is most complex to herd and catch prey using specific roles and strategies.

Table 3: Relevance levels for feature f_i.

Mutual information approach	Probabilistic approach	Condition	Relevance level
$I(f_i;C \mid f_i) \succ 0$	$p(C \mid f_i, \neg f_i) \neq p(C \mid \neg f_i)$	$\nexists$	**Strongly relevant**

$I(f_i;C\mid f_i)\succ 0$ $\wedge$ $I(f_i;C\mid S)\succ 0$	$p(C\mid f_i,\neg f_i)\neq p(C\mid \neg f_i)$ $\wedge$ $p(C\mid f_i,S)\neq p(C\mid S)$	$\exists S\subset \neg f_i$	**Weakly relevant**
$I(f_i;C\mid S)\succ 0$	$p(C\mid f_i,S)\neq p(C\mid S)$	$\exists S\subset \neg f_i$	**Irrelevant**

It is based on an attack-alternation strategy for a group of hunting sailfish that hunt a school of sardines [50]. It saves energy for the hunters to use this hunting strategy. It examines two populations: sailfish and sardines. The variables of the problem are sailfish positions in the search space, which are the candidate solutions. Sailfish and sardine search agents are mostly randomized by the algorithm. While sailfishes are scattered in the search space, sardines help find the best solution based on their positions. P^{i}_{SoBest} gives the position of the elite sailfish after the *i*-th iteration. $P^{i}_{SdInjured}$ determines the position of those "injured" sardines at iteration *i*. Sailfish and sardines are updated every iteration. According to Eq. (1), the new position P^{i+1}_{So} of a sailfish is determined at *(i+1)*-th iteration by using "elite" sailfish and "injured" sardines.

$$P_{So}^{i+1}=P_{SoBest}^{i+1}-\mu_i\times(rnd\times\frac{P_{SoBest}^{i}+P_{SdInjured}^{i}}{2}-P_{So}^{i}) \tag{1}$$

The previous position of the sailfish is denoted by P^{i}_{So}, *rnd* is a random number between 0 and 1, and μ_i indicates a coefficient calculated using Eq. (2).

$$\mu_i=2\times rnd\times PrD-PrD \tag{2}$$

PrD is the prey density, which indicates how many preys are encountered each time. As the number of prey decreases, *PrD* decreases with each iteration.

$$PrD=1-\frac{Num_{So}}{Num_{So}+Num_{Sd}} \tag{3}$$

Num_{So} is the number of sailfish and Num_{Sd} is the number of sardines.

$$Num_{So}=Num_{Sd}\times Prcnt \tag{4}$$

Where *Prcnt* indicates how many sardines make up the initial sailfish population. The number of sailfishes is always higher than the number of sardines at the beginning of any season. Sardine positions are updated in each iteration according to Eq. (5).

$$P_{Sd}^{i+1}=rnd(0,1)\times(P_{SoBest}^{i}-P_{Sd}^{i}+ATK) \tag{5}$$

$$ATK=A\times(1-(2\times itr\times k)) \tag{6}$$

P^{i}_{Sd} and P^{i+1}_{Sd} represent the previous and updated position of the sardine, and *ATK* represents the sailfish's attack power at iteration *itr*. Based on *ATK*, sardines update their positions and move a certain amount. Search agents are more likely to converge if the *ATK* is reduced. The number of sardines (γ) and the number of variables (δ) that update their position are calculated using *ATK* as follows:

$$\gamma = Num_{Sd} \times ATK \tag{7}$$

$$\delta = v \times ATK \tag{8}$$

Num_{Sd} is the number of sardines and *v* is the number of variables. Sardines are eliminated from their population if one becomes fitter than any sailfish, causing the sailfish to update its position in relation to that sardine. A random selection of sailfish and sardines ensures exploration of the search space. Sardines are able to escape sailfish after every iteration because sailfish reduce their attack power after each iteration. The *ATK* parameter balances exploration and exploitation.

4.3. Breast Cancer Wisconsin (Diagnostic) Dataset

This study utilizes the Breast Cancer Wisconsin (Diagnostic) dataset from the UCI Machine Learning Repository [51]. The dataset consists of 569 instances, each belonging to one of two diagnostic classes: malignant or benign. It includes 30 real-valued features that are computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. These features describe characteristics of the cell nuclei, such as radius, texture, perimeter, and area.

4.4. Proposed Relevance-based Sailfish Feature Selection (RBSOFS)

The proposed methodology in this study follows the Knowledge Discovery process, as illustrated in the flowchart in Figure 2. The process is organized into several distinct phases:

1. *Preprocessing and Data Splitting:* The initial phase involves preparing the raw data. This includes handling outliers and normalizing the features. Following preprocessing, the dataset is split into a training set (60% of the data) and a testing set (40%) to ensure a robust evaluation and prevent model overfitting.

2. *Hybrid Feature Selection (RBSOFS):* Our proposed method, RBSOFS, employs a two-stage hybrid approach.

   - *Filter Stage:* First, a relevance-based filter is applied to the training data. This step calculates the correlation between each feature and the target class, discarding any irrelevant or weakly correlated features to reduce the search space.

   - *Wrapper Stage:* The core of the method involves applying the Sailfish Optimizer (SO) to the filtered feature set. Since feature selection is a binary problem, a binary version of the SO algorithm is used. This requires a sigmoid transfer function, shown in Eq. (9) and Figure 3, to map the continuous position values of an agent to a probability. Then, as detailed in Eq. (10), this probability is used to update the agent's position to a discrete binary value (0 for non-selection, 1 for selection). The fitness of each feature subset generated by RBSOFS is evaluated using five renowned classifiers (NB, RF, SVM, LR, and DT) with a 10-fold cross-validation scheme.

3. *Evaluation and Comparison:* The final phase involves a comprehensive performance evaluation. The model equipped with RBSOFS is compared against two benchmarks: (1) a baseline model using all 30 features, and (2) the same five classifiers combined with eight other state-of-the-art metaheuristic algorithms. The performance is measured using multiple evaluation metrics, including Accuracy, Recall, Precision, and F1-score, based on the information from the confusion matrix [52].

