Neural Computing and Applications (2023) 35:3307–3324
https://doi.org/10.1007/s00521-022-07852-8

ORIGINAL ARTICLE

Enhanced feature selection technique using slime mould algorithm: a case study on chemical data

Ahmed A. Ewees[1,2] · Mohammed A. A. Al-qaness[3] · Laith Abualigah[4,5] · Zakariya Yahya Algamal[6,7] · Diego Oliva[8] · Dalia Yousri[9] · Mohamed Abd Elaziz[10,11,12,13]

Received: 21 December 2021 / Accepted: 16 September 2022 / Published online: 9 October 2022

Abstract
Feature selection techniques are considered one of the most important preprocessing steps, which has the most significant influence on the performance of data analysis and decision making. These FS techniques aim to achieve several objectives (such as reducing classification error and minimizing the number of features) at the same time to increase the classification rate. FS based on Metaheuristic (MH) is considered one of the most promising techniques to improve the classification process. This paper presents a modified method of the Slime mould algorithm depending on the Marine Predators Algorithm (MPA) operators as a local search strategy, which leads to increasing the convergence rate of the developed method, named SMAMPA and avoiding the attraction to local optima. The efficiency of SMAMPA is evaluated using twenty datasets and compared its results with the state-of-the-art FS methods. In addition, the applicability of SMAMPA to work with real-world problems is evaluated by using it as a quantitative structure-activity relationship (QSAR) model. The obtained results show the high ability of the developed SMAMPA method to reduce the dimension of the tested datasets by increasing the prediction rate. In addition, it provides results better than other FS techniques in terms of performance metrics.

Keywords Slime mould algorithm · Marine predators algorithm · Optimization feature selection · Quantitative structure-activity relationship (QSAR)

1 Introduction

The rapid growth of computer applications and information technologies produces a tremendous amount of data generated from various devices. The vast amount of data causes a critical problem for data mining which requires implementing practical data pre-processing steps using different techniques. Pr-processing is a necessary step that is employed to prepare and clean the data for the subsequent processing steps of the machine learning [1, 2]. Feature selection (FS) is an essential pre-processing step which is employed to reduce the size of the dataset. It is employed to select a small subset of the relevant features that capture the characteristics of the input data [3, 4]. Generally, FS methods remove noisy, unnecessary, and repeated features. Thus, an effective FS technique can boost the efficiency of data mining applications and various machine learning classification applications [5]. In general, FS methods can be classified into two types, wrapper-based and filter-based [6]. The wrapper-based techniques usually apply a classifier to obtain features, whereas the filter-based methods use data-reliant specifications to evaluate the merits of the features [6, 7]. Therefore, filter-based methods are more effective due to their fast implementation because they do not require classifiers to be involved in the FS process. To obtain a subset of features, we face some challenges. Thus, different search methods are applied to find the best features, including depth search, breadth search, random search, and hybrid search. However, exhaustive search requires a long time for extensive data, which is considered time-consuming.

Recently, with the great developments of the metahuristics (MH) optimization algorithms, that inspired from nature, various optimization problems, including FS can be solved using these MH algorithms. In the literature, different MH algorithms have been employed for this

Extended author information available on the last page of the article

purpose, such as particle swarm optimization (PSO) [8], genetic algorithm (GA) [9], artificial bee colony (ABC) [10], firefly algorithm (FA) [11], grey wolf algorithm (GOA) [12], sine cosine algorithm (SCA) [13], salp swarm algorithm [14], multi-verse optimizer (MVO) [15], Arithmetic Optimization Algorithm (AOA) [16], and others [17, 18]. However, individual MH algorithms may face severe limitations, such as slow convergence and trapping at local optima. Therefore, the hybridization concept has recently been implemented to overcome these limitations. This concept is performed by combining the operators of two MH algorithms to leverage their proprieties and advantages and avoid their shortcomings. Thus, in literature, we can find various hybrid MH methods for FS, such as a hybrid of PSO and SSA [19], differential evaluation (DE) and ABC [20], GOA and crow search algorithm (CSA) [21], DE and SCA [22], moth flame optimization (MFO) and DE [23], SSA and SCA [24], and many other hybrid MH methods [25].

