Optimized Feature Selection Using Modified Social Group Optimization

Y. V. Nagesh Meesala1[1,] Ajaya Kumar Parida[2], Anima Naik[3, *]
[1]Research Scholar, School of Computer Engineering, KIIT Deemed to be University, Department of CSE, Raghu Engineering College
[2] School of Computer Engineering, KIIT Deemed to be University
[3] Department of CSE, Raghu Engineering College, Visakhapatnam Andhra Pradesh 530003, India
E-mail: 2181072@kiit.ac.in, nagesh.myv@raghuenggcollege.in, ajaya.paridafcs@kiit.ac.in, anima.naik@raghuenggcollege.in
[*]Corresponding author

Keywords: FS, SGO, datasets, optimization algorithm, CA

Received: May 8, 2024

This paper introduces binary variants of the Modified Social Group Optimization (MSGO) algorithm designed specifically for optimal feature subset selection in a wrapper-mode classification setting. While the original SGO was proposed in 2016 and modified in 2020 to enhance its performance, it was not previously applied to feature selection problems. MSGO represents an advancement over SGO, adept at efficiently exploring the feature space to identify optimal or near-optimal feature subsets by minimizing a specified fitness function. The two newly proposed binary variants of MSGO are employed to identify the optimal feature combinations that maximize classification accuracy while minimizing the number of selected features. In these variants, the native MSGO is utilized while its continuous steps are bounded in a threshold using a suitable threshold function after squashing them. These binary algorithms are compared against six latest high-performing optimization approaches and six state-of-the-art optimization algorithms to assess their performance. Various evaluation metrics are utilized across twenty-three datasets sourced from the UCI data repository to accurately judge and compare the efficacy of these algorithms. The experimental results confirm the efficiency of the proposed approaches in improving the classification accuracy compared to other wrapper-based algorithms, which proves the ability of the MSGO algorithm to search the feature space and select the most informative attributes for classification tasks.

Povzetek: Predstavljene so različice izboljšanega algoritma socialne skupinske optimizacije (MSGO) za izbiro optimalnih podskupin lastnosti, kar poveča natančnost klasifikacije in zmanjša število izbranih lastnosti.

1 Introduction

Features or attributes are crucial elements that define key characteristics within a dataset. Feature selection (FS) stands out as a critical step in data pre-processing for both machine learning and data mining. Its primary function is to identify and select a relevant subset of features from the original dataset. Mathematically, this can be expressed as selecting a subset *S* from the set of all features *F* such that:

$$S \subseteq F$$

where *S* represents the selected subset of features and *F* denotes the entire set of features available in the dataset. The goal of feature selection is to retain only the most informative and discriminative features while discarding redundant or irrelevant ones, thereby enhancing the efficiency and accuracy of subsequent machine learning or data mining algorithms.

The primary objective of data pre-processing in data mining and machine learning is to prepare the dataset for knowledge extraction using algorithms from these fields. Classification and clustering algorithms are fundamental in data mining, operating on dataset dimensions to make predictions. However, increasing the dataset's dimensions often leads to decreased performance in these algorithms [1]. Real-world data frequently contains noisy, irrelevant, or misleading features, making it challenging to extract meaningful insights. Handling imprecise and inconsistent information has become a crucial requirement in addressing real-world problems. Feature selection (FS) is a key pre-processing step aimed at selecting a subset of features from the original set. This subset should adequately describe target concepts while maintaining high accuracy in representing the original features. FS methods can be categorized into two main types: filter and wrapper [3]. Filter-based methods assess features based on predefined criteria like information gain [4], principal component analysis [5-7], mutual information [8], Relief [9], Chi-square [10], Fisher Score [11], Laplacian score [12], etc., and select the most important features accordingly. Conversely, wrapper methods employ machine learning algorithms to evaluate feature subsets and select the optimal subset for the task at hand. Filter methods tend to be faster since they don't require learning algorithms, whereas wrapper methods generally achieve higher accuracy [13-14].

Over the last two decades, meta-heuristic algorithms have gained significant popularity among optimization researchers. This is attributed to their ability to avoid local optima, their gradient-free mechanism, and their flexibility. Meta-heuristic algorithms typically exhibit two key characteristics: exploration or diversification, which involves searching the entire solution space to find the best solution in each iteration and avoiding local optima, and exploitation or intensification, which refers to finding a better solution near the current solution, leading to faster convergence. A well-designed meta-heuristic algorithm strikes a balance between exploration and exploitation. Many researchers have leveraged meta-heuristic algorithms to tackle feature selection (FS) problems. Examples include simulated annealing [15], tabu search [16], Particle Swarm Optimization (PSO) [17], artificial bee colony (ABC) [18], and Genetic Algorithm (GA) [19]. Additionally, methods like attribute reduction algorithms using rough set theory [20], graph-based FS using ant colony optimization [21], FS methods based on rough set theory with teaching learning-based optimization (TLBO) [23-23], hybridization of rough set and differential evolution (DE) techniques [24], and integration of ABC and DE for FS [25] have been proposed and validated using datasets from the UCI repository. Moreover, newer meta-heuristic algorithms such as Grey Wolf Optimizer (GWO) [26], flower pollination algorithm [27], Dragonfly Algorithm (DA) [28], Whale Optimization Algorithm (WOA) [29], and combinations like SA integrated with WOA [30], have also shown success in solving FS problems.

The inherent randomness of meta-heuristic algorithms means there is no guarantee they will discover the optimal feature subset in FS problems. This uncertainty is supported by the No-Free-Lunch theorem, which asserts that no single optimization algorithm can universally solve all optimization problems [30]. This realization led us to investigate the efficacy of the modified social group optimization (MSGO) algorithm [31]. The original SGO algorithm was introduced in 2016, inspired by human social behavior in problem-solving [46]. SGO has garnered attention for its potential in global optimization across various applications [32-38] and has shown superior performance compared to other algorithms [39]. Surprisingly, SGO hadn't been applied to FS problems until now, prompting us to choose it as the foundation for our work. The modified version of SGO, MSGO, introduced by [40] with the parameter "SAP (Self-Awareness Probability)," aims to enhance algorithm performance. MSGO's performance was evaluated against twenty-five algorithms, including GA, PSO, DE, ABC, as well as newer ones like HHO (Harris Hawks Optimization) [41], BOA (Butterfly Optimization Algorithm) [42], SSOA (Squirrel Search Optimization Algorithm) [43], GROM (Golden Ratio Optimization Method) [44], VPL (Volleyball Premier League Algorithm) [45], etc. Given MSGO's improved performance over SGO, we opted to utilize MSGO for our FS problem.

The comprehensive aim of this paper is to propose new binary versions of modified social group optimization algorithm (bmSGO) for wrapper FS. The proposed algorithms select the optimal feature subset which decreases the feature subset length and at the same time, increases the classification accuracy.

The key contributions of the paper can be summarized as follows:

- Two binary variants of the MSGO are proposed.
- Two transfer functions are used to map the continuous search space to discrete one.
- Twenty-one UCI datasets are utilized in the experiments.
- A superior performance of the proposed binary variants is proved in the experiments.

The rest of the paper is organized as follows: Section 2 details the proposed bmSGO, while section 3 presents simulation and experimental results. Finally, section 4 concludes the work and discusses future directions.

2 Related works

2.1 Wrapper-Mode classification setting

Wrapper-mode selection is an advanced feature selection methodology that optimizes the predictive performance of a model by directly integrating the feature selection process with the model training. This approach utilizes the learning algorithm to evaluate and select feature subsets, resulting in a specific feature set that ideally enhances model accuracy and generalizability.

Key characteristics

- **Model-Centric evaluation**: Wrapper methods involve the generation of various subsets of features, each evaluated based on the performance of a chosen learning algorithm. The selection process is embedded within the training algorithm, making it an iterative and adaptive procedure. Each subset's performance is measured using metrics such as accuracy, precision, recall, or F1-score, which guides the feature selection process.

- **Iterative subset selection**:

    - **Forward selection**: This approach begins with an empty set and progressively adds features that improve model performance.

    - **Backward elimination**: This starts with the complete set of features, sequentially removing the least significant features to enhance performance.

- **Recursive feature elimination (RFE)**: This method iteratively constructs the model, removing the least important feature(s) in each iteration until the optimal subset is identified.

- **Performance metrics and validation**: To ensure the robustness of the selected feature subset, cross-validation techniques are typically employed. This step is crucial to mitigate overfitting and to ensure that the model generalizes well to unseen data. The selection criterion is based on optimizing predefined performance metrics, ensuring the chosen features contribute significantly to the model's predictive power.

2.2 Modified social group optimization (MSGO) algorithm

The MSGO algorithm is a modified version of the Social Group Optimization algorithm, where a concept has been introduced in the form that " a person acquires something new from other persons if another person has more knowledge, and he or she has the higher self-awareness probability (SAP) to achieve that knowledge". SAP defines the ability to acquire a quantity of knowledge from another person. So an extra parameter in the form of SAP is introduced in the MSGO algorithm. For a detailed description of the MSGO algorithm, please refer to the paper [40]. The MSGO algorithm, in short, is given below:

Let P_i, i=1,2,3,…., N be the persons of the social group, i.e., the social group contains N persons and each person P_i is defined by $P_i = (p_{i1}, p_{i2}, p_{i3}, \dots \dots, p_{iD})$ where D is the number of traits assigned to a person which determine dimensions of a person and f_i, i=1,2,……, N are their corresponding fitness value, respectively.

Improving Phase

The best person in the group ($best_P$) in each social group tries to propagate knowledge among all persons, which will, in turn, help others to improve their knowledge in the group.

[minvalue, index]=min$\{f(P_i), i = 1,2,3, \dots \dots \dots \dots, N\}$
$best_P$=P(index,:)
for solving the minimization problem

In the improving phase, each person gets knowledge from the group's best ($best_P$) person. The updating of each person can be computed as follows:

Algorithm 1: The Improving phase

For i= 1:N
Forj=1:D
$\boldsymbol{Pnew_{ij} = c * P_{ij} + rand * (best_P(j) - P_{ij})}$
End for
End for
Accept P*new* if it gives a better fitness than P

Where $rand$ is a random number, $rand \sim U(0,1)$, and c is known as self- introspection parameter lies in between 0 and 1.

Acquiring phase

As we know in the acquiring phase a person of social group interacts with the best person ($best_P$) of that group and also interacts randomly with other persons of the group for acquiring knowledge. A person acquires new knowledge if the other person has more knowledge. The $best_P$ is always best than others, so a person always acquires knowledge from $best_P$. A person acquires something new from other persons if other person has more knowledge, and he or she has a higher self-awareness probability (SAP) to achieve that knowledge. Self-Awareness probability (SAP) defines the ability to acquire a quantity of knowledge from other person. So the modified acquiring phase is expressed as

[value, index_num]=min$\{f(P_i), i = 1,2,3, \dots \dots \dots \dots, N\}$
$best_P$= P(index_num,:)

for solving minimization problem, where P_i's are updated value at the end of the improving phase.

Algorithm 2: The acquiring phase

For i = 1 : N
Randomly select one person P_r, where $i \neq r$
If f (P_i) < f (P_r)
If rand>SAP
For j=1:D
$Pnew_{i,j} = P_{i,j} + rand_1 * (P_{i,j} - P_{r,j}) + rand_2 * (best_P(j) - P_{i,j})$
End for
Else
For j=1:D
$Pnew_{i,:} = lb + rand * (ub - lb)$
End for
end if
Else
For j=1:D
$Pnew_{i,j} = P_{i,j} + rand_1 * (P_{r,j} - P_{i,j}) + rand_2 * (best_P(j) - P_{i,j})$
End for
End If
End for
Accept $Pnew$ if it gives a better fitness than P

Where $rand_1$ and $rand_2$ are two independent random numbers, $rand_1 \sim U(0,1)$, and $rand_2 \sim U(0,1)$. These random numbers are used to affect the stochastic nature of the algorithm; lb and ub are the lower bound and upper bound of the corresponding design variable and SAP lie in between 0.6 and 0.9.

2.3 The proposed binary modified social group optimization

The MSGO algorithm, a modification of the Social Group Optimization algorithm, introduces the concept that individuals gain new knowledge from others with higher knowledge levels, based on their self-awareness probability (SAP)[40]. For a detailed description of MSGO, please refer to the original paper [40]. Wrapper-based FS methods typically employ a classifier for training and evaluation at each generation, necessitating a self-regulating optimization algorithm to minimize evaluations. The documented advantages of MSGO inspired us to propose its use as a search method in wrapper-based FS processes. Considering the binary nature of FS problems, where the search space involves binary values [0, 1], and the simplicity of binary operators compared to continuous ones, we introduced a binary version of MSGO (bmSGO) to tackle the FS problem.
The core concept of bmSGO revolves around updating individuals' positions in a binary search space using a transfer function. To accomplish this, we employ two transfer functions: Sigmoid (S-shaped) and V-shaped transfer functions, creating two binary versions known as S-bmSGO and V-bmSGO, respectively.

2.3.1 Binary MSGO algorithm-approach 1 (S-bmSGO)

In continuous MSGO, individuals' positions are represented by continuous solutions, which need to be transformed into corresponding binary values. This transformation involves squashing continuous solutions in each dimension using a Sigmoidal (S-shaped) transfer function[46], which compels individuals to navigate in a binary search space. The S-shaped function is expressed as follows in equation (1) and is illustrated in Figure 1.

$$S\left(P_i^j(k)\right) = \frac{1}{1+e^{-P_i^j(k)}} \quad \text{..................................(1)}$$

where $P_i^j(k)$ is the position of i^{th} person in j^{th} dimension at iteration k.

To reach the binary solution of the output of the S-shaped transfer function, a threshold is applied as mentioned in Eq. (2).

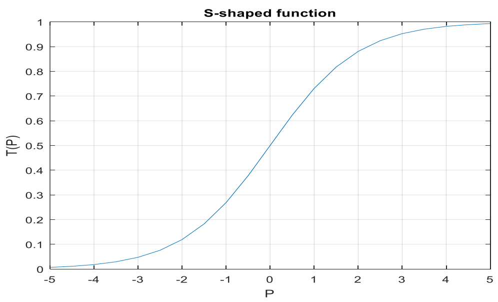


Figure 1: Sigmoid transfer function

$$TP_i^j(k) = \begin{cases} 0 & if \quad rand < S\left(P_i^j(k)\right) \\ 1 & if \quad rand \geq S\left(P_i^j(k)\right) \end{cases} \quad \text{........ (2)}$$

where $TP_i^j(k)$ and $P_i^j(k)$ indicate the transfer binary position and position of ith person in jth dimension at iteration k.

2.3.2 Binary MSGO algorithm-approach 2 (V-bmSGO)

This approach introduces a V-shaped transfer function, which is implemented using equations (3) and (4) [47]. The process of utilizing this proposed transfer function to guide individuals in navigating a binary search space is depicted in Figure 2.

$$V\left(P_i^j(k)\right) = \left|\text{erf}\left(\frac{\sqrt{\pi}}{2} P_i^j(k)\right)\right|$$
$$= \left|\frac{\sqrt{\pi}}{2}\int_0^{\frac{\sqrt{\pi}}{2}P_i^j(k)} e^{-k^2} dk\right| \quad \text{.........(3)}$$

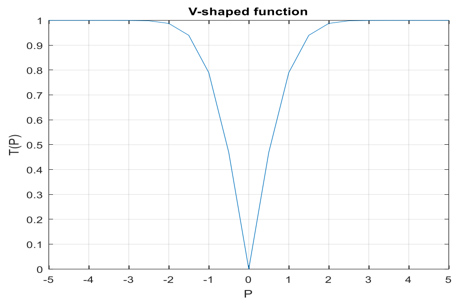


Figure 2: V-shaped transfer function

The threshold rules can be represented mathematically as:

$$TP_i^j(k) = \begin{cases} 0 & if \quad rand < V\left(P_i^j(k)\right) \\ 1 & if \quad rand \geq V\left(P_i^j(k)\right) \end{cases} \quad \text{.......... (4)}$$

where $TP_i^j(k)$ and $P_i^j(k)$ indicate the transfer binary position and position of i^{th} person in j^{th} dimension at iteration k.

The general steps of bmSGO are presented in Algorithm 3, the framework of the bmSGO algorithm is given in Fig 3.