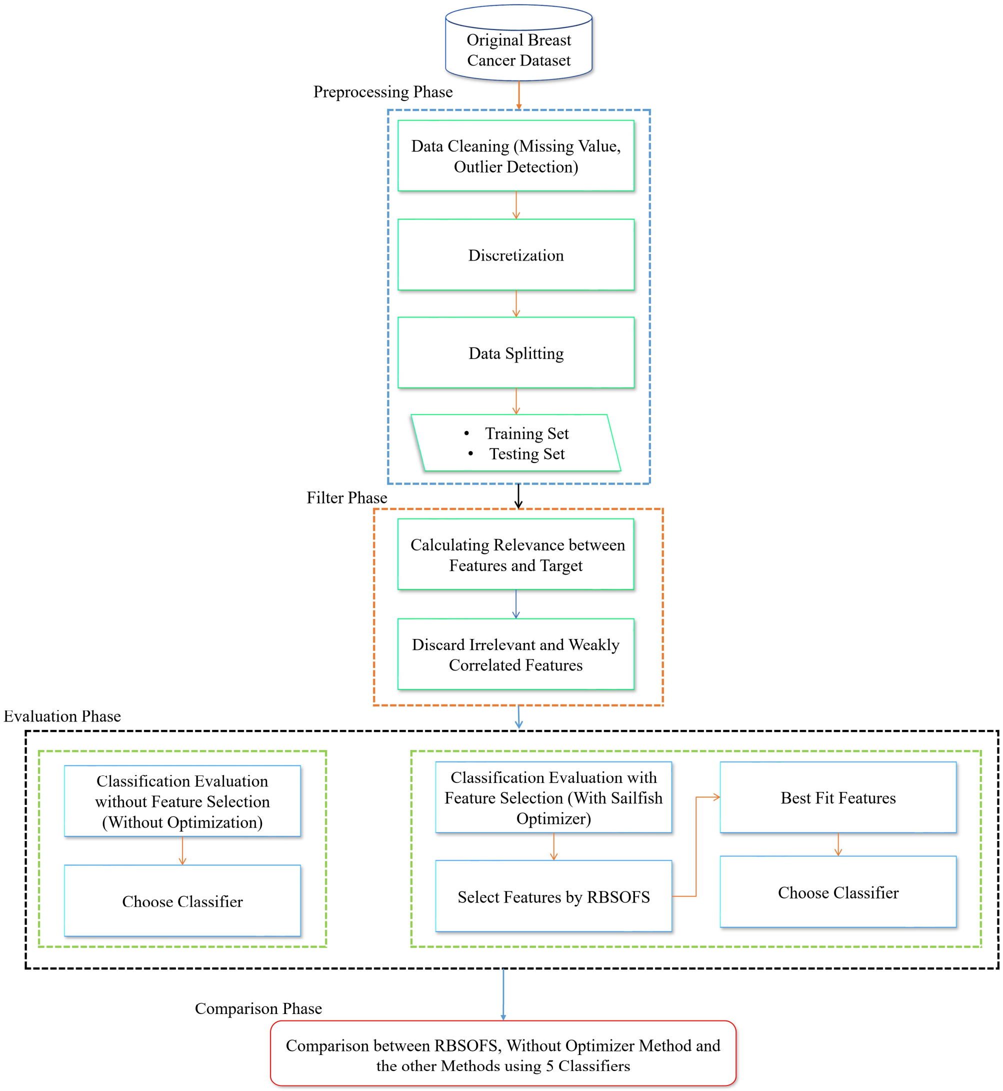
Original Breast Cancer Dataset
Preprocessing Phase
Data Cleaning (Missing Value, Outlier Detection)
Discretization
Data Splitting
• Training Set
• Testing Set
Filter Phase
Calculating Relevance between Features and Target
Discard Irrelevant and Weakly Correlated Features
Evaluation Phase
Classification Evaluation without Feature Selection (Without Optimization)
Choose Classifier
Classification Evaluation with Feature Selection (With Sailfish Optimizer)
Select Features by RBSOFS
Best Fit Features
Choose Classifier
Comparison Phase
Comparison between RBSOFS, Without Optimizer Method and the other Methods using 5 Classifiers

Figure 2. Flowchart of the proposed method.

$$T(x) = \frac{1}{1 + e^{-x}} \tag{9}$$

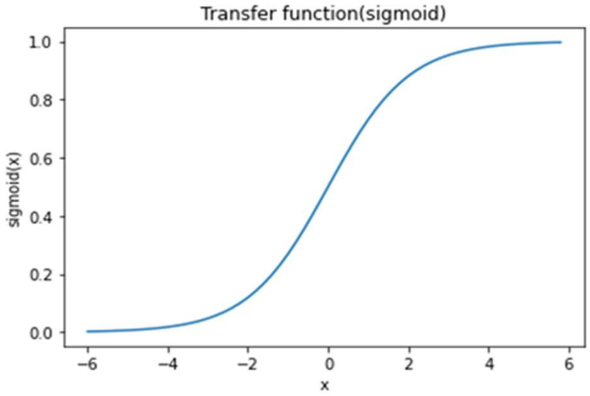

Figure 3. Transfer function for converting continuous search space to binary.

$$X^d(t) = \begin{cases} 1 & if\ \ rnd < T(X^d(t)) \\ 0 & if\ \ rnd \geq T(X^d(t)) \end{cases} \tag{10}$$

5. Results and Discussion on UCI BCWD

This section presents and discusses the experimental results obtained from applying the proposed method to the UCI Breast Cancer Wisconsin (Diagnostic) dataset.

5.1. Parameter tuning

The performance of any metaheuristic algorithm is sensitive to its parameter settings. For the proposed RBSOFS algorithm, the key parameters are the population size (controlled by *Prcnt*) and the maximum number of iterations. The *Prcnt* parameter, as defined in Eq. (4), dictates the balance between the algorithm's exploration and exploitation capabilities.

Our experiments indicated a clear trade-off: decreasing the *Prcnt* value leads to a larger search population (*Num_sd*), which enhances exploration and generally increases classification accuracy. However, this also significantly increases the computational time.

To achieve an optimal balance between high accuracy and computational efficiency, a value of *Prcnt* = 0.1 was set for all subsequent experiments. The control parameters for the other compared algorithms and the classifiers, which were set based on common values from the literature, are detailed in Table 4.

Table 4: Control parameters.

Parameter(s)	**Algorithm/ Classifier**
Number of particles = 5	GWO
Max-Iter = 30	
Number of particles = 5	DE
Max-Iter = 30	
Crossover Rate = 0.9	
Constant Factor = 0.5	
Number of particles = 5	FA
Max-Iter = 30	
Alpha, Beta, Gamma = 1	
Theta = 0.97	
Number of particles = 5	JA
Max-Iter = 30	
Number of particles = 5	SCA
Max-Iter = 30	
Alpha = 2	
Switch Probability = 0.8	FPA
Number of particles = 5	
Max-Iter = 30	
Number of particles = 5	SSA
Max-Iter = 30	
Pop-Size = 20	PSO
Max-Iter = 30	
C1, C2 = 2	
Random_state = 100	DT Classifier
Max_depth=3	
C=2	SVM Classifier
Kernel = 'linear'	
n_estimators = 300	RF Classifier

5.2. Evaluation metrics

To evaluate the performance of the classification models, several standard metrics derived from the confusion matrix are used. The core components of the confusion matrix are True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN).

- *Accuracy:* This represents the proportion of all instances that were correctly classified. While a good general indicator, it can be misleading in datasets with imbalanced classes.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{11}$$

- *Recall (Sensitivity):* This measures the proportion of actual positive instances (patients with cancer) that were correctly identified by the model. High recall is crucial in medical diagnosis to minimize the number of missed cases (False Negatives).

$$Recall = \frac{TP}{TP + FN} \tag{12}$$

- *Precision:* This measures the proportion of instances predicted as positive that were actually positive. It answers the question: "Of all the patients the model flagged as having cancer, what percentage actually did?".

$$Precision = \frac{TP}{TP + FP} \tag{13}$$

- *F1-Score:* This is the harmonic mean of Precision and Recall, providing a single metric that balances the two. It is particularly useful when dealing with imbalanced classes, as is common in medical datasets.

$$F1 - score = 2 \times \frac{(Precision \times Recall)}{Precision + Recall} \tag{14}$$

5.3. Preprocessing

As outlined in the methodology, the dataset underwent a preprocessing phase to ensure data quality before model training. This process involved two primary steps: data normalization and outlier removal. First, all 30 features were normalized to scale them to a common range, preventing features with larger values from disproportionately influencing the model. Figure 4 displays the boxplot of the features before this step, illustrating their wide-ranging scales. In contrast, Figure 5 shows the features after normalization, where they exhibit a more uniform distribution. Next, an outlier detection algorithm was applied to the normalized data to identify and remove anomalous instances that could negatively impact the training process. Figures 6 and 7 visualize this process for two sample features, where an outlier score is calculated for each data point. Data points whose scores exceeded a predefined threshold were flagged as outliers and subsequently removed from the dataset, as illustrated in Figure 8. This resulted in a cleaner, more robust dataset for the subsequent classification phases.