Following the concept of the MH hybridization, this study proposes a new and efficient FS technique using a modified version of the slim mould algorithm (SMA) by the marine predators algorithm (MPA). The SMA was developed by [26], as a new MH optimizer that can be utilized to solve various optimization problems. The oscillation mode of slime mould inspires it in nature. More so, it is adopted to solve several optimization problems in literature, such as finding optimal parameters in energy applications [27, 28], air quality forecasting [29], and other engineering applications [30–32]. In addition, MPA is recently proposed by [33] by simulating the conduct of the marine prey and predators. It has received wide attention due to its efficiency and it is adopted in various domains, for example, time series forecasting [34, 35], image segmentation [36], medical image classification [37], parameter estimation [38], and other applications [39, 40].

However, the SMA performance requires more improvements, mainly when applied to real-world applications, which motivated us to develop a new version of SMA to improve its local search process using the operators of the MPA. The main aim of using MPA operators is to enhance the exploitation ability of SMA during the process of finding the optimal solution inside the feasible region. MPA is applied as a local search method since it has been established its performance in several applications, including forecasting cases of COVID-19 [35], and photovoltaic array reconfiguration [41].

The contribution of this study can be summarized as follows:

- Develop a feature selection technique using an enhancement version of the SMA.
- Boost the capability of the local search of the SMA using the operators of MPA.
- Assess the efficiency of the SMAMPA developed method by using a set of twenty UCI datasets and comparing it with other FS methods.
- Verify the applicability of the SMAMPA by implementing it with real-world applications, such as QSAR model.

The structure of this study is as follows. The related works are presented in Sect. 2, where the preliminaries of the applied techniques, SMA, and MPA are described in Sect. 3. In Sect. 4, we describe the proposed SMAMPA approach, and in Sect. 5, the experimental evaluation is presented, including different benchmark datasets and comparisons to existing methods. Finally, the conclusions and future direction are highlighted in Sect. 6.

2 Related works

In this section, we summarize a number of the existing FS methods based on modified and improved optimization algorithms proposed in recent years. In [4], a modified version of the ABC algorithm, called a binary ABC, is proposed for FS. The searchability of the ABC is improved using the evolutionary-based similarity search mechanism, which is integrated into the existing binary ABC variants. It was evaluated using several datasets and compared to the original PSO and ABC besides several modified versions of PSO and ABC. In [42], the authors suggested an FS method based on a hybrid of the Flower Pollination Algorithm (FPA) and Clonal Selection Algorithm (CSA). The proposed BCFA was evaluated using the optimum-path forest classifier, and it showed significant performance with three different datasets. Also, It showed better performance in comparison to several optimization methods.

In [43], two binary variants of the whale optimization algorithm (WOA) were proposed for FS. The first variant is implemented by improving the search process using Tournament and Roulette Wheel selection mechanisms. In the second variant, the exploitation of the whale optimization algorithm is improved by using crossover and mutation operators. Sayed et al. [44] proposed a chaotic crow search algorithm (CSA) to overcome the limitations of the original CSA, such as trapping at local optima and low convergence rate. The new modified version, CCSA, was applied as an FS method evaluated using twenty datasets. The CCSA also was compared to different optimization techniques, and it achieved superior performance against several previous FS methods.

The authors in [45] suggested two binary versions of butterfly optimization algorithm (BOA) for FS. They used

two transfer functions for mapping continuous search spaces to discrete ones. Several UCI benchmark datasets were used to evaluate the proposed method. More so, wide comparisons to some existing FS methods were performed. Evaluation outcomes showed the superior performance of the BOA. Too and Abdullah [46] proposed an FS method using a new variant of the genetic algorithm (GA) and a fast rival GA. They applied a competition strategy to combine crossover schemes and the new selection to boost the global search ability of the GA. Twenty-three UCI benchmark datasets were utilized to test the performance of the modified GA.