Algorithm 3: Pseudo-code for bmSGO

Define self introspection parameter C, Self-Awareness probability SAP, Maximum iteration= Max_iter, pop_size=N, dimension=D
Initialize population P ‘
Evaluate each person P_i in the population
Find the best solution f^* and best person ‘gbest’
t=1
While (t< Max_iter)
For each person P_i in the population
Find newP_i using improving phase
Transform real-valued to binary one using S-shaped/V-shaped transformation
Evaluate using a fitness function
If newP_i is better than P_i then replace it and update the population
End for
Find the best solution f^* and best person gbest
For each person P_i in the population
Find newP_i using acquiring phase

Transform real-valued to binary one using S-shaped/V-shaped transformation
Evaluate using a fitness function
If newP_i is better than P_i then replace it and update the population
End for

Find the best solution f^* and best person gbest
t=t+1
end while
return the best solution f^*

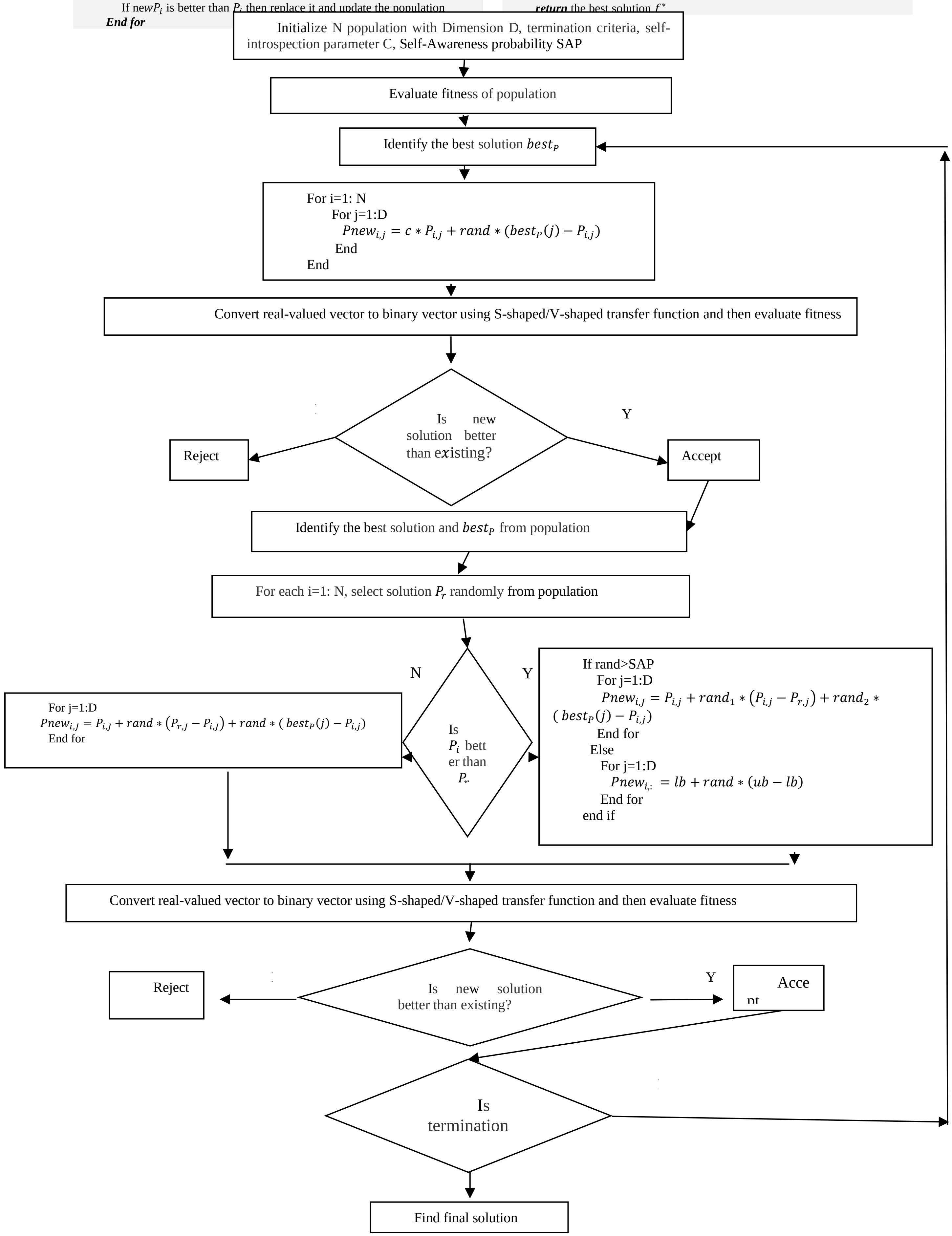


Figure 3: Framework of bmSGO algorithm

3 Simulation and experimental results

The performance of the bmSGO algorithm is demonstrated in this paper through three experiments. In experiment 1, proposed approaches are compared with each other. In the second experiment, the proposed FS approaches, their performances are compared with various state-of-the-art approaches such as PSO [51], CS [52], HS [53], BA [54], TLBO [55], GWO [26]. In the third experiment, proposed FS approaches compared with the newest approaches: CSA (Crow Search Algorithm) [56], GOA (Grasshopper Optimization Algorithm) [57], MVO (Multi-Verse Optimizer) [58], SSA (Salp Swarm Algorithm) [59], Sine Cosine Algorithm (SCA)[60]. and WOA (Whale Optimization Algorithm) [29].

3.2 Parameter settings and softwares

Here are the parameter settings for all algorithms, as detailed in Table 1. The implementations were done using MATLAB 2016a on a laptop running the Microsoft Windows 10 operating system, equipped with an Intel Core i5 processor and 8 GB of memory.

Table 1: Parameter setting for an experiment

Sl. No.	Parameter	Value(s)
1	K for validation	5
2	Population size	10
3	Maximum number fitness function evaluation	500
4	The dimension of the problem	No. of features in the dataset
5	Search domain	{0, 1}
6	Parameter P_a in CS	0.25
7	Q min Frequency minimum in BA	0
8	Q max Frequency maximum in BA	2
9	Loudness in BA	0.5
10	Pulse rate in BA	0.5
11	Acceleration constants in PSO	[2, 2]
12	Inertia w in PSO	[0.9, 0.6]
13	A parameter in GWO	Min=0 and max=2
14	Awareness probability (AP) in CSA	0.1
15	Flight length (FL) In CSA	2
16	For finding c in GOA	cmax = 1, cmin = 0.00004 for finding value of c= cmax-l*((cmax- cmin)/Max_iter, Max_iter = 50.
17	*r4* parameter in SCA	0.5
18	A parameter in SCA	2
19	A parameter in WOA	Min=0 and max=2
20	Parameter c in SSA	c = $2*e^{-(\frac{4}{L})^2}$, where L = max_iteration = 50.
21	WEP parameter in MVO	WEP is increased linearly from 0.2 to 1, and
22	TDR parameter in MVO	TDR is decreased from 0.6 to 0
23	C parameter in MSGO/bmSGO	0.2
24	'*SAP'* parameter in MSGO/bmSGO	0.7
25	hmcr parameter in HS	0.9
26	Par parameter in HS	0.3
27	bw parameter in HS	0.01

3.3 Fitness function for binary optimization algorithms for FS

In the feature selection (FS) problem, the dimension of the solution vector corresponds to the number of features in the dataset. Each element in the solution vector is either 1 or 0, where 1 signifies that the corresponding feature is selected, and 0 signifies that the feature is not selected.

The FS problem is treated as a multi-objective optimization problem with two conflicting objectives: (a) achieving the highest classification accuracy (CA), which is a maximization objective, and (b) selecting the fewest number of features, a minimization objective. To reconcile this contradiction, we introduce the classification error rate. Equation 7 combines these two objectives, converting the FS problem into a single objective problem:

$$\text{Fitness}=\alpha\gamma_R(SF)+\beta\frac{|SF|}{|TF|} \quad \text{.........................} \quad (7)$$

Here, SF represents the selected feature subset, $\gamma_R(SF)$ is the classification error rate of SF, |SF| denotes the cardinality of the selected feature subset, |TF| represents the total number of features in the original dataset, and α and β are parameters corresponding to classification quality and subset length, where α ∈ [0, 1] and α + β = 1 [30]. In our experiment, we set β = 0.01 according to [48].

1.2 Description of datasets used in the experiments

To evaluate the performance of the proposed binary approaches, we selected twenty-three benchmark datasets from the UCI data repository for our experiments. Table 2 provides details about these selected datasets, including the number of features, instances, and classes in each dataset. We included both small and high-dimensional datasets in our experiments to ensure comprehensive evaluation.

Table 2: List of datasets used in the experiments

Sl. No.	Name	No. of features	No. of instances	No. of classes
Dataset1	Breastcancer	9	699	2
Dataset 2	BreastEW	30	569	2
Dataset3	Clean1	166	476	
Dataset 4	Clean2	166	6598	
Dataset 5	CongressEW	16	435	2

Dataset 6	Exactly1	13	1000	2
Dataset 7	Exactly2	13	1000	2
Dataset 8	HeartEW	13	270	2
Dataset 9	IonosphereEW	34	351	2
Dataset 10	KrvskpEW	36	3196	2
Dataset 11	Lymphography	18	148	2
Dataset 12	M of n	13	1000	2
Dataset 13	PenglungEW	325	73	2
Dataset 14	Semeion	256	1593	2
Dataset 15	Sonar	60	208	2
Dataset 16	Spect	22	267	2
Dataset 17	Tic-tac-toe	9	958	2
Dataset 18	Votes	16	300	2
Dataset 19	WaveformEW	40	5000	3
Dataset 20	WineEW	13	178	3
Dataset 21	Zoo	16	101	6
Dataset 22	Vechile	18	846	4
Dataset 23	Dermatology	34	366	6

In this study, we utilized a K-nearest neighbors (KNN) classifier with the Euclidean distance metric to assess the classification accuracy (CA) of the selected feature subset obtained through our proposed FS method applied to the entire original dataset. We consistently used the best choice of K, which is K=5, across all datasets [49]. To facilitate evaluation, each dataset was divided in a cross-validation manner [49]. Typically, in K-fold cross-validation, K-1 folds are allocated for training and validation, while the remaining fold is reserved for testing purposes.

3.4 Evaluation criteria

Each dataset is randomly divided into three equal portions: validation, training, and testing datasets. To ensure stability and statistical significance, the algorithm is repeated 20 times. Statistical results, including average classification accuracy, average selection size, mean value, best value, worst value, and standard deviation of fitness solutions, are determined and reported in tables. The best results for each algorithm are highlighted in bold. A Wilcoxon Rank-Sum (WRS) test is conducted at a significance level of 0.05 on fitness solutions. The WRS test is a nonparametric statistical test used to determine whether the results of the proposed approaches are statistically different from those of other algorithms [50]. This statistical test yields a p-value, which is used to assess the significance level between the two algorithms.

3.5 Experiment 1: The performance comparison of MSGO, S-bmSGO, and V-bmSGO

In this experiment, we compared the performance of the proposed approaches with each other. Table 3 presents the results of the proposed approaches in terms of CA and Fig 4 provides chart result on that. Notably, the V-bmSGO algorithm exhibited superior performance compared to the original MSGO for CA. Across all datasets used in this experiment, except for tic-tac-toe and exactly2 datasets, V-bmSGO outperformed the original MSGO. In the exactly2 dataset, both the original MSGO and V-bmSGO achieved comparable performance. Additionally, S-bmSGO performed better than the original MSGO on all datasets except for the ionosphere dataset. Moreover, V-bmSGO outperformed S-bmSGO in seventeen out of twenty-three datasets, indicating its superior performance in most cases.

Regarding average selection size, V-bmSGO outperformed S-bmSGO on all datasets and was competitive with the original MSGO, as shown in Table 4 and Fig 5. The original MSGO outperformed V-bmSGO on four datasets and was equivalent on one dataset. Notably, in the breast cancer dataset, MSGO provided a 4.20 average selection size compared to V-bmSGO's 4.5 average selection size. Similarly, in the congressEW dataset, MSGO outperformed V-bmSGO with a 2.20 average selection size compared to 2.65. In the exactly1 dataset, MSGO achieved a 3.4 average selection size compared to V-bmSGO's 5.0, and in the vote dataset, MSGO's average selection size was 3 compared to V-bmSGO's 3.2.

Table 5 presents the results of the proposed approaches regarding the statistical mean fitness measure. Here, V-bmSGO outperformed the original MSGO for mean fitness measure except in the exactly2 dataset, where both performed equally. S-bmSGO also performed better than the original MSGO on all datasets except for the ionosphere and exactly2 datasets.

Moving on to the statistical best fitness measure in Table 6, we observe that V-bmSGO performed better or equivalently in most cases compared to other techniques, except for the exactly2 dataset. Similarly, Table 7 highlights that V-bmSGO exhibited superior performance for the statistical worse fitness measure, except in the exactly2 dataset where both V-bmSGO and MSGO performed equally.

In terms of the statistical standard deviation fitness measure, Table 8 shows that S-bmSGO outperformed both the original MSGO and V-bmSGO in most cases, indicating better stability and consistency.

All the best results are boldfaced in Tables 3-8. Fig 6 displays the convergence curve for all compared approaches, depicting the fitness function value in each iteration with 50 iterations and a population size of 10.

Table 9 reports the p-values of the WRS test conducted at a 5% significance level for V-bmSGO vs. S-bmSGO and MSGO. A p-value less than 0.05 indicates a significant difference at a 5% level. Values greater than or equal to 0.05 are boldfaced, indicating no significant difference. NaN indicates results that are equivalent and incomparable. From the table, we observe that in five cases for S-bmSGO and one case for MSGO out of twenty-three cases, p-values are greater than or equal to 0.05, suggesting no significant difference. Only one case shows NaN value for MSGO.

22	Dataset 22	4.7500	9.1000	**3.1000**
23	Dataset 23	19.500	21.450	**14.150**

Table 3: Results of proposed approaches based on mean CA

S. No	Datasets	MSGO	S-bmSGO	V-bmSGO
1	Dataset1	0.9639	**0.9686**	0.9664
2	Dataset 2	0.9595	**0.9672**	0.9668
3	Dataset3	0.8626	0.8769	**0.9046**
4	Dataset 4	0.9606	0.9640	**0.9657**
5	Dataset 5	0.9589	0.9647	**0.9663**
6	Dataset 6	0.7168	0.9198	**0.9400**
7	Dataset 7	0.7760	**0.7762**	0.7760
8	Dataset 8	0.8352	**0.8637**	0.8593
9	Dataset 9	0.8795	0.8764	**0.9170**
10	Dataset 10	0.9453	0.9605	**0.9699**
11	Dataset 11	0.8541	0.8912	**0.8926**
12	Dataset 12	0.8773	0.9745	**0.9958**
13	Dataset 13	0.8497	0.8755	**0.9236**
14	Dataset 14	0.9744	0.9784	**0.9837**
15	Dataset 15	0.8375	0.8620	**0.8841**
16	Dataset 16	0.8657	0.8843	**0.8877**
17	Dataset 17	0.8220	**0.8246**	0.8209
18	Dataset 18	0.9480	0.9550	**0.9733**
19	Dataset 19	0.7733	0.7912	**0.8010**
20	Dataset 20	0.9876	**0.9978**	0.9955
21	Dataset 21	0.9747	0.9990	**1**
22	Dataset 22	0.6840	0.7117	**0.7234**
23	Dataset 23	0.9732	0.9847	**0.9888**
	Average	0.8817	0.9073	**0.9175**

Table 4: Results of proposed approaches based on average NSF

S. No	Datasets	MSGO	S-bmSGO	V-bmSGO
1	Dataset1	**4.20**	6	4.5
2	Dataset 2	15.25	17.30	**12.10**
3	Dataset3	65.60	87.75	**35.35**
4	Dataset 4	73.700	87.20	**43.25**
5	Dataset 5	**2.2000**	7.3000	2.6500
6	Dataset 6	**3.4000**	6.9000	5
7	Dataset 7	**1**	1.4500	**1**
8	Dataset 8	8.7000	9	**7.1500**
9	Dataset 9	3.6500	16.20	**3.4000**
10	Dataset 10	23.50	20.10	**12.90**
11	Dataset 11	7.7500	9.4500	**6**
12	Dataset 12	10.200	6.9000	**6.1500**
13	Dataset 13	75.400	163.70	**37.15**
14	Dataset 14	136.40	148.95	**91.85**
15	Dataset 15	21	28.25	**12.850**
16	Dataset 16	10.10	10.90	**6.3500**
17	Dataset 17	8.8000	9	**8.60**
18	Dataset 18	**3**	6.4000	3.2000
19	Dataset 19	25	25	**16.15**
20	Dataset 20	10.650	7.6000	**6.5000**
21	Dataset 21	3.6000	6.8500	**2**