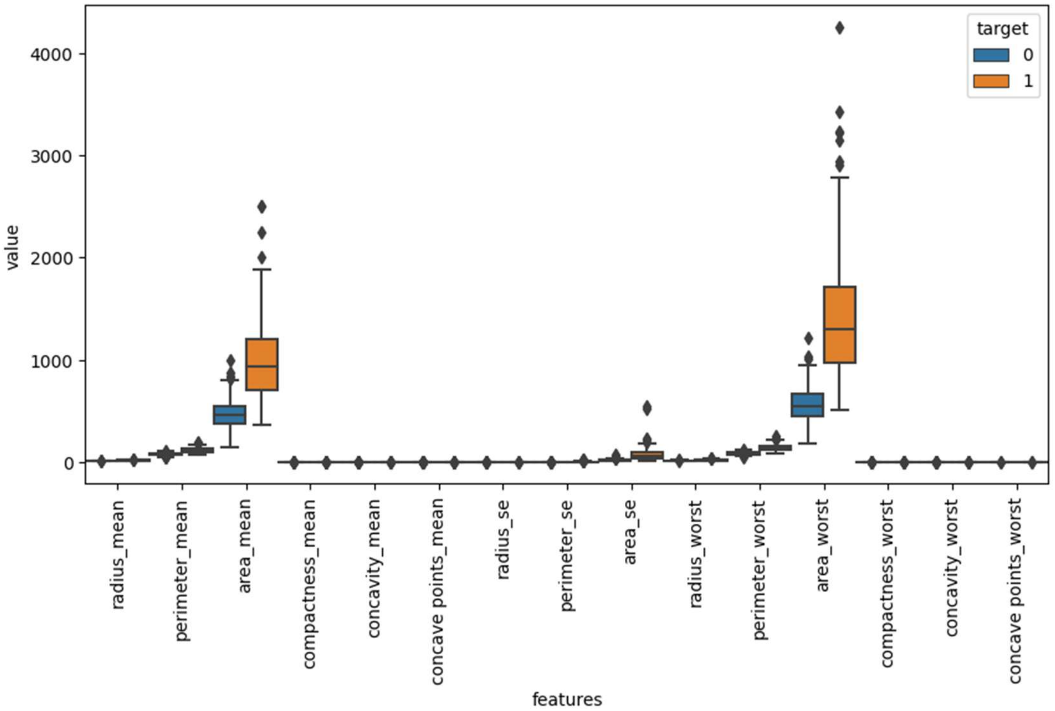

Figure 4. The boxplot of features before normalization.

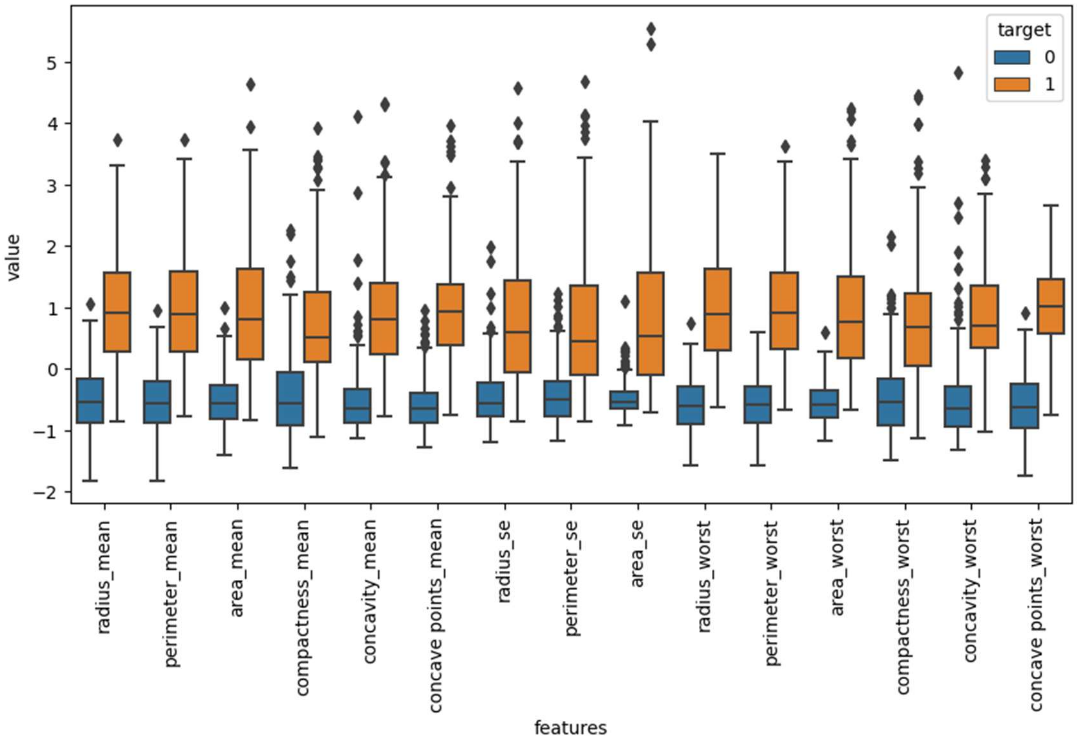

Figure 5. Boxplot of features after normalization.

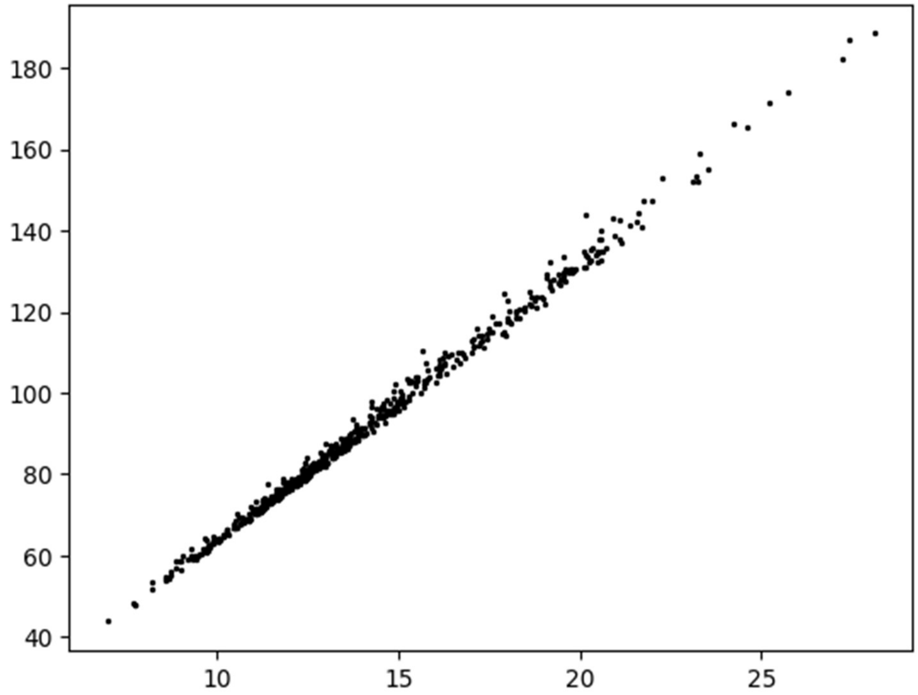

Figure 6. Scatterplot of the BCWD.

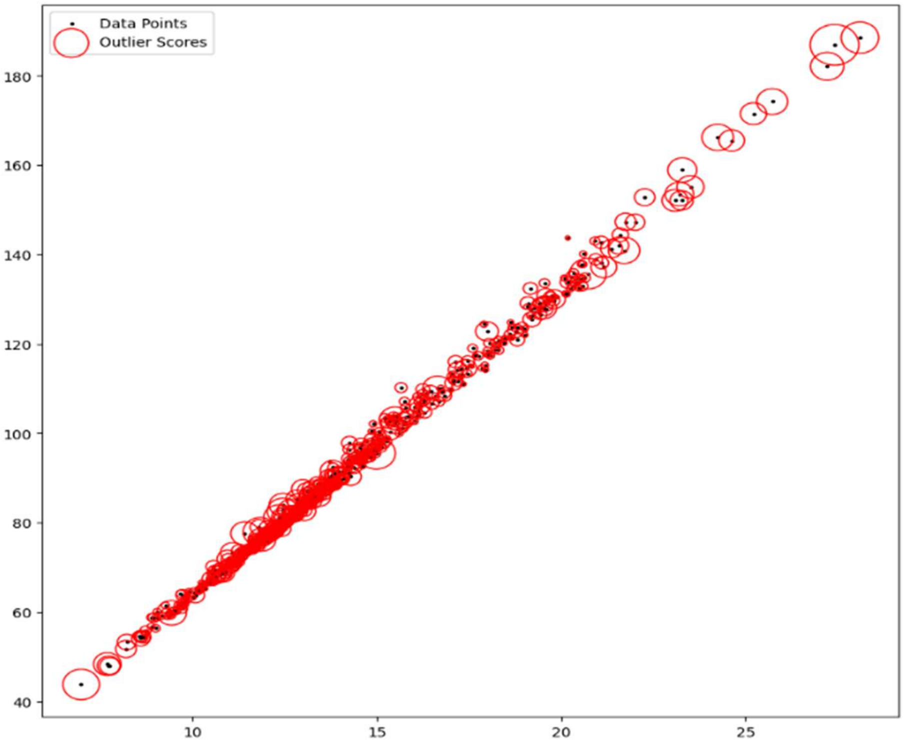

Figure 7. Outlier scores of the data points.