Zhang et al. [47] presented an improved variant of the Harris hawks optimization algorithm, called IHHO for FS. The main idea of the IHHO is by applying the salp swarm algorithm to enhance the search ability of the HHO. Several UCI datasets were used to evaluate the IHHO, and it achieved competitive performance compared to several FS methods. Another modified HHO, called Chaotic HHO (CHHO), is proposed for FS by Elgamal et al. [48]. Chaotic maps are applied to improve the population diversity of the HHO in the search space. Moreover, simulated annealing (SA) is applied to the best solution to enhance the exploitation of the HHO. They used Fourteen datasets to evaluate the CHHO compared to several optimization algorithms. Overall results showed that CHHO got the best outcomes.

The authors of [49] proposed a FS method, called ECSA, using a modified version of the crow search algorithm (CSA). The authors proposed three modifications to the traditional CSA to enhance its search capability. Sixteen UCI benchmark datasets were applied to evaluate the ESCA compared to the traditional CSA and several existing FS methods. The ESCA showed competitive performance in all experiments. Too and Mirjalili [6] suggested an FS method called hyper learning binary dragonfly algorithm. They applied a hyper learning strategy to improve the binary dragonfly algorithm, to avoid its limitations, such as trapping at local optima. They evaluate the proposed method using different UCI datasets and a new COVID-19 dataset. Zhong et al. [7] proposed a new FS method based on a modified Tree Growth Algorithm (TGA). A binary TGA is applied for FS applications, and also the evolutionary population dynamic strategy is employed to enhance the search capability of the TGA. Different UCI benchmark datasets were utilized to test the TGA performance.

Several works from the previous review were conducted for addressing FS problems by developing new methods to overcome the drawbacks of the algorithms' original versions using benchmark and real datasets. The proposed methods showed good abilities to escape getting trapped in local optima, improve the convergence rate, and improve population diversity. However, there is no optimization technique to solve all problems, as stated by the No-Free-Lunch (NFL) theorem. Accordingly, this paper proposes a new optimization method by improving the slime mould algorithm's local search ability using the MPA operators to solve different feature selection problems using benchmark and real datasets. This improvement can help balance the search methods and avoid local search problems such as traping in a local optimum and degrading the convergence rate.

3 Background

This section presents the basic definitions of the SMA and MPA, as in what follows.

3.1 Slime mould algorithm

The SMA was firstly introduced by [26] as a novel optimization mechanism for global optimization. The SMA simulates the natural behaviour of the slime mould's oscillation. The mathematical formulation of SMA is given as:

1. Phase 1 (The food approach): This step models the approach for the slime mould. The following equation describes this phase:

$$Z = \begin{cases} Z_b + v_b.(W.Z_A - Z_B) & r < p \\ v_c.Z & r \geq p \end{cases} \quad (1)$$

where v_b is defined in the range of $[-a, a]$ and v_c decreases from 1 to 0. Z_b corresponds to the best solutions. Additionally, Z_A and Z_B are two solutions selected from a randomly, whereas W represents the mould weight of the slime. While p is computed as:

$$p = \tanh|S(i) - DF|, \; i = 1, 2, ..., N \quad (2)$$

From Eq. 2, $S(i)$ corresponds to the fitness values of the Z solution. DF is the best fitness value. The value a that defines v_b in Eq. 1 is computed as:

$$a = arctanh\left(-\left(\frac{t}{max_t}\right) + 1\right) \quad (3)$$

where, t is the current iteration. max_t is the maximum number of iteration. Also, the value of W is obtained as follows:

$$W(S_{Ind}(i)) = \begin{cases} 1 + r\log((b_F - S(i))/(b_F - w_F) + 1) & Cond \\ 1 - r\log((b_F - S(i))/(b_F - w_F) + 1) & otherwise \end{cases} \quad (4)$$

in which *Cond* denotes that $S(i)$ ranks first half of the population. More so, $r \in [0, 1]$ is randomly generated. b_F and w_F and b_F represent the best and worst fitness values, respectively. Finally, S_{Ind} stores the sorted fitness values, as defined in the following formula:

$$S_{Ind} = sort(S) \quad (5)$$

2. Phase 2 (Wrap food): in this step, SMA imitates the updating position of the slime mould. The following equation is applied to compute this update.

$$Z^* = \begin{cases} rand(UB - LB) + LB & rand < z \\ Z_b(t) + v_b(WZ_A(t) - Z_B(t)) & r < p \\ v_c Z(t) & r \geq p \end{cases} \quad (6)$$

where *LB* and *UB* represent the lower and upper bounds of the search space, respectively. r and *rand* are obtained from a random distribution between [0, 1].