Table 5: Results of proposed approaches based on mean FM

S. No	Datasets	MSGO	S-bmSGO	V-bmSGO
1	Dataset1	0.0404	**0.0378**	0.0383
2	Dataset 2	0.0452	0.0382	**0.0369**
3	Dataset3	0.1400	0.1272	**0.0966**
4	Dataset 4	0.0435	0.0409	**0.0366**
5	Dataset 5	0.0420	0.0395	**0.0350**
6	Dataset 6	0.2830	0.0847	**0.0632**
7	Dataset 7	**0.2225**	0.2227	**0.2225**
8	Dataset 8	0.1699	**0.1419**	0.1448
9	Dataset 9	0.1203	0.1271	**0.0831**
10	Dataset 10	0.0607	0.0447	**0.0334**
11	Dataset 11	0.1488	0.1129	**0.1097**
12	Dataset 12	0.1293	0.0306	**0.0089**
13	Dataset 13	0.1511	0.1283	**0.0768**
14	Dataset 14	0.0305	0.0270	**0.0242**
15	Dataset 15	0.1644	0.1413	**0.1168**
16	Dataset 16	0.1376	0.1195	**0.1141**
17	Dataset 17	0.1860	**0.1836**	0.1869
18	Dataset 18	0.0534	0.0485	**0.0442**
19	Dataset 19	0.2307	0.2130	**0.2010**
20	Dataset 20	0.0204	**0.0081**	0.0094
21	Dataset 21	0.0273	0.0053	**0.0013**
22	Dataset 22	0.3154	0.2905	**0.2756**
23	Dataset 23	0.0322	0.0193	**0.0174**

Table 6: Results of proposed approaches based on the best FM

S. No	Datasets	MSGO	S-bmSGO	V-bmSGO
1	Dataset1	**0.0378**	**0.0378**	**0.0378**
2	Dataset2	0.0394	0.0331	**0.0321**
3	Dataset3	0.1171	0.1042	**0.0641**
4	Dataset 4	0.0400	0.0386	**0.0336**
5	Dataset 5	0.0355	0.0322	**0.0291**
6	Dataset 6	0.0707	**0.0046**	**0.0046**
7	Dataset 7	0.2225	**0.2216**	0.2225
8	Dataset 8	0.1447	**0.1374**	**0.1374**
9	Dataset 9	0.0962	0.1154	**0.0524**
10	Dataset 10	0.0382	0.0388	**0.0267**
11	Dataset 11	0.1120	0.0981	**0.0970**
12	Dataset 12	0.0735	**0.0046**	**0.0046**
13	Dataset 13	0.0833	0.0855	**0.0300**
14	Dataset 14	0.0265	0.0247	**0.0187**
15	Dataset 15	0.1283	0.1196	**0.0685**
16	Dataset 16	0.1093	0.1080	**0.0997**
17	Dataset 17	**0.1836**	**0.1836**	**0.1836**
18	Dataset 18	0.0421	0.0427	**0.0295**
19	Dataset 19	0.2144	0.2086	**0.1885**
20	Dataset 20	0.0157	**0.0046**	**0.0046**
21	Dataset 21	**0.0013**	0.0019	**0.0013**
22	Dataset 22	**0.2544**	0.2755	**0.2544**

23	Dataset 23	0.0164	0.0146	**0.0116**

Table 7: Results of proposed approaches based on worse FM

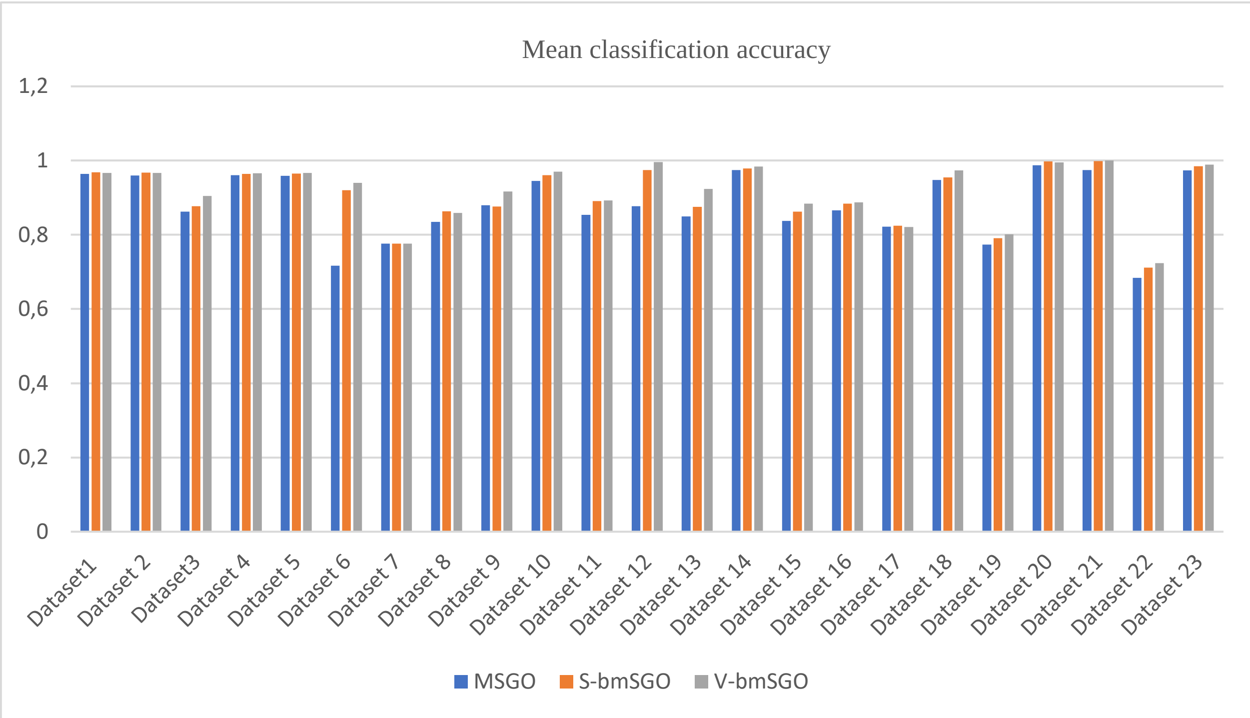
S. No	Datasets	MSGO	S-bmSGO	V-bmSGO
1	Dataset 1	0.0445	**0.0378**	0.0395
2	Dataset 2	0.0515	**0.0417**	0.0421
3	Dataset 3	0.1672	0.1391	**0.1113**
4	Dataset 4	0.0464	0.0425	**0.0400**
5	Dataset 5	0.0479	0.0471	**0.0415**
6	Dataset 6	0.2978	**0.2141**	0.2978
7	Dataset 7	**0.2225**	0.2233	**0.2225**
8	Dataset 8	0.1976	**0.1536**	0.1586
9	Dataset 9	0.1415	0.1350	**0.1078**
10	Dataset 10	0.0781	0.0518	**0.0487**
11	Dataset 11	0.1761	0.1388	**0.1377**
12	Dataset 12	0.1635	0.0675	**0.0331**
13	Dataset 13	0.1883	0.1652	**0.1084**
14	Dataset 14	0.0342	0.0290	**0.0276**
15	Dataset 15	0.1944	0.1571	**0.1443**
16	Dataset 16	0.1588	**0.1311**	0.1348
17	Dataset 17	0.2308	**0.1836**	0.2205
18	Dataset 18	0.0600	**0.0512**	0.0600
19	Dataset 19	0.2476	0.2177	**0.2095**
20	Dataset 20	0.0276	0.0165	**0.0150**
21	Dataset 21	0.0833	0.0250	**0.0013**
22	Dataset 22	0.3830	0.3415	**0.3171**
23	Dataset 23	0.0543	0.0258	**0.0215**

Table 8: Results of proposed approaches based on standard deviation FM

S. No	Datasets	MSGO	S-bmSGO	V-bmSGO
1	Dataset1	0.0020	**1.4238e-17**	3.8990e-04
2	Dataset 2	0.0036	**0.0022**	0.0026
3	Dataset3	0.0142	**0.0089**	0.0124
4	Dataset 4	0.0016	**0.0011**	0.0016
5	Dataset 5	**0.0027**	0.0042	0.0040
6	Dataset 6	**0.0505**	0.0656	0.1203
7	Dataset 7	**2.8477e-17**	4.1672e-04	**2.8477e-17**
8	Dataset 8	0.0141	**0.0043**	0.0086
9	Dataset 9	0.0137	**0.0053**	0.0120
10	Dataset 10	0.0122	**0.0029**	0.0061
11	Dataset 11	0.0172	**0.0125**	0.0138
12	Dataset 12	0.0258	0.0206	**0.0104**
13	Dataset 13	0.0293	**0.0180**	0.0184
14	Dataset 14	0.0020	**0.0011**	0.0020
15	Dataset 15	0.0154	**0.0101**	0.0192
16	Dataset 16	0.0116	**0.0060**	0.0079
17	Dataset 17	0.0106	**0**	0.0102
18	Dataset 18	0.0065	**0.0030**	0.0092
19	Dataset 19	0.0093	**0.0025**	0.0061
20	Dataset 20	**0.0028**	0.0041	0.0043
21	Dataset 21	0.0282	0.0048	**2.2247e-19**
22	Dataset 22	0.0369	0.0160	**0.0136**
23	Dataset 23	0.01 10	0.0030	**0.0020**

Table 9. p-values of the WRS test of the proposed V-bmSGO vs. S-bmSGO and MSGO (p ≥ 0.05 are boldfaced)

Sl. No	Datasets	S-bmSGO	MSGO
1	Dataset1	1.6859e-06	4.0221e-04
2	Dataset 2	**6.3700e-02**	1.8901e-07
3	Dataset3	1.4289e-07	6.7860e-08
4	Dataset 4	1.4309e-07	6.7956e-08
5	Dataset 5	3.6000e-03	3.7911e-06
6	Dataset 6	1.9900e-02	2.1869e-05
7	Dataset 7	**8.4700e-02**	**NaN**
8	Dataset 8	**7.2100e-01**	1.0446e-06
9	Dataset 9	6.3490e-08	1.2422e-07
10	Dataset 10	8.5641e-06	2.5498e-07
11	Dataset 11	4.7500e-02	3.6067e-07
12	Dataset 12	3.5135e-04	1.7721e-08
13	Dataset 13	1.0631e-07	1.0617e-07
14	Dataset 14	1.1024e-05	1.0631e-07
15	Dataset 15	1.2470e-05	1.6483e-07
16	Dataset 16	2.4000e-03	1.3451e-06
17	Dataset 17	**1.6260e-01**	**6.1470e-01**
18	Dataset 18	6.8000e-03	9.8490e-04
19	Dataset 19	7.8870e-08	6.7956e-08
20	Dataset 20	**4.9490e-01**	4.5581e-08
21	Dataset 21	7.4931e-09	9.3043e-06
22	Dataset 22	7.4664e-06	7.5345e-05
23	Dataset 23	1.2080e-04	1.5692e-05


Mean classification accuracy
1,2
1
0,8
0,6
0,4
0,2
0
Dataset1
Dataset 2
Dataset3
Dataset 4
Dataset 5
Dataset 6
Dataset 7
Dataset 8
Dataset 9
Dataset 10
Dataset 11
Dataset 12
Dataset 13
Dataset 14
Dataset 15
Dataset 16
Dataset 17
Dataset 18
Dataset 19
Dataset 20
Dataset 21
Dataset 22
Dataset 23
MSGO
S-bmSGO
V-bmSGO

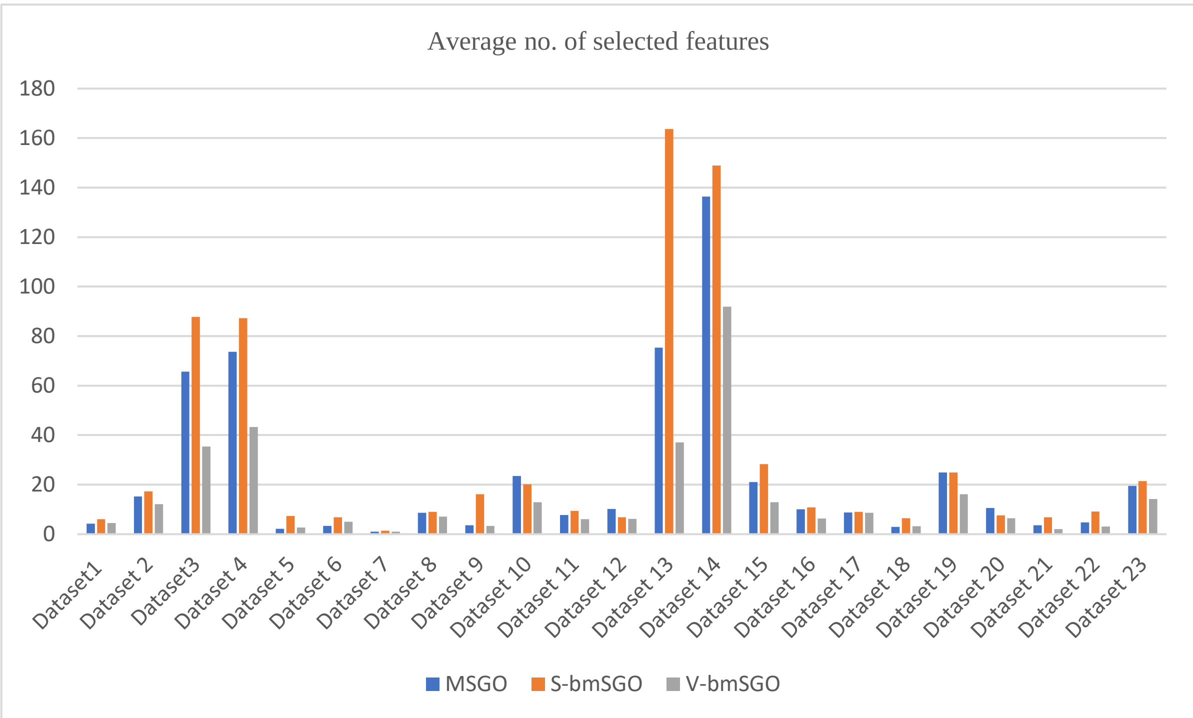
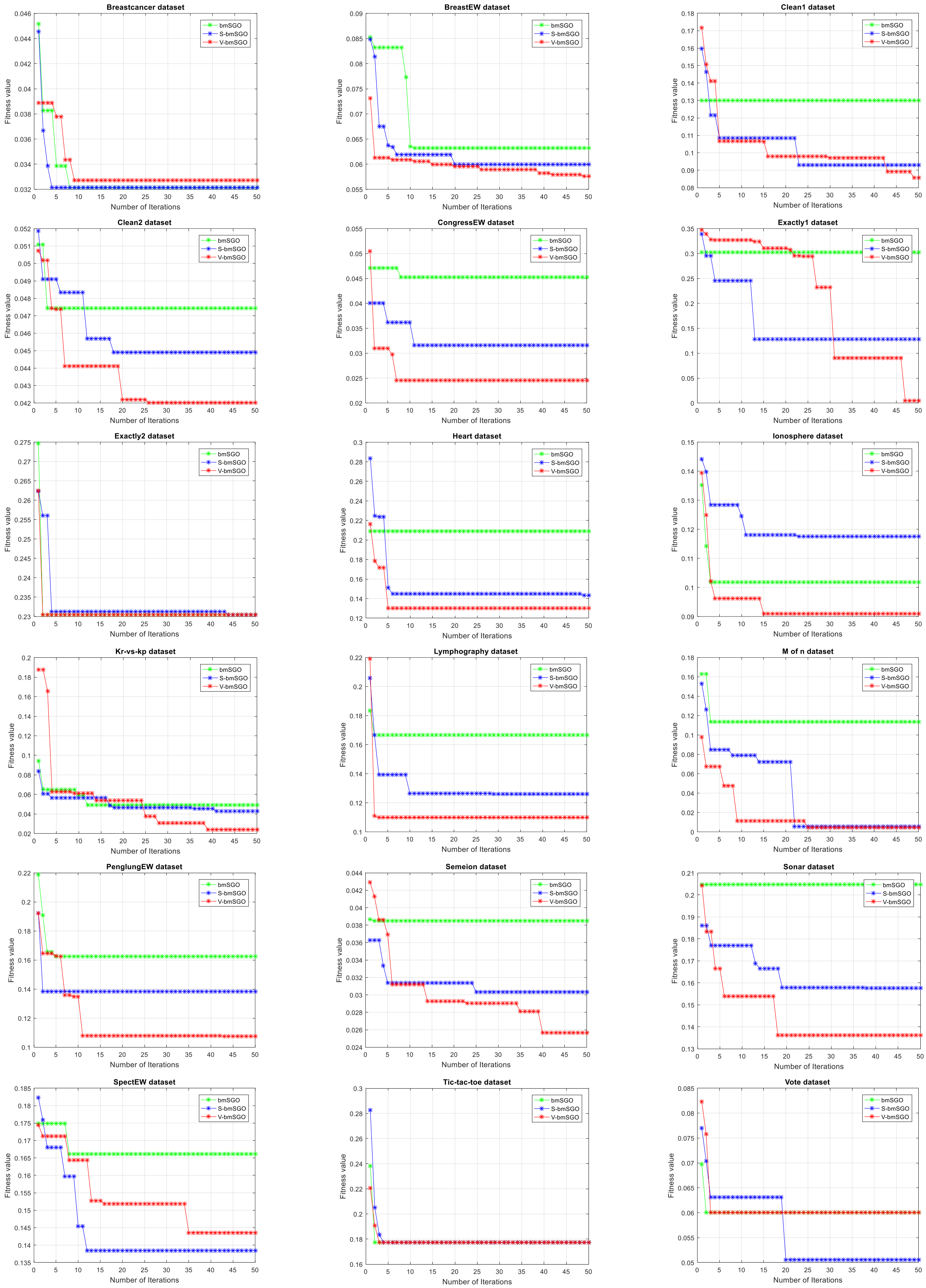