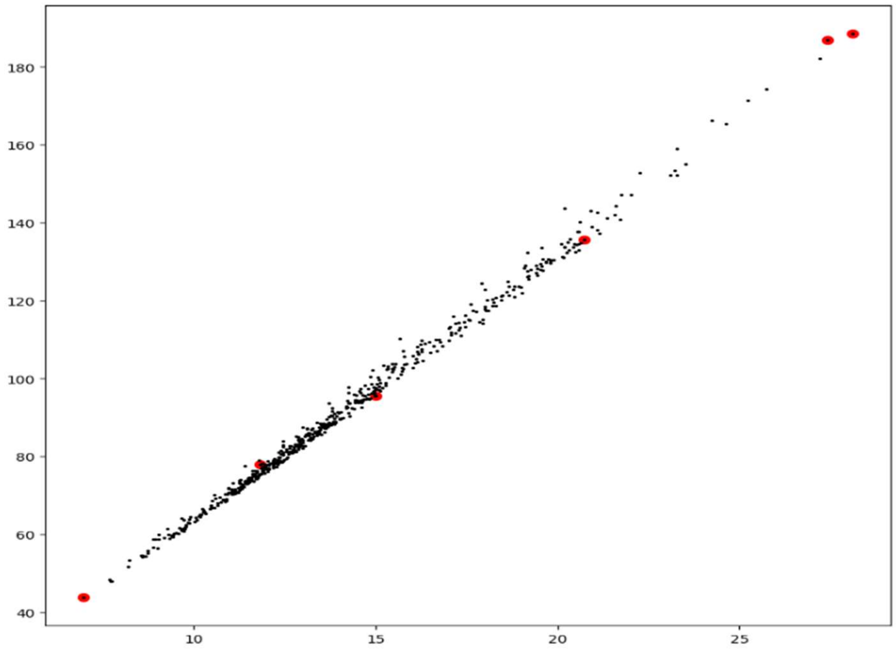

Figure 8. Outliers in the BCWD according to outlier scores.

5.4. Feature Reduction Using Correlation Analysis

This section presents the results of the initial filter phase, where features with low correlation to the target class were identified and removed. Figure 9 displays a heatmap of the correlation matrix for all 30 initial features, revealing that many features are highly correlated with each other, while some have a weak correlation with the target class. To systematically reduce the feature set, different correlation thresholds were evaluated, as summarized in Table 5. This table shows how the number of remaining features decreases as the correlation threshold increases. Based on this analysis, a threshold of 0.5 was selected for this study. Features with a correlation value below this threshold were considered weakly relevant or irrelevant and were subsequently discarded. This filtering process significantly reduced the feature space by half, from 30 initial features to 15 remaining features. The correlation heatmap for this reduced and more relevant feature set is shown in Figure 10.

Table 5: The effect of threshold value on features.

Number of features removed	**Number of remaining features**	**Threshold -value**
0	30	0
5	25	0.1
5	25	0.2
7	23	0.3
10	20	0.4

15	15	0.5
20	10	0.6
22	8	0.7
30	0	0.8

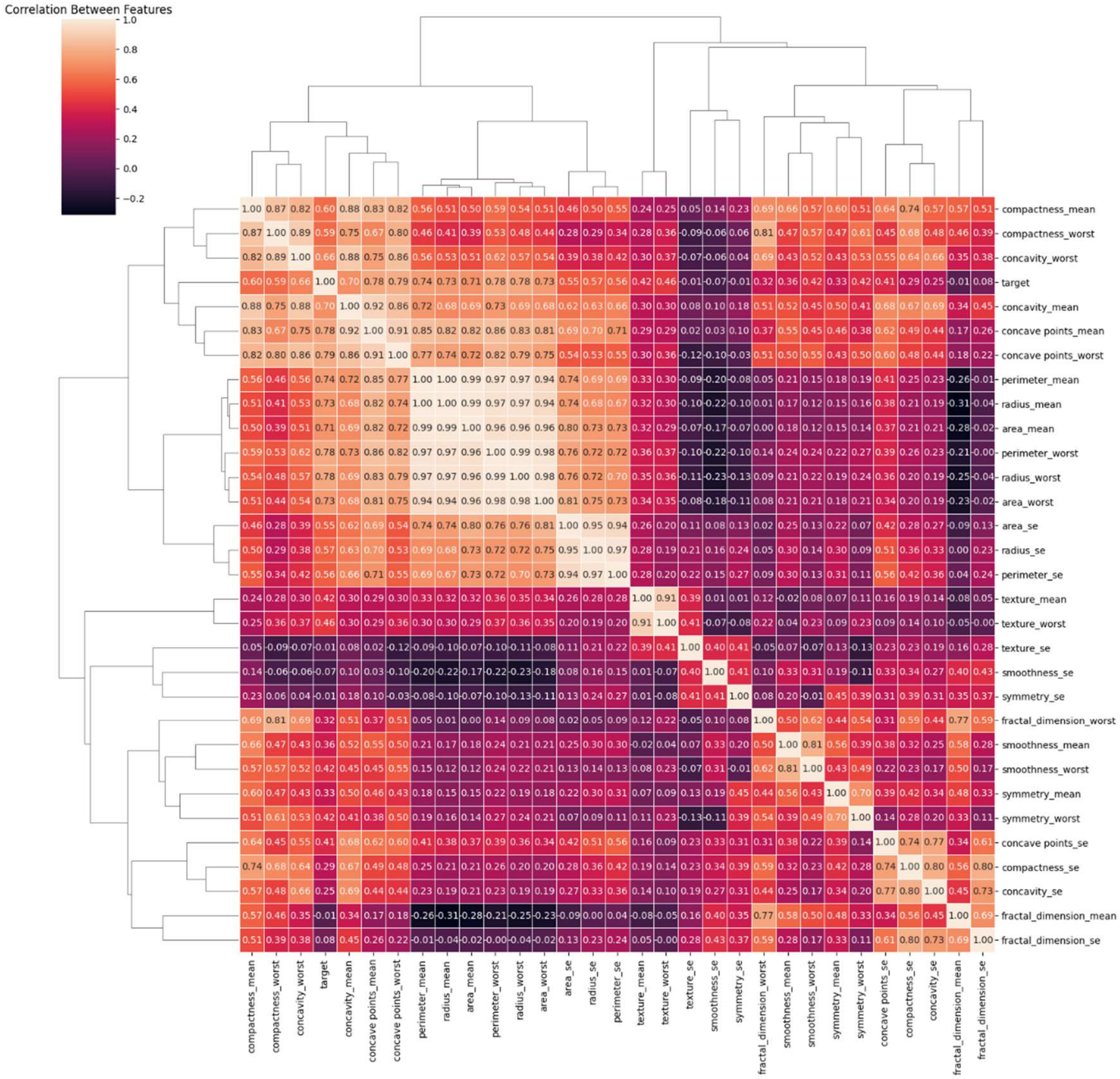

Figure 9. Correlation between features.