3. Phase 3 (Oscillation): at this step the value of v_b is updated within $[-a, a]$ and v_c inside $[-1, 1]$.

3.2 Marine predators algorithm

The MPA is a global optimization mechanism introduced in [33]. The MPA mimics the elements of marine prey and predators during hunting. As other metaheuristics, the MPA begins by taking random solutions from the search space as in Eq. 7

$$Z = LB + rand \times (UB - LB) \quad (7)$$

where, *rand* a random variable is generated in the range [0,1]. *LB* and *UB* are the upper and lower bounds that define the search space. Once the candidate solutions are generated, two matrices (named Elite matrix, which contains the fitness values and prey matrix) are formulated as:

$$Elite = \begin{bmatrix} Z_{11}^1 & Z_{12}^1 & \ldots & Z_{1d}^1 \\ Z_{21}^1 & Z_{22}^1 & \ldots & Z_{2d}^1 \\ \ldots & \ldots & \ldots & \ldots \\ Z_{n1}^1 & Z_{n2}^1 & \ldots & Z_{nd}^1 \end{bmatrix}, z = \begin{bmatrix} Z_{11} & Z_{12} & \ldots & Z_{1d} \\ Z_{21} & Z_{22} & \ldots & Z_{2d} \\ \ldots & \ldots & \ldots & \ldots \\ Z_{n1} & Z_{n2} & \ldots & Z_{nd} \end{bmatrix}, \quad (8)$$

The three phases of MPA modify the candidate solution using the velocity ratio of the predator and prey. Each step of the MPA is described below.

1. Phase 1 (High-velocity ratio): here, the prey is extremely fast, then the predator decides to be quiet and not move. This phase occurs at the beginning of the optimization process, and the movement of the prey is modeled as follows:

$$S_i = R_B \times (Elite_i - R_B \times Z_i), i = 1, 2, ..., N \quad (9)$$

$$Z_i = Z_i + P \times R \times S_i \quad (10)$$

in which $R \in [0, 1]$ refers to a vector of random numbers $P = 0.5$, and R_B is Brownian motion vector.

2. Phase 2 (Unit velocity ratio): at this phase, the velocity of the prey and the predator is the same. This case is present in half of the iterative procedure. Here, the predator updates his position using Brownian movements, and the prey uses lévy flights. In this phase, Z is divided into two parts, and to update the solution in the first part; it applies Eqs. (11)-(12) and the second one uses Eq. (13)-(14).

$$S_i = R_L \times (Elite_i - R_L \times Z_i), i = 1, 2, ..., N \quad (11)$$

$$Z_i = Z_i + P \times R \times S_i \quad (12)$$

where R_L is generated randomly by a Lévy distribution.

$$S_i = R_B \times (R_B \times Elite_i - Z_i), i = 1, 2, ..., N \quad (13)$$

$$Z_i = Elite_i + P \times CF \times S_i, \quad CF = \left(1 - \frac{t}{max_t}\right)^{2\frac{t}{max_t}} \quad (14)$$

From Eqs. 13 and 14 the values of t and max_t are the current and total number of iterations, respectively.

3. Phase 3 (low-velocity ratio): Within this phase, the predator has velocity faster than the prey, which occurred in the last third of the updating process using Eq. (15)

$$S_i = R_L \times (R_L \times Elite_i - Z_i), i = 1, 2, ..., N \quad (15)$$

$$Z_i = Elite_i + P \times CF \times S_i, \quad (16)$$

According to [33] the MPA has another two key points.

– The first one is related to the Eddy formation and the effect of fish aggregating devices (FADS) that can modify the behavior of the predators. The MPA employs the following equation to handle these situations:

$$Z_i = \begin{cases} Z_i + CF[Z_{min} + R \times (Z_{max} - Z_{min})] \times U & r_5 < FAD \\ Z_i + [FAD(1 - r) + r](Z_{r1} - Z_{r2}) & r_5 > FAD \end{cases} \quad (17)$$

From Eq. 17 U refers to a binary vector. $FAD = 0.2$. $r \in [0, 1]$. r_1 and r_2 denote random prey.