Average no. of selected features
180
160
140
120
100
80
60
40
20
0
Dataset1
Dataset 2
Dataset3
Dataset 4
Dataset 5
Dataset 6
Dataset 7
Dataset 8
Dataset 9
Dataset 10
Dataset 11
Dataset 12
Dataset 13
Dataset 14
Dataset 15
Dataset 16
Dataset 17
Dataset 18
Dataset 19
Dataset 20
Dataset 21
Dataset 22
Dataset 23
MSGO
S-bmSGO
V-bmSGO

Figure 5: Chart on mean number of selected features obtained by MSGO, S-bmSGO, V-bmSGO



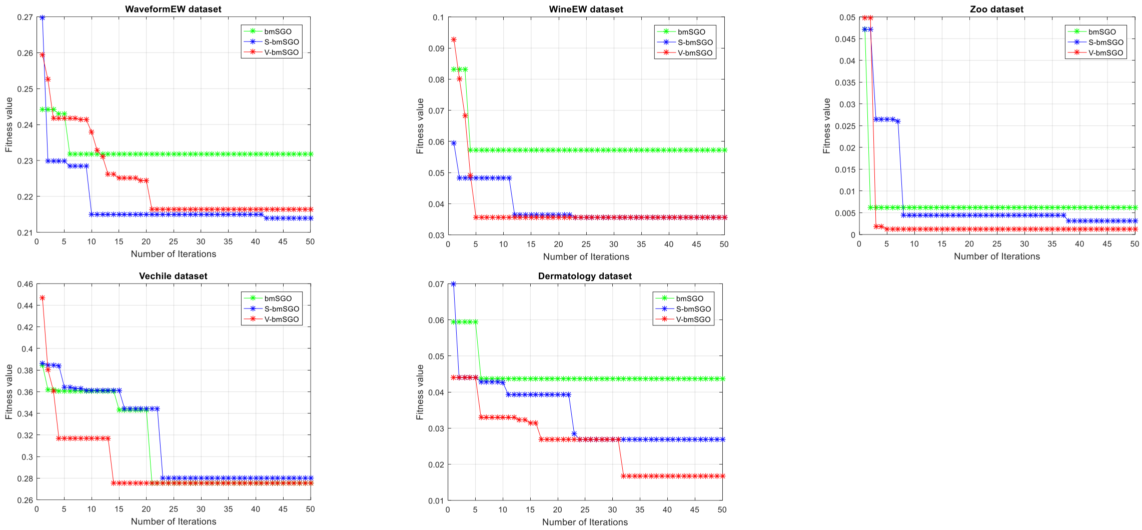


Figure 6: Convergence curve for all compared approaches for 23 UCI datasets

3.6 Experiment 2: The performance comparison with the state-of-the-art approaches

From the first experiment, it's evident that V-bmSGO displayed superior performance compared to other proposed methods in terms of CA and average selection size. In this study, we compared the performance of the best-performing method, V-bmSGO, with several state-of-the-art approaches commonly used for FS problem-solving.

Table 10 presents the CA results of V-bmSGO alongside CS, GWO, HS, BA, TLBO, and PSO and Fig 7 displays the chart on that. Across all datasets, V-bmSGO outperformed PSO and BA consistently. Additionally, it surpassed TLBO and GWO on all datasets except for tic-tac-toe. While both V-bmSGO and CS performed equally on the exactly2 dataset, V-bmSGO outperformed CS on all other datasets. Furthermore, V-bmSGO exhibited better performance than HS on fourteen out of twenty-three datasets, indicating the robustness and effectiveness of the proposed approach. In terms of ranking, V-bmSGO secured the top position, followed by HS in second place and CS in third place.

Table 11 displays the average number of selected features using V-bmSGO and other methods and Fig 8 provides chart on that. V-bmSGO exhibits significantly better performance across all datasets except for breast cancer and tic-tac-toe datasets, where the PSO algorithm shows superior performance. This superiority of V-bmSGO can be attributed to its enhanced capability in exploring and exploiting the feature space effectively, leading to the discovery of high-performance regions.

The statistical measures including mean, best, worst, and standard deviation obtained from multiple runs of the algorithms on all datasets are detailed in Tables 12-15. Specifically, Table 12 reveals that V-bmSGO surpasses CS, PSO, and BA in terms of the mean statistical measure across all datasets. Moreover, V-bmSGO outperforms GWO and TLBO on all datasets except for the tic-tac-toe dataset, and it also outperforms HS in thirteen datasets.

Table 13 presents the statistical best fitness measure across datasets. Notably, V-bmSGO outperforms PSO and BA in this measure across all datasets. It also outperforms GWO in all datasets except for the tic-tac-toe dataset, where both achieve equivalent results. Moreover, V-bmSGO outperforms CS in fourteen datasets and performs equally with seven datasets. It also surpasses TLBO in eighteen datasets and performs equally with five datasets. In comparison with HS, V-bmSGO outperforms it in twelve datasets and performs equally in ten datasets, with HS outperforming V-bmSGO in one dataset out of twenty-three.

Moving on to Table 14, which reports the statistical worst fitness measure, V-bmSGO again outperforms CS, PSO, and BA across all datasets. It also surpasses GWO and TLBO on all datasets except the tic-tac-toe dataset. Additionally, V-bmSGO outperforms HS on fourteen datasets, whereas HS outperforms V-bmSGO on nine datasets out of twenty-three.

Table 15 focuses on the statistical standard deviation fitness measure across datasets. Here, V-bmSGO outperforms GWO in six datasets and TLBO in eleven datasets. Conversely, GWO outperforms V-bmSGO in four datasets, and TLBO outperforms V-bmSGO in four datasets.

Lastly, Table 16 presents the p-values of the WRS test at a 5% significance level, comparing V-bmSGO with other state-of-the-art approaches. The p-values less than 0.05 indicate a significant difference at this level. Notably, V-bmSGO shows p-values greater than or equal to 0.05 for comparisons with CS in one case, GWO in one case, HS in ten cases, and TLBO in one case out of twenty-three cases. For PSO and BA, all comparisons have p-values less than 0.05.

Overall, V-bmSGO exhibits strong performance compared to state-of-the-art approaches across various statistical measures and datasets.

Table 10. Results of V-bmSGO and all other approaches based on average CA

Sl. No	Datasets	CS	GWO	HS	PSO	BA	TLBO	V-bmSGO
1	Dataset1	0.9671	0.9613	**0.9686**	0.9621	0.9611	0.9633	0.9664
2	Dataset 2	0.9656	0.9568	**0.9689**	0.9509	0.9528	0.9619	0.9668
3	Dataset3	0.8870	0.8437	0.8962	0.8313	0.8275	0.8571	**0.9046**
4	Dataset 4	0.9643	0.9584	**0.9662**	0.9574	0.9579	0.9604	0.9657
5	Dataset 5	0.9638	0.9511	**0.9670**	0.9454	0.9468	0.9550	0.9663
6	Dataset 6	0.8522	0.7111	**0.9657**	0.7027	0.6960	0.7439	0.9400
7	Dataset 7	**0.7760**	0.7643	0.7737	0.7663	0.7572	0.7688	**0.7760**
8	Dataset 8	0.8541	0.8204	**0.8644**	0.8148	0.8067	0.8330	0.8593
9	Dataset 9	0.8776	0.8483	0.8827	0.8460	0.8494	0.8568	**0.9170**
10	Dataset 10	0.9587	0.9376	0.9679	0.9117	0.9112	0.9359	**0.9699**
11	Dataset 11	0.8791	0.8189	0.8917	0.8074	0.8128	0.8378	**0.8926**
12	Dataset 12	0.9532	0.8718	0.9897	0.8324	0.8326	0.8873	**0.9958**
13	Dataset 13	0.8755	0.8149	0.8919	0.8044	0.7989	0.8374	**0.9236**
14	Dataset 14	0.9779	0.9737	0.9805	0.9716	0.9701	0.9763	**0.9837**
15	Dataset 15	0.8611	0.8139	0.8731	0.8207	0.8111	0.8260	**0.8841**
16	Dataset 16	0.8813	0.8549	0.8870	0.8470	0.8496	0.8619	**0.8877**
17	Dataset 17	0.8071	**0.8246**	**0.8246**	0.7693	0.7630	**0.8246**	0.8209
18	Dataset 18	0.9560	0.9323	0.9650	0.9283	0.9347	0.9383	**0.9733**
19	Dataset 19	0.7915	0.7733	**0.8014**	0.7545	0.7488	0.7775	0.8010
20	Dataset 20	0.9910	0.9888	0.9910	0.9674	0.9708	0.9899	**0.9955**
21	Dataset 21	0.9777	0.9346	0.9960	0.9291	0.9336	0.9449	**1**
22	Dataset 22	0.6872	0.6043	0.7234	0.6096	0.6043	0.6457	**0.7234**
23	Dataset 23	0.9850	0.9724	**0.9913**	0.9552	0.9481	0.9765	0.9888
Average		0.8996	0.8666	0.9143	0.8559	0.8541	0.8765	**0.9175**

Table 11: Results of V-bmSGO and all other approaches based on average NSF

Sl. No	Datasets	CS	GWO	HS	PSO	BA	TLBO	V-bmSGO
1	Dataset1	5.3000	4.8000	6	**4.2000**	4.6000	8.1500	4.5
2	Dataset 2	15.85	20.35	18.30	15.90	14.95	27.05	**12.10**
3	Dataset3	79.25	92.55	116.55	81.25	82.45	158.20	**35.35**
4	Dataset 4	81.80	88.90	105.40	82.450	84.65	148.15	**43.25**
5	Dataset 5	5.7500	12.500	7.9000	7.9500	8.5000	15.45	**2.6500**
6	Dataset 6	7.5500	10.10	6.3500	7	6.5500	12.35	**5**
7	Dataset 7	1.4500	5.7000	3.5500	4.3500	4.9500	11.55	**1**
8	Dataset 8	8.5500	10.80	8.6500	7.6500	6.8500	12.70	**7.1500**
9	Dataset 9	13.95	17.20	15.35	16	16.40	31.05	**3.4000**
10	Dataset 10	18.95	31.35	20.10	19.300	20	35	**12.90**
11	Dataset 11	7.4500	8.7500	8.8500	8.2000	8.2500	16	**6**
12	Dataset 12	7.3000	11.450	6.4500	7.9000	8.0500	12.65	**6.1500**
13	Dataset 13	153.75	225.35	209.45	164.15	159.15	305.30	**37.15**
14	Dataset 14	127.45	175.15	183.55	132.95	132.40	260.95	**91.85**
15	Dataset 15	28.20	31.10	34.15	29.40	30.75	55.45	**12.850**
16	Dataset 16	8.9500	11.20	11.80	10.60	10.80	15	**6.3500**
17	Dataset 17	7.0500	9	9	**5.9000**	5.9500	9	8.60
18	Dataset 18	5.8000	8.1000	5.0500	7.1000	7.2500	14.20	**3.2000**
19	Dataset 19	22.75	36.40	26.45	21.05	21.65	39.40	**16.15**
20	Dataset 20	7.1000	11.80	7.6500	7.2000	7.4000	12.35	**6.5000**
21	Dataset 21	6.4500	8.7500	5.5000	8.3500	8.3000	14.150	**2**
22	Dataset 22	7.2500	9.9000	8.8500	9.3000	9.3000	15.70	**3.1000**
23	Dataset 23	19.800	26.900	22.750	18.500	17.600	32.600	**14.150**

Table 12: Results of V-bmSGO and all other approaches based on mean FM

Sl. No	Datasets	CS	GWO	HS	PSO	BA	TLBO	V-bmSGO
1	Dataset1	0.0384	0.0437	**0.0378**	0.0421	0.0436	0.0420	0.0383
2	Dataset 2	0.0393	0.0495	**0.0368**	0.0539	0.0517	0.0442	0.0369
3	Dataset3	0.1167	0.1603	0.1098	0.1719	0.1757	0.1487	**0.0966**
4	Dataset 4	0.0402	0.0466	0.0398	0.0471	0.0467	0.0446	**0.0366**
5	Dataset 5	0.0395	0.0562	0.0376	0.0590	0.0580	0.0515	**0.0350**
6	Dataset 6	0.1521	0.2938	**0.0388**	0.2997	0.3060	0.2608	0.0632
7	Dataset 7	0.2229	0.2377	0.2268	0.2347	0.2442	0.2337	**0.2225**
8	Dataset 8	0.1510	0.1861	**0.1409**	0.1892	0.1967	0.1731	0.1448
9	Dataset 9	0.1253	0.1552	0.1207	0.1571	0.1539	0.1472	**0.0831**
10	Dataset 10	0.0461	0.0705	0.0373	0.0927	0.0935	0.0717	**0.0334**
11	Dataset 11	0.1239	0.1841	**0.1012**	0.1527	0.1899	0.1660	0.1097
12	Dataset 12	0.0519	0.1357	0.0152	0.1720	0.1719	0.1198	**0.0089**
13	Dataset 13	0.1280	0.1902	0.1135	0.1987	0.2039	0.1672	**0.0768**
14	Dataset 14	0.0267	0.0326	0.0262	0.0332	0.0346	0.0304	**0.0242**
15	Dataset 15	0.1423	0.1894	0.1313	0.1824	0.1922	0.1776	**0.1168**
16	Dataset 16	0.1215	0.1488	**0.1121**	0.1563	0.1538	0.1427	0.1141
17	Dataset 17	0.1988	**0.1836**	**0.1836**	0.2349	0.2412	**0.1836**	0.1869
18	Dataset 18	0.0472	0.0721	**0.0378**	0.0754	0.0692	0.0659	0.0442
19	Dataset 19	0.2121	0.2336	0.2033	0.2483	0.2541	0.2292	**0.2010**
20	Dataset 20	0.0144	0.0202	**0.0076**	0.0378	0.0346	0.0182	0.0094
21	Dataset 21	0.0261	0.0702	0.0074	0.0754	0.0709	0.0604	**0.0013**
22	Dataset 22	0.3137	0.3973	0.2787	0.3917	0.3970	0.3562	**0.2756**
23	Dataset 23	0.0207	0.0352	0.0253	0.0498	0.0566	0.0307	**0.0174**