5.5. Impact of Feature Selection on Model Performance

To validate the effectiveness of the proposed RBSOFS method, this section compares the performance of the five classifiers under two scenarios: (1) using all 30 features (baseline "Without FS" model) and (2) using the optimal feature subset selected by RBSOFS. The results, summarized in Table 6, demonstrate the significant and positive impact of our feature selection method. Applying RBSOFS led to a substantial improvement in the accuracy of all five classifiers. For instance, the Random Forest (RF) and Logistic Regression (LR) models achieved the peak accuracy of 98.24% with the selected features, an increase of over 5% and 4% respectively compared to the baseline. The most dramatic improvement was observed with the Support Vector Machine (SVM) classifier, whose accuracy surged by 6.72% (from 91.22% to 97.94%) after feature selection. These findings confirm that removing irrelevant and redundant features via RBSOFS not only simplifies the models but also consistently and significantly enhances their predictive accuracy.

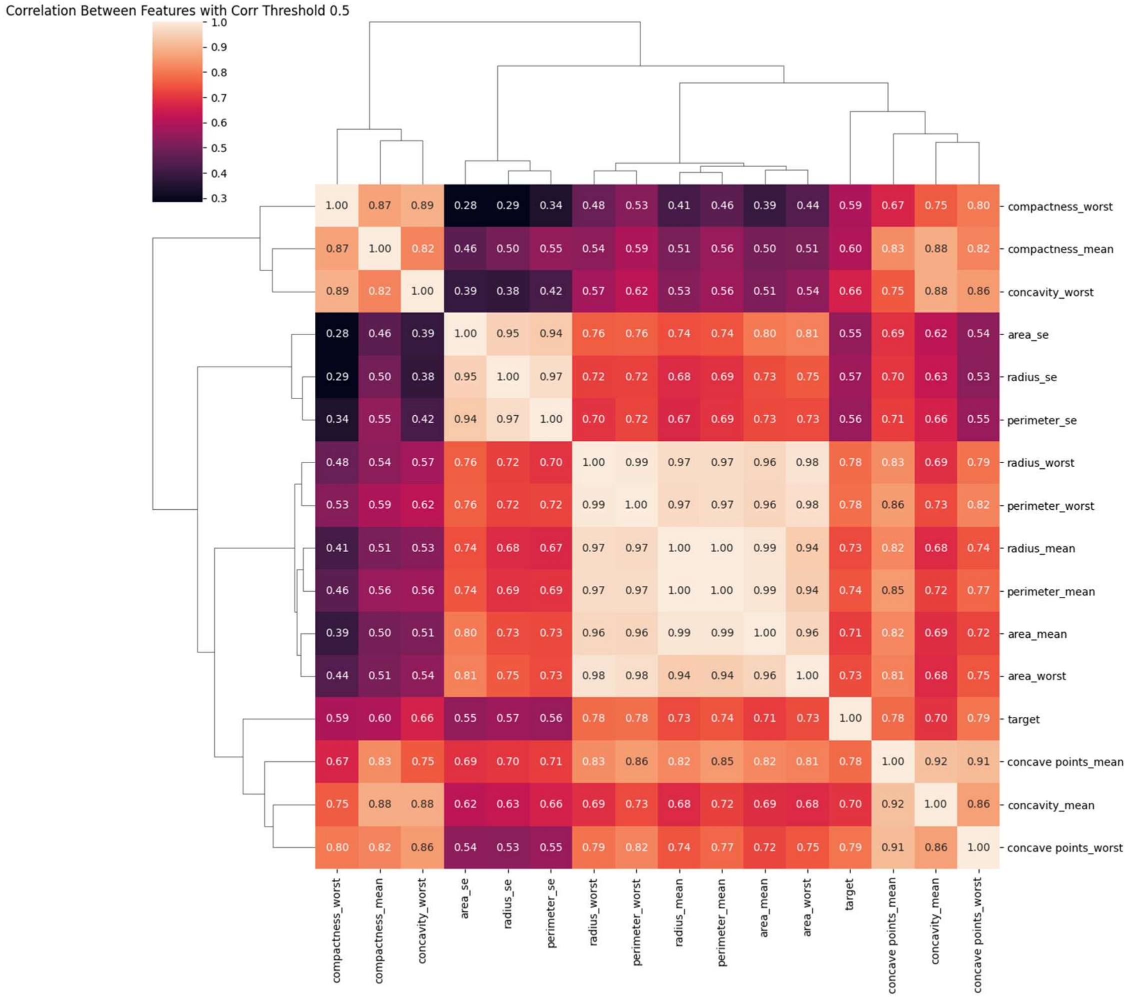

Figure 10. Correlation between feature with threshold 0.5.

Table 6: Accuracies obtained by the proposed method and without FS method.

RF	**NB**	**SVM**	**DT**	**LR**	**Classifier /Method**
98.24	97.13	97.94	97.07	98.24	RBSOFS
92.98	91.22	91.22	93.56	94.15	Without FS

5.6. Comparative Analysis and Discussion

This section presents a comprehensive comparative analysis of the proposed RBSOFS method against eight other state-of-the-art feature selection algorithms. The discussion is structured to highlight the superiority, stability, and underlying reasons for the method's success.

5.6.1. Superiority in Performance Metrics

As detailed in Tables 7 through 11, the RBSOFS method consistently outperformed all competing algorithms when paired with each of the five classifiers (LR, DT, SVM, NB, and RF). In nearly all test cases, RBSOFS achieved the highest accuracy, Recall, Precision, and F1-score.

The most significant finding, however, is that RBSOFS achieved these superior results while using the fewest features. A prime example is the comparison with the Logistic Regression (LR) classifier (Table 7): RBSOFS achieved 98.24% accuracy and 99.08% F1-score with only 6 features. In contrast, the next-best method (PSO) required 15 features to reach a lower accuracy of 96.42%. This ability to create a highly accurate yet far more compact model is the primary advantage of the proposed method and was a consistent trend across all classifiers. The high Recall (e.g., 98.18% with LR) and perfect Precision (100% with LR) further underscore the model's reliability for medical diagnosis, minimizing missed cases while avoiding false alarms.

Table 7: Comparison of the proposed method and competing metaheuristic methods based on evaluation criteria (LR Classifier).

F1-score	**Recall**	**Precision**	**#Features**	**Accuracy**	**Feature Selection Method + Classifier**
91.32	91.57	91.08	15	92.40	DE + LR
90.01	90.01	90.01	16	91.80	FA + LR
92.47	91.57	93.43	16	94.01	FPA + LR
94.43	93.28	95.61	15	94.90	GWO + LR
95.11	94.69	95.28	14	93.73	JA + LR
95.00	95.00	95.00	14	94.27	SCA + LR
93.11	92.98	93.25	16	94.73	SSA + LR
95.23	93.75	96.77	15	96.42	PSO + LR
99.08	**98.18**	**100**	**6**	**98.24**	RBSOFS+ LR

Table 8: Comparison of the proposed method and competing metaheuristic methods based on evaluation criteria (DT Classifier).

F1-score	**Recall**	**Precision**	**#Features**	**Accuracy**	**Feature Selection Method + Classifier**
88.94	89.39	88.51	16	92.39	DE + DT
93.09	92.66	93.53	16	94.32	FA + DT
91.93	90.47	93.44	17	92.80	FPA + DT

89.62	90.17	89.08	15	93.82	GWO + DT
94.39	94.53	94.26	15	92.80	JA + DT
93.72	93.44	94.02	16	92.45	SCA + DT
93.10	92.34	93.88	16	94.08	SSA + DT
94.48	92.30	96.77	14	95.66	PSO + DT
99.17	**98.36**	**100**	**6**	**97.07**	RBSOFS + DT

Table 9: Comparison of the proposed method and competing metaheuristic methods based on evaluation criteria (SVM Classifier).

F1-score	**Recall**	**Precision**	**#Features**	**Accuracy**	**Feature Selection Method + Classifier**
91.21	90.94	91.49	15	94.55	DE + SVM
95.61	95.47	95.76	16	95.43	FA + SVM
92.46	92.19	92.75	14	94.50	FPA + SVM
92.48	91.56	93.43	15	94.96	GWO + SVM
93.72	93.12	94.34	15	94.55	JA + SVM
91.86	90.78	92.98	16	95.08	SCA + SVM
94.39	94.53	94.26	15	95.66	SSA + SVM
96.08	95.45	96.72	14	97.65	PSO + SVM
98.43	**100**	**96.92**	**7**	**97.94**	RBSOFS + SVM

Table 10: Comparison of the proposed method and competing metaheuristic methods based on evaluation criteria (NB Classifier).