– The second one is the marine memory, here Z remembers its position, so, this behavior gives MPA

ability to save the previous Z_b. This solution is used and compared with the new Z_b.

4 The SMAMPA method

The SMAMPA is described in this section. It applies both SMA and MPA algorithms to improve its performance. In this context, the MPA applies as a local search of the original version of SMA to improve its ability to solve optimization problems. This improvement adds more flexibility to the method to explore the search space and improve diversity.

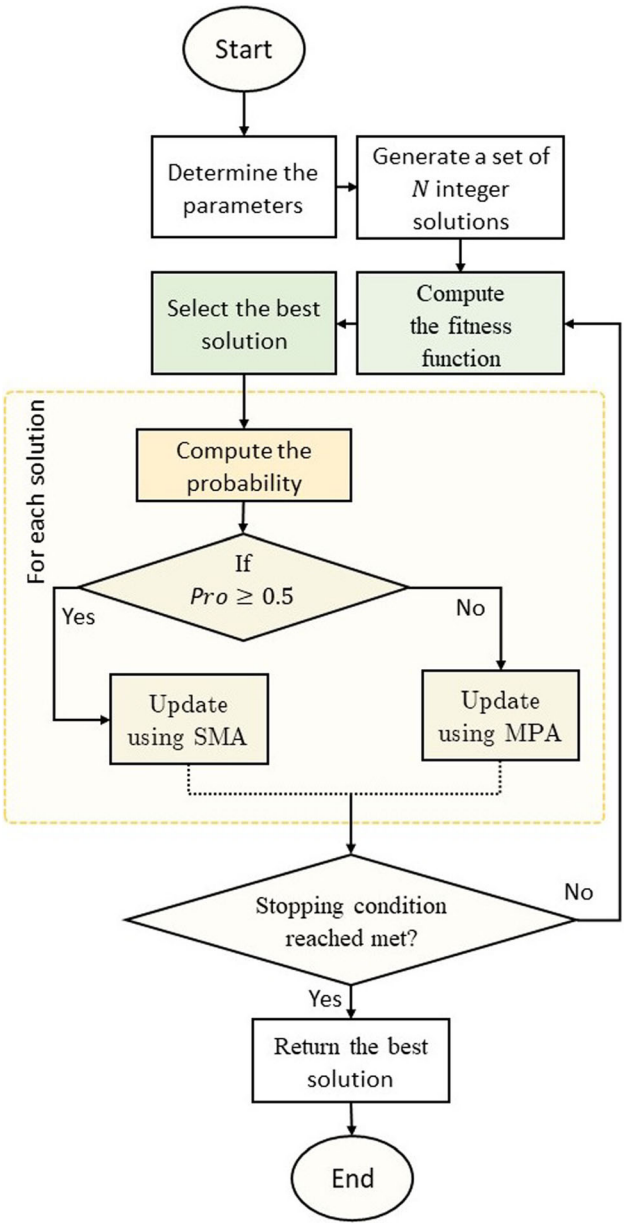

Fig. 1 The SMAMPA structure and workflow

The basic structure of the SMAMPA is shown in Fig. 1. It starts by defining the parameters and creating the search space by initialling the problem population. After this step, the best solution is determined and saved by evaluating the fitness function. Furthermore, each solution is updated by either the SMA or MPA algorithms; this switching is based on the quality of the fitness function value; the quality is calculated as in Equation 19. Therefore, if the probability of the solution is more significant than α, the solution will be updated by SMA, else it will be updated by MPA. In this paper, the probability value (α) is set to 0.5. These steps are iterated for all solutions; then, the best solution, among all solutions, is selected. This sequence loops till reaching the stop condition, then the final results are presented. In detail, the SMAMPA begins by initializing the parameters of both SMA and MPA. Then the SMA generates a Z $[x_i, i = 1, 2, .., X_N]$ random binary population with N and D size and dimension. Then, the first fitness values are computed by the operators of the SMA. The following equation is used to calculate the fitness function value Eq. (18):