Table 13 Results of V-bmSGO and all other approaches based on best FM

S. No	Datasets	CS	GWO	HS	PSO	BA	TLBO	V-bmSGO
1	Dataset1	**0.0378**	0.0384	**0.0378**	0.0384	0.0384	**0.0378**	**0.0378**
2	Dataset 2	0.0328	0.0366	**0.0300**	0.0432	0.0422	0.0331	0.0321
3	Dataset3	0.0830	0.1254	0.0980	0.1298	0.1544	0.1295	**0.0641**
4	Dataset 4	0.0369	0.0412	0.0362	0.0434	0.0436	0.0431	**0.0336**
5	Dataset 5	**0.0291**	0.0471	0.0310	0.0415	0.0485	0.0388	**0.0291**
6	Dataset 6	**0.0046**	0.2639	**0.0046**	0.2287	0.2710	**0.0046**	**0.0046**
7	Dataset 7	**0.2225**	0.2233	**0.2225**	0.2233	0.2233	**0.2225**	**0.2225**
8	Dataset 8	**0.1374**	0.1617	**0.1374**	0.1586	**0.1374**	0.1470	**0.1374**
9	Dataset 9	0.1068	0.1394	0.0870	0.1347	0.1273	0.1376	**0.0524**
10	Dataset 10	0.0302	0.0503	0.0275	0.0526	0.0648	0.0503	**0.0267**
11	Dataset 11	0.0975	0.1265	0.0853	0.1527	0.1505	0.1120	**0.0970**
12	Dataset 12	**0.0046**	0.1020	**0.0046**	0.0331	0.0940	0.0814	**0.0046**
13	Dataset 13	0.0842	0.1650	0.0864	0.1120	0.1389	0.1134	**0.0300**
14	Dataset 14	0.0208	0.0296	0.0241	0.0286	0.0298	0.0271	**0.0187**
15	Dataset 15	0.0995	0.1483	0.1002	0.1475	0.1568	0.1570	**0.0685**
16	Dataset 16	0.1075	0.1149	0.1001	0.1237	0.1301	0.1311	**0.0997**
17	Dataset 17	**0.1836**	**0.1836**	**0.1836**	0.2143	0.2051	**0.1836**	**0.1836**
18	Dataset 18	0.0361	0.0512	**0.0295**	0.0547	0.0572	0.0518	**0.0295**
19	Dataset 19	0.1992	0.2230	0.1960	0.2048	0.2302	0.2139	**0.1885**
20	Dataset 20	0.0062	0.0173	**0.0046**	0.0165	0.0150	0.0069	**0.0046**
21	Dataset 21	0.0025	0.0044	**0.0013**	0.0475	0.0259	0.0050	**0.0013**
22	Dataset 22	0.2761	0.3609	0.2766	0.3204	0.3409	0.2982	**0.2544**
23	Dataset 23	0.0164	0.0188	**0.0116**	0.0275	0.0269	**0.0116**	**0.0116**

Table 14: Results of V-bmSGO and all other approaches based on worse FM

S. No	Datasets	CS	GWO	HS	PSO	BA	TLBO	V-bmSGO
1	Dataset1	0.0395	0.0480	**0.0378**	0.0514	0.0486	0.0474	0.0395
2	Dataset 2	0.0477	0.0583	**0.0411**	0.0641	0.0612	0.0528	0.0421
3	Dataset3	0.1383	0.1881	0.1196	0.2045	0.2009	0.1634	**0.1113**
4	Dataset 4	0.0442	0.0492	0.0426	0.0495	0.0498	0.0467	**0.0400**
5	Dataset 5	0.0492	0.0600	0.0490	0.0737	0.0580	0.0593	**0.0415**
6	Dataset 6	0.2960	0.3209	**0.2937**	0.3242	0.3357	0.3007	0.2978
7	Dataset 7	0.2241	0.2654	0.2449	0.2687	0.2754	0.2607	**0.2225**
8	Dataset 8	0.1741	0.2007	**0.1463**	0.2173	0.2327	0.1926	0.1586
9	Dataset 9	0.1391	0.1634	0.1347	0.1687	0.1740	0.1598	**0.1078**
10	Dataset 10	0.1219	0.0781	0.0507	0.1847	0.1607	0.1718	**0.0487**
11	Dataset 11	0.1650	0.2324	**0.1260**	0.2447	0.2335	0.1945	0.1377
12	Dataset 12	0.1658	0.1527	0.1486	0.2279	0.2299	0.1486	**0.0331**
13	Dataset 13	0.1654	0.2237	0.1400	0.2460	0.2460	0.1967	**0.1084**
14	Dataset 14	0.0301	0.0346	0.0282	0.0361	0.0377	0.0333	**0.0276**
15	Dataset 15	0.1665	0.2149	0.1586	0.2228	0.2331	0.1979	**0.1443**
16	Dataset 16	0.1366	0.1736	**0.1241**	0.1911	0.1837	0.1555	0.1348
17	Dataset 17	0.2122	**0.1836**	**0.1836**	0.2650	0.2886	**0.1836**	0.2205
18	Dataset 18	0.0566	0.0914	**0.0427**	0.0980	0.0908	0.0782	0.0600
19	Dataset 19	0.2222	0.2428	**0.2088**	0.2834	0.2822	0.2413	**0.2095**
20	Dataset 20	0.0173	0.0211	0.0173	0.0625	0.0714	0.0211	**0.0150**
21	Dataset 21	0.0637	0.1196	0.0434	0.1308	0.1295	0.0930	**0.0013**
22	Dataset 22	0.3631	0.4468	0.2799	0.4457	0.4479	0.4047	**0.3171**
23	Dataset 23	0.0275	0.0813	0.0282	0.0768	0.0958	0.1828	**0.0215**

Table 15: Results of V-bmSGO and all other approaches based on standard deviation FM

S.NO	Datasets	CS	GWO	HS	PSO	BA	TLBO	V-bmSGO
1	Dataset1	7.6316e-04	0.0031	**1.4238e-17**	0.0038	0.0033	0.0032	3.8990e-04
2	Dataset 2	0.0038	0.0050	0.0027	0.0049	0.0052	0.0050	**0.0026**
3	Dataset3	0.0145	0.0166	**0.0066**	0.0150	0.0119	0.0116	0.0124
4	Dataset 4	0.0019	0.0021	0.0018	0.0016	0.0019	**0.0011**	0.0016
5	Dataset 5	0.0053	0.0043	0.0050	0.0074	0.0087	0.0058	**0.0040**
6	Dataset 6	0.1121	**0.0147**	0.0848	0.0213	0.0190	0.0650	0.1203
7	Dataset 7	6.3506e-04	0.0161	0.0080	0.0155	0.0176	0.0122	**2.8477e-17**
8	Dataset 8	0.0115	0.0108	0.0029	0.0169	0.0225	0.0127	**0.0086**
9	Dataset 9	0.0092	0.0063	0.0106	0.0084	0.0121	**0.0061**	0.0120
10	Dataset 10	0.0194	0.0071	**0.0059**	0.0348	0.0320	0.0315	0.0061
11	Dataset 11	0.0174	0.0265	**0.0102**	0.0265	0.0264	0.0225	0.0138
12	Dataset 12	0.0396	0.0145	0.0348	0.0428	0.0297	0.0193	**0.0104**
13	Dataset 13	0.0204	0.0202	**0.0087**	0.0268	0.0247	0.0223	0.0184
14	Dataset 14	0.0021	**0.0012**	0.0013	0.0020	0.0022	0.0019	0.0020
15	Dataset 15	0.0159	0.0162	0.0166	0.0190	0.0197	**0.0112**	0.0192
16	Dataset 16	0.0071	0.0153	**0.0059**	0.0166	0.0152	0.0075	0.0079
17	Dataset 17	0.0117	**0**	**0**	0.0165	0.0214	**0**	0.0102
18	Dataset 18	0.0047	0.0097	**0.0031**	0.0124	0.0101	0.0077	0.0092
19	Dataset 19	0.0061	0.0053	**0.0038**	0.0165	0.0154	0.0067	0.0061
20	Dataset 20	0.0040	**0.0012**	0.0040	0.0124	0.0157	0.0039	0.0043
21	Dataset 21	0.0263	0.0251	0.0107	0.0166	0.0225	0.0180	**2.2247e-19**
22	Dataset 22	0.0219	0.0240	7.4916e-04	0.0265	0.0305	0.0339	**0.0136**
23	Dataset 23	0.0032	0.0170	0.0025	0.0158	0.0198	0.0363	**0.0020**

Table 16: p-values of the WRS test of V-bmSGO vs all approaches (p ≥ 0.05 are boldfaced)

SL. No	Datasets	CS	GWO	HS	PSO	BA	TLBO
1	Dataset1	**7.8060e-01**	1.2020e-07	1.6859e-06	1.0360e-06	3.4040e-07	2.5951e-04
2	Dataset 2	1.8500e-02	2.9409e-07	**8.9240e-01**	6.7098e-08	6.7098e-08	8.0327e-06
3	Dataset3	6.6000e-05	6.7956e-08	2.5937e-05	6.7765e-08	6.7956e-08	6.7956e-08
4	Dataset 4	3.9874e-06	6.7574e-08	1.1034e-05	6.7860e-08	6.7956e-08	6.7956e-08
5	Dataset 5	7.0000e-03	4.8217e-08	**1.8300e-01**	6.9802e-08	5.1661e-08	1.7356e-07
6	Dataset 6	1.1400e-02	3.6573e-05	**7.7130e-01**	1.6111e-06	1.6111e-06	5.0421e-04
7	Dataset 7	1.9600e-02	6.8412e-09	9.6000e-03	6.3827e-09	7.8321e-09	2.8596e-08
8	Dataset 8	1.4500e-02	5.5156e-08	**8.7900e-01**	7.7888e-08	5.8203e-07	3.4322e-07
9	Dataset 9	7.3942e-08	6.3852e-08	2.0937e-07	6.3219e-08	6.4034e-08	6.4034e-08
10	Dataset 10	1.0373e-04	6.7288e-08	1.7900e-02	6.7956e-08	6.7956e-08	6.7956e-08
11	Dataset 11	1.4000e-03	1.1222e-07	**6.0580e-01**	6.2147e-08	6.1529e-08	1.7479e-07
12	Dataset 12	7.1756e-06	1.8535e-08	**7.5950e-01**	2.4567e-08	1.9447e-08	1.9319e-08
13	Dataset 13	1.0631e-07	6.7956e-08	1.0486e-07	6.7478e-08	6.7860e-08	6.7956e-08
14	Dataset 14	2.7378e-04	6.7765e-08	1.0000e-03	6.7574e-08	6.7765e-08	9.1601e-08
15	Dataset 15	2.0334e-05	6.7669e-08	1.0600e-02	6.7574e-08	6.3501e-08	6.7383e-08
16	Dataset 16	3.6276e-04	3.3302e-07	**7.4480e-01**	8.9345e-08	7.6694e-08	1.0372e-07
17	Dataset 17	2.2000e-03	**1.6260e-01**	**1.6260e-01**	3.4064e-08	2.4747e-08	**1.6260e-01**
18	Dataset 18	3.6400e-02	1.1439e-07	1.7800e-02	1.0678e-07	5.6044e-07	7.4790e-07
19	Dataset 19	6.6737e-06	6.7574e-08	**2.9770e-01**	1.6571e-07	6.7956e-08	6.7765e-08
20	Dataset 20	7.9867e-05	4.9221e-08	**1.7470e-01**	6.1091e-08	7.2599e-08	1.3985e-06
21	Dataset 21	7.9772e-09	2.8546e-08	7.9189e-09	7.9772e-09	7.9189e-09	7.8754e-09
22	Dataset 22	2.1396e-07	3.4463e-08	1.0236e-05	3.4357e-08	3.4410e-08	6.3893e-08
23	Dataset 23	**1.7500e-02**	2.9407e-06	3.6602e-04	6.6438e-08	6.5597e-08	1.0720e-02

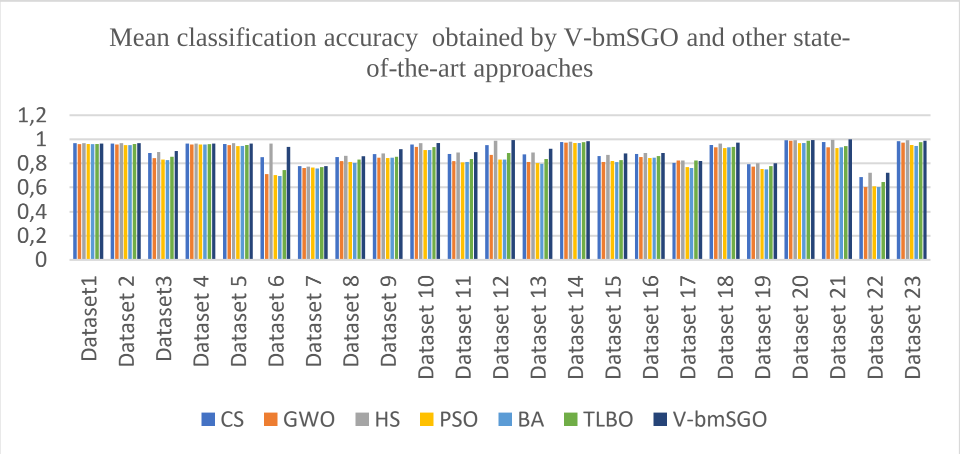


Figure 7: Chart on mean classification accuracy obtained by V-bmSGO and other state-of-the-art approaches

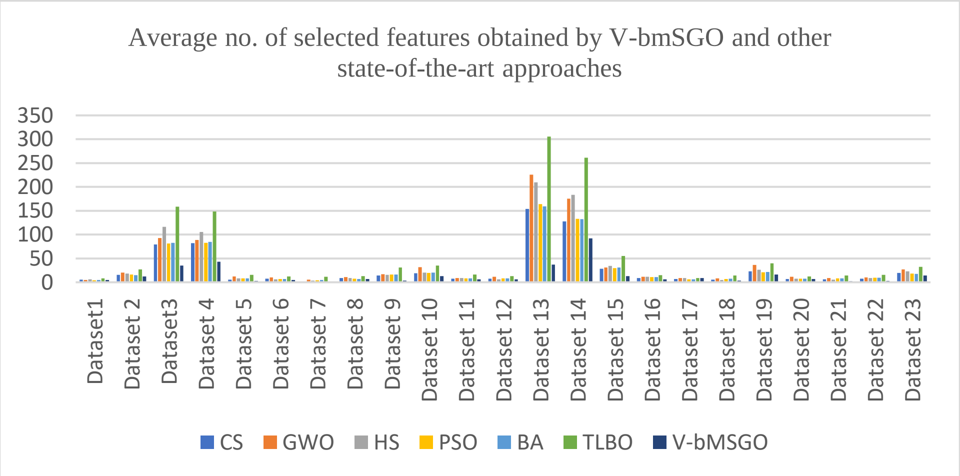


Fig 8: Chart on average no. of selected features using V-bmSGO and other state-of-the-art approaches

3.7 Experiment 3: The performance comparison with the latest optimization algorithms

In this comparative analysis, the proposed approach V-bMSGO is pitted against several latest optimization algorithms including CSA, GOA, MVO, SCA, WOA, and SSA. The classification accuracy (CA) results obtained by these algorithms are presented in Table 17 and visually shown by Fig 9. Notably, V-bmSGO showcases superior performance over all other optimizers on the majority of datasets, with the exception being the tic-tac-toe dataset where CSA, MVO, SSA, and WOA perform equally well. This outcome underscores V-bmSGO's adeptness in effectively navigating the solution search space and identifying the optimal feature subset with the highest CA. The rankings in Table 17 highlight V-bmSGO in first place, followed by MVO in second place, and SCA in third place, demonstrating the robustness and efficacy of V-bmSGO in comparison to contemporary optimization algorithms.

The optimal feature subset selection results are summarized in Table 18 and visually shown by chart Fig 10. Across all datasets except the tic-tac-toe dataset, the V-bmSGO approach demonstrates exceptional performance. Notably, in the tic-tac-toe dataset, GOA outperforms V-bmSGO. This observation suggests that the V-shaped transfer function implemented in V-bmSGO can substantially enhance the original MSGO's performance when it comes to selecting the minimum number of attributes or features.

The statistical mean fitness measure results are summarized in Table 19. One can remark that V-bmSGO outperformed GOA in all datasets. V-bmSGO outperformed CSA, MVO, SSA, WOA, and SCA in all datasets except one dataset i.e., the tic-tac-toe dataset out of twenty-three datasets.

The statistical best fitness measure results are summarized in Table 20. The V-bmSGO outperformed CSA in all datasets except three datasets: exactly2, heartEW, and tic-tac-toe datasets where both perform equally. The V-bmSGO outperformed GOA in all datasets except exactly2 dataset where both perform equally. The V-bmSGO outperformed MVO and WOA in all datasets except two datasets: breast cancer and tic-tac-toe datasets where all perform equally. V-bmSGO outperforms SSA in twenty datasets and performs equally in three datasets. The V-bmSGO outperformed SCA in all datasets except two datasets: exactly2 and tic-tac-toe datasets where both perform equally.

The statistical worst fitness measure results are summarized in Table 21. Here we see that V-bmSGO outperformed GOA and SCA in all datasets. V-bmSGO outperformed CSA, MVO, SSA, and WOA in all datasets except the tic-tac-toe dataset.