F1-score	**Recall**	**Precision**	**#Features**	**Accuracy**	**Feature Selection Method + Classifier**
88.78	89.69	87.84	16	91.39	DE + NB
89.35	89.23	89.48	14	92.45	FA + NB
90.71	88.90	92.60	15	91.04	FPA + NB
92.46	91.88	93.06	15	90.63	GWO + NB
90.56	89.85	91.29	15	92.04	JA + NB
90.60	90.47	90.74	17	92.80	SCA + NB
91.24	90.00	92.53	16	93.44	SSA + NB
92.62	89.85	95.58	12	95.37	PSO + NB
96.96	**96.96**	**96.96**	**6**	**97.13**	RBSOFS + NB

Table 11: Comparison of the proposed method and competing metaheuristic methods based on evaluation criteria (RF Classifier).

F1-score	**Recall**	**Precision**	**#Features**	**Accuracy**	**Feature Selection Method + Classifier**
92.50	92.50	92.50	15	94.71	DE + RF
94.43	94.37	94.51	13	94.90	FA + RF
93.10	92.34	93.88	18	94.49	FPA + RF
93.15	93.29	93.02	15	94.32	GWO + RF
89.96	88.75	91.21	15	94.09	JA + RF
93.09	92.66	93.53	18	94.26	SCA + RF
93.80	94.07	93.54	14	95.72	SSA + RF
95.58	94.28	96.92	15	97.52	PSO + RF
99.19	**100**	**98.41**	**7**	**98.24**	RBSOFS + RF

5.6.2. Stability and Consistency of Results

The stability of the proposed algorithm is visualized in the boxplots shown in Figures 11-15. These figures illustrate the distribution of accuracy results over multiple runs for each classifier. The boxplot for RBSOFS consistently shows a higher median accuracy (the red line), a smaller interquartile range (a tighter box, indicating less dispersion), and fewer outliers compared to the other methods. This indicates that RBSOFS is not only more accurate but also provides more stable and reliable performance. While PSO also showed acceptable stability, RBSOFS remained superior across all five classifier scenarios.

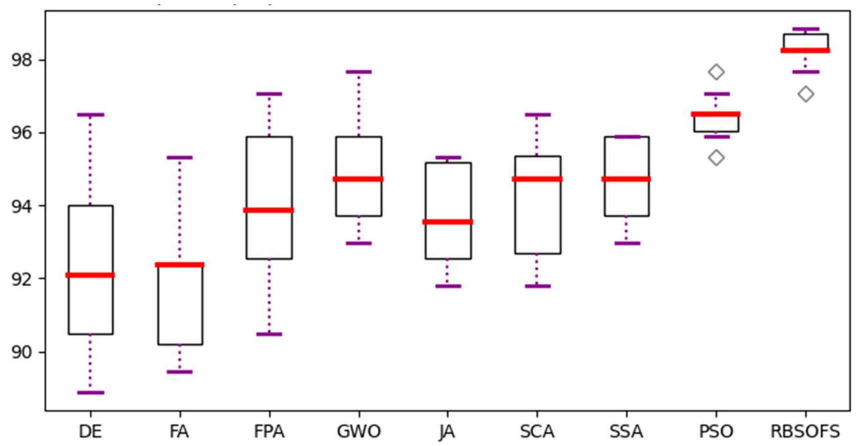

Figure 11. The boxplot of methods (LR Classifier).

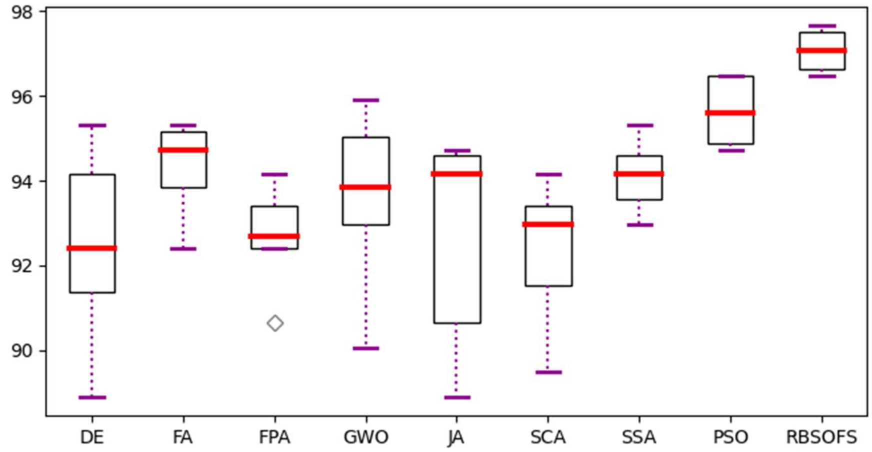

Figure 12. The boxplot of methods (DT Classifier).

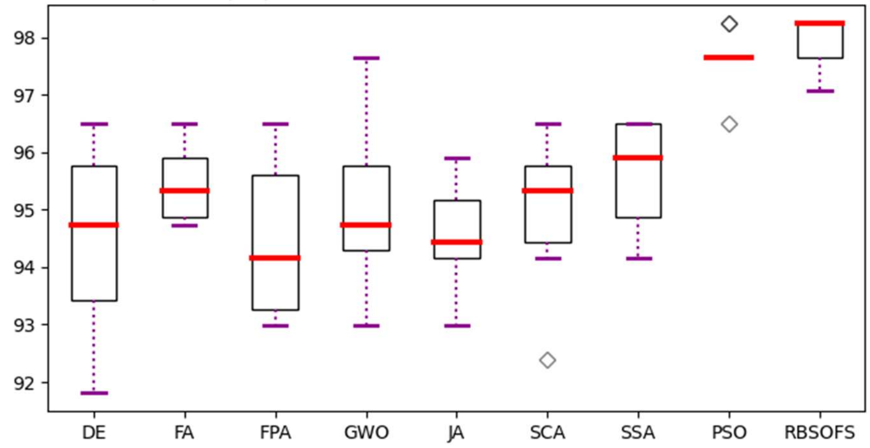

Figure 13. The boxplot of methods (SVM Classifier).

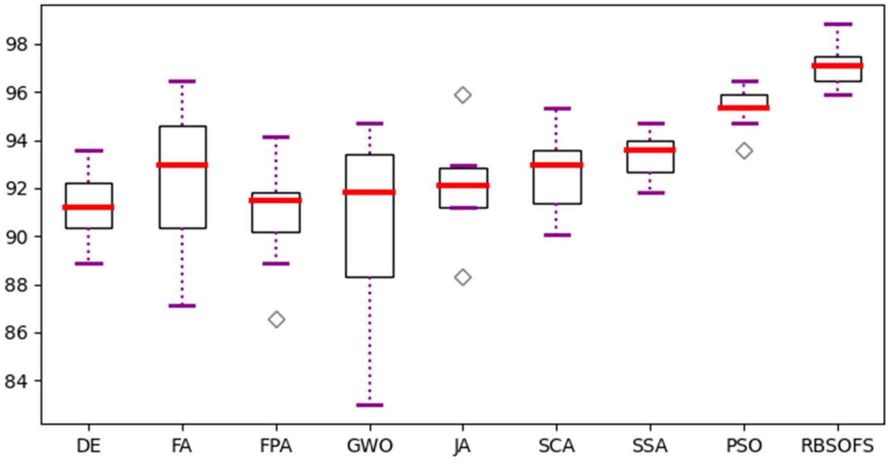

Figure 14. The boxplot of methods (NB Classifier).