$$f(x_i(t)) = \xi E_{x_i(t)} + (1 - \xi)(\frac{|x_i(t)|}{|C|}) \tag{18}$$

where $E_{x_i(t)}$ defines the classification error (in this study we use kNN as a classifier). $\xi \in [0, 1]$ balances between the classification error and the number of the selected features. The proposed method calculates the probability (Pro_i) by Eq. 19 to update the solution by the operators of MPA or SMA (i.e., if $Pro_i > 0.5$ the SMA will be used else, MPA will be used)

$$Pro_i = \frac{F_i}{\sum_{i=1}^{N} F} \tag{19}$$

where, f is the values of the fitness function. These sequences are iterated until meeting the stop condition. In the final step, the best solution is presented as the output of the proposed method.

5 Experiment results and discussion

5.1 Performance metrics

Minimum (Min) result and maximum (Max) result of the fitness value are applied using Eqs. 20 and 21, respectively.

$$Min = \min_{1 \le k \le N} F_i \tag{20}$$

$$Max = \max_{1 \le k \le N} F_i \tag{21}$$

where F is the fitness function values

Accuracy: It is used to compute the classification accuracy in the experiments. It is calculated using Eq. 22.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{22}$$

where TP and TN define true positive and true negative. FP and FN define false positive and false negative.

Standard deviation (Std): It is computed using Eq. 23. It evaluates the stability of the algorithms. The results of the fitness function are used to compute this measure ($\overline{F}$ is the mean of F).

$$Std = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(F_k - \overline{F})^2} \tag{23}$$

5.2 Compared techniques and parameter settings

The SMAMPA is evaluated and compared to nine recently published metaheuristic algorithms (i.e., MPA, GA, SMA, PSO, HHO, SSA, MFO, WOA, and GOA) in the fitness values (i.e., minimum and maximum), standard deviation, accuracy, classification accuracy, and computational time. The SMAMPA method is also compared with eight advanced metaheuristic algorithms (i.e., BDA [50], BSSAS3 [14], bGWO2 [12], GLR [51], SbBOA [45], BGOAM [52], Das [53], and S-bBOA [45]).

The parameters setting of these algorithms is identical to that declared in their original studies. Table 1 presents the settings of the parameters of all applied methods. The MATLAB 2015a executes all the algorithms. All methods run on a 16GB RAM Intel Core i7 1.8 GHz 2.3 GHz processor. The solution numbers applied in this paper are set to to 30. The maximum iteration number is set to 500. Each competitor algorithm is applied 30 independent runs and the average of its results are presented in the tables.

5.3 Experiment series 1: UCI datasets

In this section, twenty benchmark datasets are tested to demonstrate the SMAMPA optimizer's efficiency. These datasets were taken from the Machine Learning Repository (UCI) [61]. Table 2 shows the tested datasets that contain different numbers of features, number of instances, and number of classes. The applied datasets are collected from different areas, including biology, games, physics, and biomedical.

The results obtained by the given SMAMPA method in the average measure of the fitness function, as stated in

Table 1 Parameters setting of the applied methods

No.	Algorithm	Reference	Parameter	Value
1	MPA	[33]	γ	γ>1
			P	0.0
2	SMA	[26]	z	0.01
3	GA	[54]	Selection	Roulette wheel (Proportionate)
			Crossover	Whole arithmetic
			Probability	0.8,
			α	[-0.5, 1.5])
4	HHO	[55]	α	1.5
5	PSO	[56]	Topology	Fully connected
			Cognitive and social constant	(C1, C2) 2, 2
			Inertia weight	Linear reduction values [0.9 0.1]
			Velocity limit	10% of dimension range
6	SSA	[57]	v_0	0
7	WOA	[58]	α	Decreased from 2 to 0
			b	2
8	MFO	[59]	Convergence constant a	[-2 -1]
			Spiral factor b	1
9	GOA	[60]	Attraction distance	2.079 to 4
			l	1.5
			f	0.5
			c_max	1
			c_min	0.00001