The statistical standard deviation fitness measure results are summarized in Table 22. Here we see that V-bmSGO outperformed in nine datasets, CSA outperformed in four datasets, GOA outperformed in two datasets, MVO outperformed in seven datasets, SSA outperformed in two datasets, WOA outperformed in two datasets, and SCA outperformed in no one datasets out of twenty-one datasets. CSA, MVO, SSA, and WOA performed equally in the tic-tac-toe dataset in regards to standard deviation fitness measure.

Table 23 presents the p-values of the WRS test obtained at a 5% significance level for comparing V-bmSGO with other newest approaches. A p-value less than 0.05 indicates a significant difference at a 5% level of significance. In Table 23, p-values greater than or equal to 0.05 are boldfaced. It can be observed that for all the newest approaches except GOA, there is one case out of twenty-one where the p-value is greater than or equal to 0.05.

Table 17: Results of proposed V-bmSGO and latest approaches based on CA

S. No	Datasets	CSA	GOA	MVO	SSA	WOA	SCA	V-bmSGO
1	Dataset1	0.9567	0.9601	0.9643	0.9611	0.9623	0.9610	**0.9664**
2	Dataset 2	0.9432	0.9512	0.9633	0.9519	0.9582	0.9582	**0.9668**
3	Dataset3	0.8055	0.8349	0.8555	0.8351	0.8431	0.8403	**0.9046**
4	Dataset 4	0.9556	0.9584	0.9605	0.9581	0.9593	0.9595	**0.9657**
5	Dataset 5	0.9468	0.9404	0.9578	0.9500	0.9518	0.9525	**0.9663**
6	Dataset 6	0.6874	0.6962	0.7599	0.6973	0.7219	0.7292	**0.9400**
7	Dataset 7	0.7455	0.7578	0.7678	0.7592	0.7623	0.7650	**0.7760**
8	Dataset 8	0.8444	0.8063	0.8456	0.8181	0.8319	0.8330	**0.8593**
9	Dataset 9	0.8318	0.8483	0.8565	0.8517	0.8557	0.8494	**0.9170**
10	Dataset 10	0.9309	0.9009	0.9510	0.9335	0.9453	0.9487	**0.9699**
11	Dataset 11	0.7588	0.8108	0.8378	0.8196	0.8243	0.8243	**0.8926**
12	Dataset 12	0.8600	0.8243	0.9169	0.8687	0.8903	0.8921	**0.9958**
13	Dataset 13	0.7838	0.8097	0.8392	0.8027	0.8280	0.8238	**0.9236**
14	Dataset 14	0.9738	0.9710	0.9772	0.9733	0.9762	0.9747	**0.9837**
15	Dataset 15	0.7654	0.8087	0.8221	0.8106	0.8125	0.8173	**0.8841**
16	Dataset 16	0.8201	0.8493	0.8627	0.8541	0.8511	0.8541	**0.8877**
17	Dataset 17	**0.8246**	0.7709	**0.8246**	**0.8246**	**0.8246**	0.8218	0.8209
18	Dataset 18	0.8913	0.9287	0.9373	0.9277	0.9323	0.9337	**0.9733**
19	Dataset 19	0.7644	0.7574	0.7850	0.7663	0.7763	0.7798	**0.8010**
20	Dataset 20	0.9747	0.9742	0.9893	0.9888	0.9893	0.9888	**0.9955**

21	Dataset 21	0.8772	0.9309	0.9427	0.9313	0.9458	0.9404	**1**
22	Dataset 22	0.5202	0.6149	0.6362	0.5957	0.5957	0.6277	**0.7234**
23	Dataset 23	0.8232	0.9497	0.9828	0.9560	0.9560	0.9773	**0.9888**
Average		0.8385	0.8546	0.8798	0.8624	0.8693	0.8719	**0.9175**

Table 18: Results of proposed V-bmSGO and latest approaches based on the average number of. Features

S. No	Datasets	CSA	GOA	MVO	SSA	WOA	SCA	V-bmSGO
1	Dataset1	9	5	5.1000	4.7000	5.3000	5.1500	**4.5**
2	Dataset 2	29.70	16.05	22.70	16.70	21.80	22.60	**12.10**
3	Dataset3	166	85.75	112.45	82.80	118.90	108.85	**35.35**
4	Dataset 4	162.10	82.45	102.05	84.40	97.45	96.35	**43.25**
5	Dataset 5	16	7.8000	12.30	11.80	11.30	10.40	**2.6500**
6	Dataset 6	12.90	6.9000	10.05	7.8500	10.60	10	**5**
7	Dataset 7	13	4.5000	6.7500	4.9000	7.8500	5.4000	**1**
8	Dataset 8	12.90	8.2500	9.7500	9.4000	10.20	10.15	**7.1500**
9	Dataset 9	34	16.05	21.30	15.20	18.60	18.900	**3.4000**
10	Dataset 10	34.90	20.15	28.60	30.35	30.75	29.600	**12.90**
11	Dataset 11	18	9.3000	10.30	9.2500	11.050	9.7500	**6**
12	Dataset 12	12.90	7.3000	9.4000	11.950	10.750	10.750	**6.1500**
13	Dataset 13	323.35	164.45	236.15	201.85	220.70	197.65	**37.15**
14	Dataset 14	265	134.45	205.60	191.70	213.35	180.80	**91.85**
15	Dataset 15	60	29.85	37.50	29.55	36.55	33.45	**12.850**
16	Dataset 16	21.60	11.05	13.55	10.45	13.600	13.60	**6.3500**
17	Dataset 17	9	**6.4000**	9	9	9	8.8000	8.60
18	Dataset 18	15.800	6.4500	8.7000	7.1000	8.7500	8.3500	**3.2000**
19	Dataset 19	39.85	23.05	32.30	34.85	34.75	33.15	**16.15**
20	Dataset 20	12.950	7.4000	8.7000	12.75	11.45	10.700	**6.5000**
21	Dataset 21	15.700	7.8500	9.5500	8.5000	10.150	9.1500	**2**
22	Dataset 22	18	8.4000	10.750	9.2500	10	10.250	**3.1000**
23	Dataset 23	33.60	18.150	25.700	17.400	27.450	27.400	**14.150**

Table 19: Results of proposed V-bmSGO and latest approaches based on mean FM

S. No	Datasets	CSA	GOA	MVO	SSA	WOA	SCA	V-bmSGO
1	Dataset1	0.0403	0.0450	0.0410	0.0437	0.0432	0.0443	**0.0383**
2	Dataset 2	0.0451	0.0536	0.0439	0.0532	0.0486	0.0489	**0.0369**
3	Dataset3	0.1483	0.1686	0.1499	0.1683	0.1625	0.1646	**0.0966**
4	Dataset 4	0.0439	0.0462	0.0453	0.0465	0.0462	0.0459	**0.0366**
5	Dataset 5	0.0479	0.0639	0.0495	0.0569	0.0547	0.0535	**0.0350**
6	Dataset 6	0.2713	0.3061	0.2454	0.3057	0.2835	0.2758	**0.0632**
7	Dataset 7	0.2237	0.2432	0.2351	0.2422	0.2414	0.2368	**0.2225**
8	Dataset 8	0.1672	0.1981	0.1604	0.1873	0.1743	0.1732	0.1448
9	Dataset 9	0.1428	0.1549	0.1483	0.1513	0.1483	0.1546	**0.0831**
10	Dataset 10	0.0603	0.1037	0.0565	0.0743	0.0627	0.0591	**0.0334**
11	Dataset 11	0.1547	0.1925	0.1663	0.1837	0.1801	0.1793	**0.1097**
12	Dataset 12	0.1114	0.1796	0.0895	0.1392	0.1169	0.1151	**0.0089**
13	Dataset 13	0.1639	0.1935	0.1665	0.2015	0.1771	0.1805	**0.0768**
14	Dataset 14	0.0304	0.0338	0.0304	0.0336	0.0316	0.0319	**0.0242**
15	Dataset 15	0.1770	0.1944	0.1824	0.1925	0.1917	0.1864	**0.1168**
16	Dataset 16	0.1380	0.1543	0.1421	0.1492	0.1536	0.1506	**0.1141**
17	Dataset 17	**0.1836**	0.2339	**0.1836**	**0.1836**	**0.1836**	0.1862	0.1869
18	Dataset 18	0.0571	0.0747	0.0675	0.0760	0.0725	0.0709	**0.0442**
19	Dataset 19	0.2303	0.2459	0.2210	0.2401	0.2302	0.2263	**0.2010**
20	Dataset 20	0.0174	0.0313	0.0173	0.0209	0.0194	0.0194	**0.0094**
21	Dataset 21	0.0545	0.0733	0.0627	0.0734	0.0600	0.0647	**0.0013**
22	Dataset 22	0.3458	0.3859	0.3662	0.4054	0.0600	0.3743	**0.2756**
23	Dataset 23	0.0277	0.0551	0.0246	0.0487	0.3710	0.0305	**0.0174**

Table 20: Results of proposed V-bmSGO and latest approaches based on the best FM

S. No	Datasets	CSA	GOA	MVO	SSA	WOA	SCA	V-bmSGO
1	Dataset1	0.0384	0.0384	**0.0378**	0.0384	**0.0378**	0.0395	**0.0378**
2	Dataset 2	0.0401	0.0397	0.0386	0.0477	0.0421	0.0389	**0.0321**
3	Dataset3	0.1380	0.1343	0.1297	0.1421	0.1332	0.1299	**0.0641**
4	Dataset 4	0.0380	0.0411	0.0432	0.0443	0.0433	0.0418	**0.0336**
5	Dataset 5	0.0310	0.0465	0.0432	0.0413	0.0420	0.0388	**0.0291**
6	Dataset 6	0.0826	0.2703	0.0747	0.2858	0.2579	0.2002	**0.0046**
7	Dataset 7	**0.2225**	**0.2225**	0.2233	**0.2225**	0.2233	**0.2225**	**0.2225**
8	Dataset 8	**0.1374**	0.1609	0.1389	**0.1374**	0.1470	0.1455	**0.1374**
9	Dataset 9	0.1166	0.1391	0.1338	0.1276	0.1279	0.1376	**0.0524**
10	Dataset 10	0.0482	0.0625	0.0473	0.0569	0.0558	0.0409	**0.0267**
11	Dataset 11	0.1243	0.1516	0.1260	0.1260	0.1265	0.1388	**0.0970**
12	Dataset 12	0.0331	0.1356	0.0252	0.0889	0.0715	0.0861	**0.0046**
13	Dataset 13	0.1180	0.1652	0.1414	0.1654	0.1425	0.1388	**0.0300**
14	Dataset 14	0.0273	0.0284	0.0281	0.0285	0.0279	0.0268	**0.0187**
15	Dataset 15	0.1476	0.1685	0.1581	0.1667	0.1663	0.1568	**0.0685**
16	Dataset 16	0.1089	0.1167	0.1098	0.1154	0.1371	0.1163	**0.0997**
17	Dataset 17	**0.1836**	0.2062	**0.1836**	**0.1836**	**0.1836**	**0.1836**	**0.1836**
18	Dataset 18	0.0493	0.0566	0.0559	0.0566	0.0553	0.0566	**0.0295**
19	Dataset 19	0.2180	0.2173	0.2024	0.2209	0.2142	0.2158	**0.1885**
20	Dataset 20	0.0150	0.0157	0.0077	0.0173	0.0069	0.0077	**0.0046**
21	Dataset 21	0.0063	0.0050	0.0273	0.0452	0.0044	0.0265	**0.0013**
22	Dataset 22	0.2993	0.3193	0.3210	0.3620	0.3215	0.3210	**0.2544**
23	Dataset 23	0.0170	0.0221	0.0185	0.0323	0.0182	0.0188	**0.0116**

Table 21: Results of proposed V-bmSGO and latest approaches based on worst FM

S. No	Datasets	CSA	GOA	MVO	SSA	WOA	SCA	V-bMSGO
1	Dataset1	0.0440	0.0536	0.0463	0.0497	0.0480	0.0491	**0.0395**
2	Dataset 2	0.0515	0.0629	0.0535	0.0616	0.0573	0.0542	**0.0421**
3	Dataset3	0.1623	0.1878	0.1710	0.1853	0.1886	0.1918	**0.1113**
4	Dataset 4	0.0464	0.0489	0.0472	0.0492	0.0499	0.0486	**0.0400**
5	Dataset 5	0.0550	0.1043	0.0542	0.0600	0.0950	0.0719	**0.0415**
6	Dataset 6	0.2985	0.3397	0.2985	0.3209	0.3068	0.3068	**0.2978**
7	Dataset 7	0.2248	0.2704	0.2500	0.2628	0.2611	0.2718	**0.2225**
8	Dataset 8	0.1852	0.2262	0.1779	0.2007	0.2007	0.1926	**0.1586**
9	Dataset 9	0.1557	0.1702	0.1610	0.1631	0.1675	0.1663	**0.1078**
10	Dataset 10	0.0735	0.1914	0.0659	0.0781	0.0717	0.0723	**0.0487**
11	Dataset 11	0.1912	0.2341	0.2096	0.2464	0.2313	0.2090	**0.1377**
12	Dataset 12	0.1486	0.2335	0.1419	0.1486	0.1486	0.1439	**0.0331**
13	Dataset 13	0.1928	0.2254	0.1952	0.2241	0.1955	0.2215	**0.1084**
14	Dataset 14	0.0335	0.0381	0.0318	0.0348	0.0334	0.0336	**0.0276**
15	Dataset 15	0.1954	0.2243	0.2056	0.2228	0.2149	0.2231	**0.1443**
16	Dataset 16	0.1537	0.1684	0.1550	0.1684	0.1716	0.1716	**0.1348**
17	Dataset 17	**0.1836**	0.2650	**0.1836**	**0.1836**	**0.1836**	0.2350	0.2205
18	Dataset 18	0.0691	0.1034	0.0848	0.1034	0.0854	0.0873	**0.0600**
19	Dataset 19	0.2428	0.2662	0.2301	0.2428	0.2374	0.2359	**0.2095**
20	Dataset 20	0.0211	0.0499	0.0196	0.0211	0.0211	0.0269	**0.0150**
21	Dataset 21	0.0662	0.1314	0.1163	0.1256	0.0848	0.0854	**0.0013**
22	Dataset 22	0.3836	0.4268	0.4052	0.4285	0.4047	0.4462	**0.3171**
23	Dataset 23	0.0344	0.1228	0.0347	0.0687	0.0964	0.1129	**0.0215**

Table 22: Results of proposed V-bmSGO and latest approaches based on standard deviation FM

S. No	Datasets	CSA	GOA	MVO	SSA	WOA	SCA	V-bMSGO
1	Dataset1	0.0019	0.0040	0.0030	0.0036	0.0034	0.0025	**3.8990e-04**
2	Dataset 2	0.0028	0.0060	0.0036	0.0037	0.0041	0.0036	**0.0026**
3	Dataset3	**0.0072**	0.0145	0.0090	0.0137	0.0129	0.0124	0.0124
4	Dataset 4	0.0020	0.0023	**0.0012**	**0.0012**	0.0014	0.0019	0.0016
5	Dataset 5	0.0065	0.0139	**0.0028**	0.0058	0.0110	0.0063	0.0040
6	Dataset 6	0.0498	**0.0152**	0.0482	0.0123	0.0129	0.0247	0.1203
7	Dataset 7	6.3628e-04	0.0188	0.0107	0.0143	0.0147	0.0151	**2.8477e-17**

8	Dataset 8	0.0144	0.0192	0.0119	0.0165	0.0137	0.0136	**0.0086**
9	Dataset 9	0.0093	**0.0081**	0.0086	0.0108	0.0106	0.0085	0.0120
10	Dataset 10	0.0065	0.0441	**0.0044**	0.0068	0.0052	0.0076	0.0061
11	Dataset 11	0.0181	0.0277	0.0241	0.0318	0.0272	0.0241	**0.0138**
12	Dataset 12	0.0385	0.0227	0.0314	0.0192	0.0226	0.0158	**0.0104**
13	Dataset 13	0.0233	0.0204	**0.0105**	0.0227	0.0152	0.0224	0.0184
14	Dataset 14	0.0016	0.0028	**0.0012**	0.0017	0.0016	0.0015	0.0020
15	Dataset 15	0.0123	0.0145	**0.0112**	0.0167	0.0158	0.0167	0.0192
16	Dataset 16	0.0098	0.0131	0.0115	0.0144	0.0113	0.0133	0.0079
17	Dataset 17	**0**	0.0189	**0**	**0**	**0**	0.0115	0.0102
18	Dataset 18	**0.0055**	0.0129	0.0078	0.0140	0.0081	0.0098	0.0092
19	Dataset 19	0.0072	0.0118	0.0060	0.0062	**0.0056**	0.0058	0.0061
20	Dataset 20	**0.0020**	0.0112	0.0024	8.6003e-04	0.0030	0.0034	0.0043
21	Dataset 21	0.0178	0.0271	0.0190	0.0175	0.0217	0.0119	**2.2247e-19**
22	Dataset 22	0.0232	0.0266	0.0264	0.0208	0.0219	0.0248	**0.0136**
23	Dataset 23	0.0043	0.0260	0.0059	0.0100	0.0167	0.0200	**0.0020**