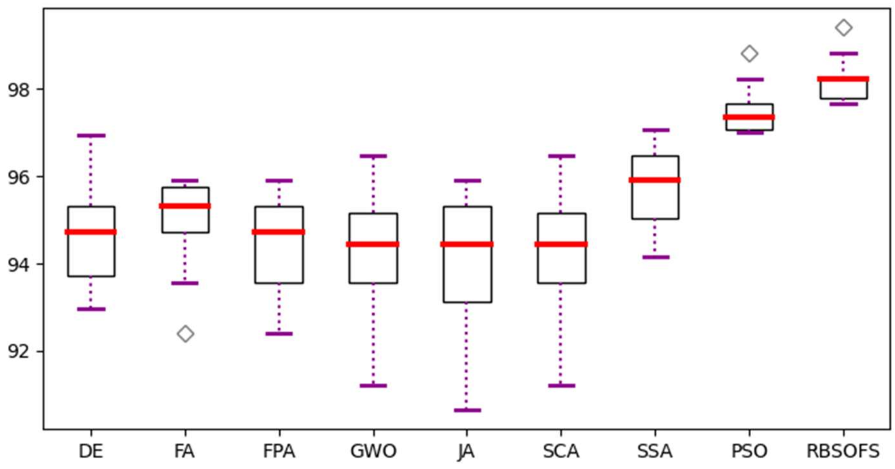

Figure 15. The boxplot of methods (RF Classifier).

5.6.3. Discussion

The consistent superiority of the RBSOFS method can be attributed to several key factors that create a synergistic and effective feature selection process. First, the hybrid two-stage design is central to its success. The initial filter phase acts as an efficient and low-cost mechanism to prune the search space. By removing a significant number of features that have a low individual correlation with the target class, it provides a smaller, more manageable, and higher-quality set of candidate features. This step is crucial because it allows the more computationally intensive wrapper phase to focus its search on a region of the solution space that is already known to have high potential. Second, this design creates an effective balance between exploration and exploitation, a critical aspect of any metaheuristic search. The filter stage can be viewed as a broad, high-level *exploration* that identifies the most promising area of the search space. The wrapper phase, driven by the sophisticated search mechanics of the Sailfish Optimizer, then performs a deep and focused *exploitation* within this refined area. This focused approach prevents the optimizer from wasting computational resources on unpromising regions and increases the probability of discovering a compact, globally near-optimal feature set. Finally, from a practical machine learning perspective, the drastic reduction in feature dimensionality has significant benefits. The parsimonious models produced by RBSOFS, which use as few as 6 features, are less susceptible to overfitting and are to generalize better to new, unseen data. From a clinical viewpoint, these simpler models are highly desirable as they are not only more computationally efficient but are also more interpretable, potentially reducing the number of clinical tests required for a diagnosis and increasing trust in the model's predictions.

6. Conclusion and future works

This study addressed the critical challenge of feature selection for breast cancer diagnosis from high-dimensional data. We introduced a novel hybrid method, Relevance-Based Feature Selection using Sailfish Optimizer (RBSOFS), which effectively identifies small, yet highly discriminative, feature subsets. The experimental results validated the superiority of the RBSOFS framework. For example, by applying the LR classifier to the Breast Cancer Wisconsin (Diagnostic) dataset, the proposed method achieved a peak classification accuracy of 98.24% while utilizing a minimal subset of only 6 to 7 features. This performance, which represents an optimal balance between accuracy and model complexity, consistently surpassed eight other state-of-the-art algorithms across five different classifiers. The success of this approach stems from its effective two-stage design, where a filter pre-processes the search space, allowing the optimizer to efficiently pinpoint the most potent feature combination. In conclusion, RBSOFS proves to be a robust and efficient tool for building simpler and more interpretable models for breast cancer diagnosis, offering significant potential for clinical application.

While the proposed method has demonstrated strong performance, it is important to acknowledge its limitations, which in turn suggest clear avenues for future research. First, the initial filter stage relies on univariate correlation and might overlook features that are valuable only through complex interactions; future work could explore more sophisticated filtering techniques. Second, like all metaheuristics, RBSOFS is stochastic and does not guarantee finding the global optimum; therefore, further enhancements to the optimization algorithm itself present another research direction. Finally, the method was validated on a single benchmark dataset. The most critical next steps involve applying the RBSOFS framework to other complex medical datasets to test its generalizability, perhaps by integrating it with deep learning models. Furthermore, a collaboration with medical experts to provide a deeper biological interpretation of the selected features would be invaluable for further validating the clinical applicability of the model and enhancing trust in its predictions.

Mohammad Ansari Shiri is a PhD candidate in the Department of Computer Science at Shahid Bahonar University of Kerman. Programming, Ideas, Writing- original draft preparation.

Najme Mansouri is currently a faculty of Computer Science at Shahid Bahonar University of Kerman. Testing of existing code components, Writing- Reviewing and Editing

Declarations

Ethics approval and consent to participate

Not applicable

Consent for publication

Not applicable

Availability of data and materials

The datasets generated and/or analyzed during the current study are not publicly available due, but are available from the corresponding author on reasonable request.

Competing interests

The authors declare that they have no competing interests

Funding

Not applicable

Acknowledgements

Not applicable

References

[1] Kaur, S., Kumar, Y., Koul, A., & Kumar Kamboj, S. (2023). A systematic review on metaheuristic optimization techniques for feature selections in disease diagnosis: open issues and challenges. *Archives of Computational Methods in Engineering*, *30*(3), 1863-1895.

[2] Sayed, G. I., Darwish, A., & Hassanien, A. E. (2020). Binary whale optimization algorithm and binary moth flame optimization with clustering algorithms for clinical breast cancer diagnoses. *Journal of Classification*, *37*, 66-96.

[3] Chatterjee, S., Biswas, S., Majee, A., Sen, S., Oliva, D., & Sarkar, R. (2022). Breast cancer detection from thermal images using a Grunwald-Letnikov-aided Dragonfly algorithm-based deep feature selection method. *Computers in biology and medicine*, *141*, 105027.

[4] Zhao, H., Sinha, A. P., & Ge, W. (2009). Effects of feature construction on classification performance: An empirical study in bank failure prediction. *Expert Systems with Applications*, *36*(2), 2633-2644.

[5] Chen, H., Li, T., Fan, X., & Luo, C. (2019). Feature selection for imbalanced data based on neighborhood rough sets. *Information sciences*, *483*, 1-20.

[6] Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of machine learning research*, *3*(Mar), 1157-1182.

[7] Hegazy, A. E., Makhlouf, M. A., & El-Tawel, G. S. (2019). Feature selection using chaotic salp swarm algorithm for data classification. *Arabian Journal for Science and Engineering*, *44*, 3801-3816.

[8] Too, J., & Abdullah, A. R. (2020). Chaotic atom search optimization for feature selection. *Arabian Journal for Science and Engineering*, *45*(8), 6063-6079.

[9] Chakraborty, S., Saha, A. K., Chakraborty, R., & Saha, M. (2021). An enhanced whale optimization algorithm for large scale optimization problems. *Knowledge-Based Systems*, *233*, 107543.

[10] Kelidari, M., & Hamidzadeh, J. (2021). Feature selection by using chaotic cuckoo optimization algorithm with levy flight, opposition-based learning and disruption operator. *Soft Computing*, *25*(4), 2911-2933.

[11] Agrawal, P., Ganesh, T., & Mohamed, A. W. (2021). Chaotic gaining sharing knowledge-based optimization algorithm: an improved metaheuristic algorithm for feature selection. *Soft Computing*, *25*(14), 9505-9528.

[12] Ma, W., Zhou, X., Zhu, H., Li, L., & Jiao, L. (2021). A two-stage hybrid ant colony optimization for high-dimensional feature selection. *Pattern Recognition*, *116*, 107933.

[13] Alweshah, M., Alkhalaileh, S., Albashish, D., Mafarja, M., Bsoul, Q., & Dorgham, O. (2021). A hybrid mine blast algorithm for feature selection problems. *Soft Computing*, *25*, 517-534.

[14] Sharma, M., & Kaur, P. (2021). A comprehensive analysis of nature-inspired meta-heuristic techniques for feature selection problem. *Archives of Computational Methods in Engineering*, *28*, 1103-1127.

[15] Ahmed, S., Ghosh, K. K., Mirjalili, S., & Sarkar, R. (2021). AIEOU: Automata-based improved equilibrium optimizer with U-shaped transfer function for feature selection. *Knowledge-Based Systems*, *228*, 107283.