Table 23. p-values of the WRS test of the proposed V-bmSGO vs other latest approaches ($p \geq 0.05$ are boldfaced)

Sl No	Datasets	CSA	GOA	MVO	SSA	WOA	SCA
1	Dataset1	3.0183e-06	1.2282e-07	0.0162	9.1944e-07	1.1012e-04	3.5056e-08
2	Dataset 2	1.0400e-07	9.0467e-08	6.7931e-07	6.7098e-08	7.2653e-08	1.2246e-07
3	Dataset3	6.7956e-08	6.7765e-08	6.7860e-08	6.7956e-08	6.7956e-08	6.7860e-08
4	Dataset 4	1.2346e-07	6.7956e-08	6.7860e-08	6.7956e-08	6.7956e-08	6.7956e-08
5	Dataset 5	1.9150e-06	5.2268e-08	4.9584e-08	7.3982e-08	5.2115e-08	9.4721e-08
6	Dataset 6	5.5020e-04	2.8501e-07	8.2286e-04	6.6955e-07	2.3276e-04	2.3378e-04
7	Dataset 7	7.9391e-08	2.9150e-08	7.6046e-09	3.5000e-07	7.7603e-09	2.8052e-08
8	Dataset 8	6.4855e-06	5.8100e-08	3.2972e-05	5.5671e-07	3.4060e-07	7.9910e-07
9	Dataset 9	6.3943e-08	6.4034e-08	6.3943e-08	6.3943e-08	6.3943e-08	6.4034e-08
10	Dataset 10	9.1728e-08	6.7956e-08	7.8870e-08	3.9954e-08	6.7860e-08	9.1728e-08
11	Dataset 11	2.3496e-07	6.1882e-08	1.5191e-07	9.5732e-08	9.7339e-08	6.1882e-08
12	Dataset 12	2.4407e-08	1.9447e-08	3.9591e-08	8.8511e-09	1.9287e-08	1.8752e-08
13	Dataset 13	6.7860e-08	6.7860e-08	6.7956e-08	6.5970e-08	6.7860e-08	6.7860e-08
14	Dataset 14	7.8760e-08	6.7765e-08	6.7860e-08	5.7186e-08	6.7765e-08	7.8760e-08
15	Dataset 15	6.7669e-08	6.7574e-08	6.7669e-08	6.7765e-08	5.5557e-08	6.7765e-08
16	Dataset 16	8.8830e-07	1.8580e-07	8.8725e-07	2.3202e-07	6.5783e-08	2.1635e-07
17	Dataset 17	**1.6260e-01**	3.9697e-08	**1.6260e-01**	**1.6260e-01**	**1.6260e-01**	**6.1470e-01**
18	Dataset 18	3.7422e-05	2.4022e-07	3.6990e-07	2.3960e-07	1.5480e-07	3.6944e-07
19	Dataset 19	6.7956e-08	6.7956e-08	2.9598e-07	2.9550e-08	6.7956e-08	6.7860e-08
20	Dataset 20	1.0387e-07	6.1794e-08	2.1954e-07	1.0238e-08	3.0918e-07	2.3084e-07
21	Dataset 21	3.7422e-05	2.4022e-07	3.6990e-07	2.3960e-07	1.5480e-07	3.6944e-07
22	Dataset 22	4.6779e-08	3.4251e-08	3.4251e-08	3.4198e-08	3.4410e-08	3.4304e-08
23	Dataset 23	1.1681e-06	8.9593e-08	2.0600e-02	6.6344e-08	1.9000e-03	3.2429e-05

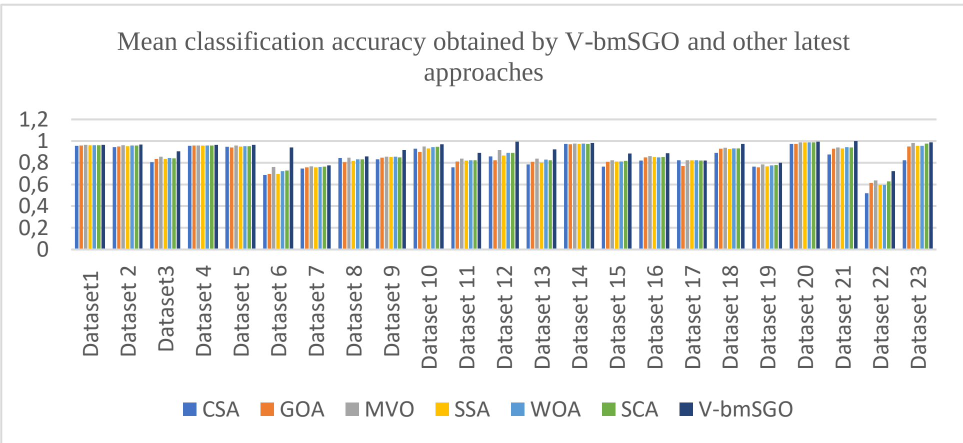


Figure 9: Chart on mean classification accuracy obtained by V-bmSGO and other latest approaches

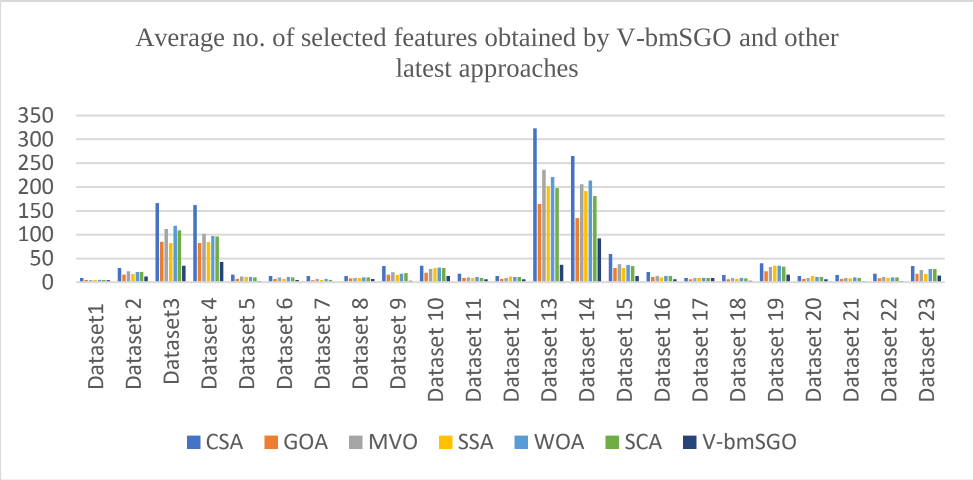


Figure 10: Chart on average number of selected features obtained by V-bmSGO and other latest approaches

Table 24: Overall mean CA and average NSF of all algorithms

Dataset	CS		GWO		HS		PSO		BA	
	CA	NSF	CA	NSF	CA	NSF	CA	NSF	CA	NSF
Dataset1	96.71%	5.3000	96.13%	4.8000	**96.86%**	6	96.21%	**4.2000**	96.11%	4.6000
Dataset 2	96.56%	15.85	95.68%	20.35	**96.89%**	18.30	95.09%	15.90	95.28%	14.95
Dataset3	88.70%	79.25	84.37%	92.55	89.62%	116.55	83.13%	81.25	82.75%	82.45
Dataset 4	96.43%	81.80	95.84%	88.90	**96.62%**	105.40	95.74%	82.450	95.79%	84.65
Dataset 5	96.38%	5.7500	95.11%	12.500	**96.70%**	7.9000	94.54%	7.9500	94.68%	8.5000
Dataset 6	85.22%	7.5500	71.11%	10.10	**96.57%**	6.3500	70.27%	7	69.60%	6.5500
Dataset 7	**77.60%**	1.4500	76.43%	5.7000	77.37%	3.5500	76.63%	4.3500	75.72%	4.9500
Dataset 8	85.41%	8.5500	82.04%	10.80	**86.44%**	8.6500	81.48%	7.6500	80.67%	6.8500
Dataset 9	87.76%	13.95	84.83%	17.20	88.27%	15.35	84.60%	16	84.94%	16.40
Dataset 10	95.87%	18.95	93.76%	31.35	96.79%	20.10	91.17%	19.300	91.12%	20
Dataset 11	87.91%	7.4500	81.89%	8.7500	89.17%	8.8500	80.74%	8.2000	81.28%	8.2500
Dataset 12	95.32%	7.3000	87.18%	11.450	98.97%	6.4500	83.24%	7.9000	83.26%	8.0500
Dataset 13	87.55%	153.75	81.49%	225.35	89.19%	209.45	80.44%	164.15	79.89%	159.15
Dataset 14	97.79%	127.45	97.37%	175.15	98.05%	183.55	97.16%	132.95	97.01%	132.40
Dataset 15	86.11%	28.20	81.39%	31.10	87.31%	34.15	82.07%	29.40	81.11%	30.75
Dataset 16	88.13%	8.9500	85.49%	11.20	88.70%	11.80	84.70%	10.60	84.96%	10.80
Dataset 17	80.71%	7.0500	**82.46%**	9	**82.46%**	9	76.93%	**5.9000**	76.30%	5.9500
Dataset 18	95.60%	5.8000	93.23%	8.1000	96.50%	5.0500	92.83%	7.1000	93.47%	7.2500
Dataset 19	79.15%	22.75	77.33%	36.40	**80.14%**	26.45	75.45%	21.05	74.88%	21.65
Dataset 20	99.10%	7.1000	98.88%	11.80	99.10%	7.6500	96.74%	7.2000	97.08%	7.4000
Dataset 21	97.77%	6.4500	93.46%	8.7500	99.60%	5.5000	92.91%	8.3500	93.36%	8.3000
Dataset 22	68.72%	7.2500	60.43%	9.9000	**72.34%**	8.8500	60.96%	9.3000	60.43%	9.3000
Dataset 23	98.50%	19.800	97.24%	26.900	**99.13%**	22.750	95.52%	18.500	94.81%	17.600
	TLBO		CSA		GOA		MVO		SSA	
	CA	NSF	CA	NSF	CA	NSF	CA	NSF	CA	NSF
Dataset1	96.33%	8.1500	95.67%	9	96.01%	5	96.43%	5.1000	96.11%	4.7000
Dataset 2	96.19%	27.05	94.32%	29.70	95.12%	16.05	96.33%	22.70	95.19%	16.70
Dataset3	85.71%	158.20	80.55%	166	83.49%	85.75	85.55%	112.45	83.51%	82.80
Dataset 4	96.04%	148.15	95.56%	162.10	95.84%	82.45	96.05%	102.05	95.81%	84.40
Dataset 5	95.50%	15.45	94.68%	16	94.04%	7.8000	95.78%	12.30	95.00%	11.80
Dataset 6	74.39%	12.35	68.74%	12.90	69.62%	6.9000	75.99%	10.05	69.73%	7.8500
Dataset 7	76.88%	11.55	74.55%	13	75.78%	4.5000	76.78%	6.7500	75.92%	4.9000
Dataset 8	83.30%	12.70	84.44%	12.90	80.63%	8.2500	84.56%	9.7500	81.81%	9.4000
Dataset 9	85.68%	31.05	83.18%	34	84.83%	16.05	85.65%	21.30	85.17%	15.20
Dataset 10	93.59%	35	93.09%	34.90	90.09%	20.15	95.10%	28.60	93.35%	30.35
Dataset 11	83.78%	16	75.88%	18	81.08%	9.3000	83.78%	10.30	81.96%	9.2500
Dataset 12	88.73%	12.65	86.00%	12.90	82.43%	7.3000	91.69%	9.4000	86.87%	11.950
Dataset 13	83.74%	305.30	78.38%	323.35	80.97%	164.45	83.92%	236.15	80.27%	201.85
Dataset 14	97.63%	260.95	97.38%	265	97.10%	134.45	97.72%	205.60	97.33%	191.70
Dataset 15	82.60%	55.45	76.54%	60	80.87%	29.85	82.21%	37.50	81.06%	29.55
Dataset 16	86.19%	15	82.01%	21.60	84.93%	11.05	86.27%	13.55	85.41%	10.45
Dataset 17	82.46%	9	82.46%	9	77.09%	6.4000	82.46%	9	82.46%	9

Dataset 18	93.83%	14.20	89.13%	15.800	92.87%	6.4500	93.73%	8.7000	92.77%	7.1000
Dataset 19	77.75%	39.40	76.44%	39.85	75.74%	23.05	78.50%	32.30	76.63%	34.85
Dataset 20	98.99%	12.35	97.47%	12.950	97.42%	7.4000	98.93%	8.7000	98.88%	12.75
Dataset 21	94.49%	14.150	87.72%	15.700	93.09%	7.8500	94.27%	9.5500	93.13%	8.5000
Dataset 22	64.57%	15.70	52.02%	18	61.49%	8.4000	63.62%	10.750	59.57%	9.2500
Dataset 23	97.65%	32.600	82.32%	33.60	94.97%	18.150	98.28%	25.700	95.60%	17.400

	WOA		SCA		V-bmSGO	
	CA	NSF	CA	NSF	CA	NSF
Dataset1	96.23%	5.3000	96.10%	5.1500	96.64%	4.5
Dataset 2	95.82%	21.80	95.82%	22.60	96.68%	**12.10**
Dataset3	84.31%	118.90	84.03%	108.85	90.46%	**35.35**
Dataset 4	95.93%	97.45	95.95%	96.35	96.57%	**43.25**
Dataset 5	95.18%	11.30	95.25%	10.40	96.63%	**2.6500**
Dataset 6	72.19%	10.60	72.92%	10	94.00%	**5**
Dataset 7	76.23%	7.8500	76.50%	5.4000	77.60%	**1**
Dataset 8	83.19%	10.20	83.30%	10.15	85.93%	**7.1500**
Dataset 9	85.57%	18.60	84.94%	18.900	91.70%	**3.4000**
Dataset 10	94.53%	30.75	94.87%	29.600	96.99%	**12.90**
Dataset 11	82.43%	11.050	82.43%	9.7500	89.26%	**6**
Dataset 12	89.03%	10.750	89.21%	10.750	99.58%	**6.1500**
Dataset 13	82.80%	220.70	82.38%	197.65	92.36%	**37.15**
Dataset 14	97.62%	213.35	97.47%	180.80	98.37%	**91.85**
Dataset 15	81.25%	36.55	81.73%	33.45	88.41%	**12.850**
Dataset 16	85.11%	13.600	85.41%	13.60	88.77%	**6.3500**
Dataset 17	82.46%	9	82.18%	8.8000	82.09%	8.60
Dataset 18	93.23%	8.7500	93.37%	8.3500	97.33%	**3.2000**
Dataset 19	77.63%	34.75	77.98%	33.15	80.10%	**16.15**
Dataset 20	98.93%	11.45	98.88%	10.700	99.55%	**6.5000**
Dataset 21	94.58%	10.150	94.04%	9.1500	100.00%	**2**
Dataset 22	59.57%	10	62.77%	10.250	72.34%	**3.1000**
Dataset 23	95.60%	27.450	97.73%	27.400	98.88%	**14.150**

According to the data provided by Table 24, the HS algorithm achieves the highest mean classification accuracy across several datasets. Specifically, it performs best on datasets 1, 2, 4, 5, 6, 8, 17, 19, 22, and 23. In contrast, the SGO algorithm demonstrates superior performance in a larger number of datasets, excelling in datasets 3, 7, 9, 10, 11, 12, 13, 14, 15, 16, 18, 20, 21, and 22. Interestingly, algorithms such as TLBO, GWO, CSA, MVO, and SSA each reach their peak mean classification accuracy solely on dataset 17.

Examining the mean number of selected features, the HS algorithm shows optimal performance only for dataset 1, whereas the PSO algorithm is best suited for dataset 17. For the remaining datasets, which include datasets 2 through 16 and 18 through 23, the SGO algorithm consistently selects the most effective number of features.