[16] Srichandan, S., Kumar, T. A., & Bibhudatta, S. (2018). Task scheduling for cloud computing using multi-objective hybrid bacteria foraging algorithm. *Future Computing and Informatics Journal*, *3*(2), 210-230.

[17] Dokeroglu, T., Sevinc, E., Kucukyilmaz, T., & Cosar, A. (2019). A survey on new generation metaheuristic algorithms. *Computers & Industrial Engineering*, *137*, 106040.

[18] Talbi, E. G. (2009). *Metaheuristics: from design to implementation*. John Wiley & Sons.

[19] Agrawal, P., Abutarboush, H. F., Ganesh, T., & Mohamed, A. W. (2021). Metaheuristic algorithms on feature selection: A survey of one decade of research (2009-2019). *Ieee Access*, *9*, 26766-26791.

[20] Meraihi, Y., Gabis, A. B., Ramdane-Cherif, A., & Acheli, D. (2021). A comprehensive survey of Crow Search Algorithm and its applications. *Artificial Intelligence Review*, *54*(4), 2669-2716.

[21] Beni, G., & Wang, J. (1993). Swarm intelligence in cellular robotic systems. In *Robots and biological systems: towards a new bionics?* (pp. 703-712). Berlin, Heidelberg: Springer Berlin Heidelberg.

[22] Ting, S. L., Ip, W. H., & Tsang, A. H. (2011). Is Naive Bayes a good classifier for document classification. *International Journal of Software Engineering and Its Applications*, *5*(3), 37-46.

[23] Saritas, M. M., & Yasar, A. (2019). Performance analysis of ANN and Naive Bayes classification algorithm for data classification. *International journal of intelligent systems and applications in engineering*, *7*(2), 88-91.

[24] Mountrakis, G., Im, J., & Ogole, C. (2011). Support vector machines in remote sensing: A review. *ISPRS journal of photogrammetry and remote sensing*, *66*(3), 247-259.

[25] Istia, S. S., & Purnomo, H. D. (2018, November). Sentiment analysis of law enforcement performance using support vector machine and K-nearest neighbor. In *2018 3rd International Conference on Information Technology, Information System and Electrical Engineering (ICITISEE)* (pp. 84-89). IEEE.

[26] Criminisi, A., Shotton, J., & Konukoglu, E. (2012). Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning. *Foundations and trends® in computer graphics and vision*, *7*(2–3), 81-227.

[27] Zhang, K., Cheng, Y., Xie, Y., Honbo, D., Agrawal, A., Palsetia, D., ... & Choudhary, A. (2011, December). SES: Sentiment elicitation system for social media data. In *2011 IEEE 11th international conference on data mining workshops* (pp. 129-136). IEEE.

[28] Safavian, S. R., & Landgrebe, D. (1991). A survey of decision tree classifier methodology. *IEEE transactions on systems, man, and cybernetics*, *21*(3), 660-674.

[29] Robert, C. (2014). Machine learning, a probabilistic perspective.

[30] Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science*, *349*(6245), 255-260.

[31] Chen, J., Li, K., Tang, Z., Bilal, K., Yu, S., Weng, C., & Li, K. (2016). A parallel random forest algorithm for big data in a spark cloud computing environment. *IEEE Transactions on Parallel and Distributed Systems*, *28*(4), 919-933.

[32] Snoek, J., Larochelle, H., & Adams, R. P. (2012). Practical bayesian optimization of machine learning algorithms. *Advances in neural information processing systems*, *25*.

[33] Akinola, O. O., Ezugwu, A. E., Agushaka, J. O., Zitar, R. A., & Abualigah, L. (2022). Multiclass feature selection with metaheuristic optimization algorithms: a review. *Neural Computing and Applications*, *34*(22), 19751-19790.

[34] Faramarzi, A., Heidarinejad, M., Stephens, B., & Mirjalili, S. (2020). Equilibrium optimizer: A novel optimization algorithm. *Knowledge-Based Systems*, *191*, 105190.

[35] Tubishat, M., Abushariah, M.A., Idris, N. and Aljarah, I., Improved whale optimization algorithm for feature selection in Arabic sentiment analysis. Applied Intelligence, 49(5), 2019, 1688-1707.

[36] Bacanin, N., Venkatachalam, K., Bezdan, T., Zivkovic, M., & Abouhawwash, M. (2023). A novel firefly algorithm approach for efficient feature selection with COVID-19 dataset. *Microprocessors and Microsystems*, *98*, 104778.

[37] Al-Wajih, R., Abdulkadir, S. J., Aziz, N., Al-Tashi, Q., & Talpur, N. (2021). Hybrid binary grey wolf with Harris hawks optimizer for feature selection. *IEEE Access*, *9*, 31662-31677.

[38] Tiwari, A., & Chaturvedi, A. (2022). A hybrid feature selection approach based on information theory and dynamic butterfly optimization algorithm for data classification. *Expert Systems with Applications*, *196*, 116621.

[39] Zhou, Y., Zhang, W., Kang, J., Zhang, X., & Wang, X. (2021). A problem-specific non-dominated sorting genetic algorithm for supervised feature selection. *Information Sciences*, *547*, 841-859.

[40] Kılıç, F., Kaya, Y. and Yildirim, S., A novel multi population based particle swarm optimization for feature selection. Knowledge-Based Systems, 219, 2021, 106894.

[41] Got, A., Moussaoui, A. and Zouache, D., Hybrid filter-wrapper feature selection using whale optimization algorithm: A multi-objective approach. *Expert Systems with Applications*, *183*, 2021, 115312.

[42] El_Rahman, S. A. (2021). Predicting breast cancer survivability based on machine learning and features selection algorithms: a comparative study. *Journal of Ambient Intelligence and Humanized Computing*, *12*, 8585-8623.

[43] Dey, C., Bose, R., Ghosh, K. K., Malakar, S., & Sarkar, R. (2022). LAGOA: Learning automata based grasshopper optimization algorithm for feature selection in disease datasets. *Journal of Ambient Intelligence and Humanized Computing*, 1-20.

[44] Kalita, D. J., Singh, V. P., & Kumar, V. (2022). Two-way threshold-based intelligent water drops feature selection algorithm for accurate detection of breast cancer. *Soft Computing*, *26*(5), 2277-2305.

[45] Bansal, P., Vanjani, A., Mehta, A., Kavitha, J. C., & Kumar, S. (2022). Improving the classification accuracy of melanoma detection by performing feature selection using binary Harris hawks optimization algorithm. *Soft Computing*, *26*(17), 8163-8181.

[46] Alrefai, N., & Ibrahim, O. (2022). Optimized feature selection method using particle swarm intelligence with ensemble learning for cancer classification based on microarray datasets. *Neural Computing and Applications*, *34*(16), 13513-13528.

[47] Liu, Y., Zou, X., Ma, S., Avdeev, M., & Shi, S. (2022). Feature selection method reducing correlations among features by embedding domain knowledge. *Acta Materialia*, *238*, 118195.

[48] Vergara, J. R., & Estévez, P. A. (2014). A review of feature selection methods based on mutual information. *Neural computing and applications*, *24*, 175-186.

[49] Wang, L., Jiang, S., & Jiang, S. (2021). A feature selection method via analysis of relevance, redundancy, and interaction. *Expert Systems with Applications*, *183*, 115365.

[50] Shadravan, S., Naji, H. R., & Bardsiri, V. K. (2019). The Sailfish Optimizer: A novel nature-inspired metaheuristic algorithm for solving constrained engineering optimization problems. *Engineering Applications of Artificial Intelligence*, *80*, 20-34.

[51] Wolberg,William, Mangasarian,Olvi, Street,Nick, and Street,W.. (1995). Breast Cancer Wisconsin (Diagnostic). UCI Machine Learning Repository.

[52] Amarnath, B., Balamurugan, S., & Alias, A. (2016). Review on feature selection techniques and its impact for effective data classification using UCI machine learning repository dataset. *Journal of Engineering Science and Technology*, *11*(11), 1639-1646.