Furthermore, the SGO algorithm stands out by achieving both the highest mean classification accuracy and the best mean number of selected features for a significant subset of datasets. These include datasets 3, 7, 9, 10, 11, 12, 13, 14, 15, 16, 18, 20, 21, and 22. This dual accomplishment underscores the robustness and efficiency of the SGO algorithm across a diverse set of conditions.

Overall discussion

Based on the experimental results, we can confidently conclude that our proposed approaches demonstrate significant efficacy in solving FS problems compared to other methods. Notably, the outcomes of the stochastic wrapper-based FS approach consistently stand out. However, it's worth noting that the subset of features selected by the algorithm may vary depending on the specific application, posing a challenge for users in deciding which subset to adopt. Furthermore, our proposed approach employs the KNN classifier, which is a straightforward choice. Future investigations could explore the integration of alternative classifiers such as support vector machines or random forests, which may offer additional insights and performance enhancements in feature selection tasks.

4 Conclusion

In this study, we introduced bmSGO algorithms to address the FS problem using a wrapper approach. Our method involved converting continuous MSGO into binary form using transfer functions, specifically employing the V-shaped transfer function in V-bmSGO and the S-shaped transfer function in S-bmSGO. These approaches were designed to evaluate different search capabilities within the algorithms.

To frame the FS problem, we transformed it into a single-objective optimization challenge with a fitness function that reflects classification performance while minimizing the number of features. We conducted evaluations using twenty-three datasets from the UCI repository, comparing our bmSGO approaches against six state-of-the-art FS methods (PSO, HS, CS, BA, TLBO, GWO) and six latest optimization algorithms (SCA, SSA, CSA, GOA, MVO, WOA).

Our experimental findings indicate that our approaches perform exceptionally well in solving FS problems. Particularly, V-bmSGO showed a significant improvement over MSGO in terms of classification accuracy and feature selection. The simulation outcomes demonstrated that V-bmSGO excelled in searching the feature set space and converging towards optimal or near-optimal solutions better than other algorithms.

For future research, we aim to apply the bmSGO algorithm to diverse real-world problems such as facial emotion recognition, handwriting recognition, and script recognition. Additionally, hybridizing the MSGO algorithm with other population-based meta-heuristic algorithms for FS problems could be a promising path to explore.

Compliance with ethical standards

Conflict of interest: Authors declare that they have no conflict of interest in the publication of this paper.

References

[1] Bennasar, M., Hicks, Y., & Setchi, R. (2015). Feature selection using Joint Mutual Information Maximisation. Expert Systems with Applications, 42(22), 8520–8532. doi: 10.1016/j.eswa.2015.07.007

[2] Zawbaa, H. M., Emary, E., & Grosan, C. (2016). Feature selection via chaotic antlion optimization. PloS One, 11(3), e0150652. doi: 10.1371/journal.pone.0150652

[3] Liu, H., & Yu, L. (2005). Toward integrating feature selection algorithms for classification and clustering. IEEE Transactions on Knowledge and Data Engineering, 17(4), 491–502. doi:10.1109/tkde.2005.66

[4] Y. Yang, J.O. Pedersen, A comparative study on feature selection in text categorization, Icml, 97 (1997) 412–420.

[5] Naik, A., & Satapathy, S. C. (2013). Efficient clustering of dataset based on particle swarm optimization. International Journal of Computer Science Engineering and Information Technology Research (IJCSEITR), 3(1), 39–48.

[6] Naik, Anima, & Satapathy, S. C. (2014). Efficient clustering of dataset based on differential evolution. In Advances in Intelligent Systems and Computing. Proceedings of the International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA) 2013 (pp. 217–227). doi:10.1007/978-3-319-02931-3_26

[7] Han, J., Kamber, M., & Pei, J. (2012). Data Preprocessing. In Data Mining (pp. 83–124). doi:10.1016/b978-0-12-381479-1.00003-4

[8] Peng, H., Long, F., & Ding, C. (2005). Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(8), 1226–1238. doi:10.1109/TPAMI.2005.159

[9] Kira, K., & Rendell, L. A. (1992). A practical approach to feature selection. In Machine Learning Proceedings 1992 (pp. 249–256). doi:10.1016/b978-1-55860-247-2.50037-1

[10] Zheng, Z., Wu, X., & Srihari, R. (2004). Feature selection for text categorization on imbalanced data. SIGKDD Explorations: Newsletter of the Special Interest Group (SIG) on Knowledge Discovery & Data Mining, 6(1), 80–89. doi:10.1145/1007730.1007741

[11] Gu, Q., Li, Z., & Han, J. (2012). Generalized Fisher score for feature selection. doi:10.48550/ARXIV.1202.3725

[12] Huang, H. (2024). Feature Extraction and Classification of Text Data by Combining Two-stage Feature Selection Algorithm and Improved Machine Learning Algorithm. *Informatica*, *48*(8). https://doi.org/10.31449/inf.v48i8.5763

[13] Hamla, H., & Ghanem, K. (2024). A hybrid feature selection based on fisher score and SVM-RFE for microarray data. *Informatica*, *48*(1). https://doi.org/10.31449/inf.v48i1.4759

[14] Mafarja, M. M., & Mirjalili, S. (2017). Hybrid Whale Optimization Algorithm with simulated annealing for feature selection. Neurocomputing, 260, 302–312. doi: 10.1016/j.neucom.2017.04.053

[15] Jensen, R., & Shen, Q. (2004). Semantics-preserving dimensionality reduction: rough and fuzzy-rough-based approaches. IEEE Transactions on Knowledge and Data Engineering, 16(12), 1457–1471. doi:10.1109/tkde.2004.96

[16] Hedar, A.-R., Wang, J., & Fukushima, M. (2008). Tabu search for attribute reduction in rough set theory. Soft Computing, 12(9), 909–918. doi:10.1007/s00500-007-0260-1

[17] Bello, Rafael, Gomez, Y., Nowe, A., & Garcia, M. M. (2007, October). Two-step particle swarm optimization to solve the feature selection problem. Seventh International Conference on Intelligent Systems Design and Applications (ISDA 2007). Presented at the Seventh International Conference on Intelligent Systems Design and Applications (ISDA 2007), Rio de Janeiro, Brazil. doi:10.1109/isda.2007.4389688

[18] Wang, J., Li, T., & Ren, R. (2010, August). A real time IDSs based on artificial Bee Colony-support vector machine algorithm. Third International Workshop on Advanced Computational Intelligence. Presented at the 2010 Third International Workshop on Advanced Computational Intelligence (IWACI), Suzhou, China. doi:10.1109/iwaci.2010.5585107

[19] Panda, D., Panda, D., Dash, S. R., & Parida, S. (2021). Extreme Learning Machines with feature selection using GA for effective prediction of fetal heart disease: A Novel Approach. *Informatica*, *45*(3). https://doi.org/10.31449/inf.v45i3.3223

[20] Mafarja, M., & Abdullah, S. (2013). Record-to-record travel algorithm for attribute reduction in rough set theory. J Theor Appl Inf Technol, 49, 507–

513.https://doi.org/10.1080/00207721.2013.791000

[21] Kashef, S., & Nezamabadi-pour, H. (2015). An advanced ACO algorithm for feature subset selection. Neurocomputing, 147, 271–279. doi: 10.1016/j.neucom.2014.06.067

[22] Satapathy, S., Naik, A., & Parvathi, K. (2013). Rough set and teaching learning-based optimization technique for optimal features selection. Open Computer Science, 3(1). doi:10.2478/s13537-013-0102-4

[23] Satapathy, S. C., Naik, A., & Parvathi, K. (2013). Unsupervised feature selection using rough set and teaching learning-based optimisation. International Journal of Artificial Intelligence and Soft Computing, 3(3), 244. doi:10.1504/ijaisc.2013.053401

[24] Satapathy, S. C., & Naik, A. (2012). Hybridization of rough set and differential evolution technique for optimal features selection. In Advances in Intelligent and Soft Computing. Advances in Intelligent and Soft Computing (pp. 453–460). doi:10.1007/978-3-642-27443-5_52

[25] Zorarpacı, E., & Özel, S. A. (2016). A hybrid approach of differential evolution and artificial bee colony for feature selection. Expert Systems with Applications, 62, 91–103. doi: 10.1016/j.eswa.2016.06.004

[26] Emary, E., Zawbaa, H. M., & Hassanien, A. E. (2016). Binary grey wolf optimization approaches for feature selection. Neurocomputing, 172, 371–381. doi: 10.1016/j.neucom.2015.06.083

[27] Zawbaa, H. M., & Emary, E. (2018). Applications of flower pollination algorithm in feature selection and knapsack problems. In Studies in Computational Intelligence. Nature-Inspired Algorithms and Applied Optimization (pp. 217–243). doi:10.1007/978-3-319-67669-2_10

[28] Mirjalili, S. (2016). Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. Neural Computing & Applications, 27(4), 1053–1073. doi:10.1007/s00521-015-1920-1

[29] Mafarja, M., & Mirjalili, S. (2018). Whale optimization approaches for wrapper feature selection. Applied Soft Computing, 62, 441–453. doi: 10.1016/j.asoc.2017.11.006

[30] Mafarja, M. M., & Mirjalili, S. (2017). Hybrid Whale Optimization Algorithm with simulated annealing for feature selection. Neurocomputing, 260, 302–312. doi: 10.1016/j.neucom.2017.04.053

[31] Satapathy, S., & Naik, A. (2016). Social group optimization (SGO): a new population evolutionary optimization technique. Complex & Intelligent Systems, 2(3), 173–203. doi:10.1007/s40747-016-0022-8

[32] Naik, A., Satapathy, S. C., Ashour, A. S., & Dey, N. (2018). Social group optimization for global optimization of multimodal functions and data clustering problems. Neural Computing & Applications, 30(1), 271–287. doi:10.1007/s00521-016-2686-9

[33] Dey, N., Rajinikanth, V., Ashour, A., & Tavares, J. M. (2018). Social group optimization supported segmentation and evaluation of skin melanoma images. Symmetry, 10(2), 51. doi:10.3390/sym10020051

[34] Rajinikanth, V., & Satapathy, S. C. (2018). Segmentation of ischemic stroke lesion in brain MRI based on social group optimization and fuzzy-Tsallis entropy. Arabian Journal for Science and Engineering, 43(8), 4365–4378. doi:10.1007/s13369-017-3053-6

[35] Madhavi, G., & Harika, V. (2018). Implementation of Social Group Optimization to Economic Load Dispatch Problem. International Journal of Applied Engineering Research, 13, 11195–11200.

[36] Monisha, R., Mrinalini, R., Nithila Britto, M., Ramakrishnan, R., & Rajinikanth, V. (2019). Social group optimization and Shannon's function-based RGB image multi-level thresholding. In Smart Innovation, Systems and Technologies. Smart Intelligent Computing and Applications (pp. 123–132). doi:10.1007/978-981-13-1927-3_13

[37] Praveen, S. P., Rao, K. T., & Janakiramaiah, B. (2018). Effective allocation of resources and task scheduling in cloud environment using social group optimization. Arabian Journal for Science and Engineering, 43(8), 4265–4272. doi:10.1007/s13369-017-2926-z

[38] Phani Praveen S., & Rao, K. T. (2018). Client-awareness resource allotment and job scheduling in heterogeneous cloud by using social group optimization. International journal of natural computing research, 7(1), 15–31. doi:10.4018/ijncr.2018010102

[39] Naik, A., & Satapathy, S. C. (2021). A comparative study of social group optimization with a few recent optimization algorithms. Complex & Intelligent Systems, 7(1), 249–295. doi:10.1007/s40747-020-00189-6

[40] Naik, A., Satapathy, S. C., & Abraham, A. (2020). Modified Social Group Optimization—a meta-heuristic algorithm to solve short-term hydrothermal scheduling. Applied Soft Computing, 95(106524), 106524. doi: 10.1016/j.asoc.2020.106524

[41] Heidari, A. A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M., & Chen, H. (2019). Harris hawk's optimization: Algorithm and applications. Future Generations Computer Systems: FGCS, 97, 849–872. doi: 10.1016/j.future.2019.02.028

[42] S. Arora, S. Singh, (2018) Butterfly optimization algorithm: a novel approach for global optimization, Soft Comput. http://dx.doi.org/10.1007/ s00500-018-3102-4.

[43] Jain, Mohit, Singh, V., & Rani, A. (2019). A novel nature-inspired algorithm for optimization: Squirrel search algorithm. Swarm and

Evolutionary Computation, 44, 148–175. doi: 10.1016/j.swevo.2018.02.013
[44] Nematollahi, A. F., Rahiminejad, A., & Vahidi, B. (2020). A novel meta-heuristic optimization method based on golden ratio in nature. Soft Computing, 24(2), 1117–1151. doi:10.1007/s00500-019-03949-w
[45] Moghdani, Reza, & Salimifard, K. (2018). Volleyball premier league algorithm. Applied Soft Computing, 64, 161–185. doi: 10.1016/j.asoc.2017.11.043
[46] Mirjalili, S., & Hashim, S. Z. M. (2012). BMOA: Binary magnetic optimization algorithm. International Journal of Machine Learning and Computing, 204–208. doi:10.7763/ijmlc. 2012.v2.114
[47] Mirjalili, S., & Lewis, A. (2013). S-shaped versus V-shaped transfer functions for binary Particle Swarm Optimization. Swarm and Evolutionary Computation, 9, 1–14. doi: 10.1016/j.swevo.2012.09.002
[48] Emary, E., Zawbaa, H. M., & Hassanien, A. E. (2016). Binary grey wolf optimization approaches for feature selection. Neurocomputing, 172, 371–381. doi: 10.1016/j.neucom.2015.06.083
[49] Hastie, T., Tibshirani, R., & Friedman, J. (2003). The elements of statistical learning (1st ed.). doi:10.1007/978-0-387-21606-5
[50] Derrac, J., García, S., Molina, D., & Herrera, F. (2011). A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm and Evolutionary Computation, 1(1), 3–18. doi: 10.1016/j.swevo.2011.02.002
[51] Kennedy, J., & Eberhart, R. (2002). Particle swarm optimization. Proceedings of ICNN'95 - International Conference on Neural Networks. Presented at the ICNN'95 - International Conference on Neural Networks, Perth, WA, Australia. doi:10.1109/icnn.1995.488968
[52] Yang, Xin She, & Deb, S. (2010). Engineering optimisation by cuckoo search. International Journal of Mathematical Modelling and Numerical Optimisation, 1(4), 330. doi:10.1504/ijmmno.2010.035430
[53] Geem, Z., & Yang, X. S. (2009). Harmony search as a metaheuristic algorithm in: Music-Inspired Harmony Search Algorithm (pp. 1–14). Berlin/Heidelberg: Springer. doi:10.1007/978-3-642-00185-7
[54] Rizk-Allah, R. M., & Hassanien, A. E. (2018). New binary bat algorithm for solving 0–1 knapsack problem. Complex & Intelligent Systems, 4(1), 31–53. doi:10.1007/s40747-017-0050-z
[55] Rao, R. V., Savsani, V. J., & Vakharia, D. P. (2011). Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems. Computer Aided Design, 43(3), 303–315. doi: 10.1016/j.cad.2010.12.015
[56] Sayed, G. I., Hassanien, A. E., & Azar, A. T. (2019). Feature selection via a novel chaotic crow search algorithm. Neural Computing & Applications, 31(1), 171–188. doi:10.1007/s00521-017-2988-6
[57] Saremi, S., Mirjalili, S., & Lewis, A. (2017). Grasshopper optimisation algorithm: Theory and application. Advances in Engineering Software (Barking, London, England: 1992), 105, 30–47. doi: 10.1016/j.advengsoft.2017.01.004
[58] Mirjalili, S., Mirjalili, S. M., & Hatamlou, A. (2016). Multi-Verse Optimizer: a nature-inspired algorithm for global optimization. Neural Computing & Applications, 27(2), 495–513. doi:10.1007/s00521-015-1870-7
[59] Mirjalili, S., Gandomi, A. H., Mirjalili, S. Z., Saremi, S., Faris, H., & Mirjalili, S. M. (2017). Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems. Advances in Engineering Software (Barking, London, England: 1992), 114, 163–191. doi: 10.1016/j.advengsoft.2017.07.002
[60] S. Mirjalili. (2016) SCA: A sine cosine algorithm for solving optimization problems. Knowledge-Based Systems, 96 120–133. https://doi.org/10.1016/j.knosys.2015.12.022

Abbreviations

Feature selection= FS
Wilcoxon's rank sum= WRS
Fitness measure=FM
Classification accuracy=CA
Number of selected features= NSF