Table 2 The details descriptions of the used UCI datasets

Name	Features	Instances	Classes	Type
breastWDBCD	30	569	2	Biology
ionosphereD	34	351	2	Physical
wineD	13	178	3	Chemistry
breastcancerD	9	699	2	Biology
sonarD	60	208	2	Biology
glassD	9	214	7	Physics
tic-tac-toeD	9	958	2	Game
LymphographyD	18	148	2	Biology
waveformD	40	5000	3	Physics
clean1dataD	166	476	2	Artificial
ZooD	16	101	6	Artificial
SPECTD	22	267	2	Biology
ecoliD	7	336	8	Biology
CongressEWD	16	435	2	Politics
M-of-nD	13	1000	2	Biology
ExactlyD	13	1000	2	Biology
Exactly2D	13	1000	2	Biology
VoteD	16	300	2	Politics
heartD	13	270	2	Biology
krvskpD	36	3196	2	Game

(18), are recorded in Table 3. SMAMPA is observed to beat the other comparative well-known methods in 85% of the tested datasets. PSO algorithm is the second-best method. SMAMPA got better performance than other comparative methods for all tested datasets except Sonar, ExactlyD, and krvskpD datasets. According to the average fitness values measure, the results demonstrated that the given SMAMPA has a promising ability in addressing this kind of problem.

The results are given by the introduced SMAMPA in terms of minimum fitness values, as stated in Eq. (20), are recorded in Table 4. SMAMPA is observed to defeat the other comparative well-known methods in 75% of the tested datasets. PSO algorithm is the second-best method. Based on minimum fitness values, SMAMPA has achieved the minimum fitness values with promising results for most datasets compared to other rival algorithms. It got better results in almost all the tested datasets except glassD, WaveformD, SpectD, Exactly2D, and krvskpD datasets. The results confirmed that the proposed SMAMPA could solve different feature selection challenges according to the minimum fitness values.

The results achieved by the SMAMPA for the maximum fitness values, as declared in Eq. (21), are shown in Table 5. SMAMPA is recognized to overcome the other

Table 3 Results of the fitness values measure

	SMAMPA	MPA	SMA	GA	HHO	PSO	SSA	WOA	MFO	GOA
breastWDBCD	**0.0925**	0.181	0.3409	0.1250	0.1194	0.1049	0.1118	0.1404	0.1733	0.2134
ionosphereD	**0.1537**	0.279	0.4061	0.2340	0.2144	0.1885	0.2276	0.2442	0.2750	0.3253
wineD	**0.0000**	0.015	0.1715	0.0151	0.0058	0.0038	0.0566	0.0207	0.1259	0.1514
breastcancerD	**0.1543**	0.26	0.4009	0.2153	0.1924	0.1669	0.2093	0.2193	0.2604	0.3268
glassD	**0.1347**	0.146	0.2226	0.1521	0.1428	0.1409	0.1508	0.1501	0.1854	0.2197
sonarD	0.1196	0.275	0.4116	0.2075	0.1951	**0.1133**	0.1839	0.2409	0.2717	0.3255
LymphographyD	**0.2576**	0.41	0.5342	0.3557	0.3015	0.2767	0.3168	0.3590	0.4508	0.5160
tic-tac-toeD	**0.0000**	0.094	0.5068	0.1666	0.0018	0.0079	0.0223	0.0237	0.4428	0.5172
waveformD	**0.6337**	0.681	0.9037	0.6513	0.6568	0.6346	0.6499	0.6598	0.6736	0.7360
clean1dataD	**0.1920**	0.285	0.4374	0.2569	0.2627	0.2242	0.2677	0.2962	0.2716	0.3439
SPECTD	**0.3078**	0.406	0.4791	0.3695	0.3528	0.3355	0.3571	0.3814	0.4045	0.4789
ZooD	**0.0000**	0.004	0.1878	0.0150	0.0033	0.0042	0.0483	0.0078	0.0901	0.1220
ecoliD	**0.1998**	0.21	0.3398	0.2235	0.2171	0.2169	0.2252	0.2208	0.2764	0.3337
CongressEWD	**0.1132**	0.274	0.4035	0.1842	0.1645	0.1363	0.1812	0.1775	0.2308	0.3025
ExactlyD	0.0050	0.292	0.5858	0.1897	0.0539	**0.0000**	0.0576	0.2399	0.4333	0.5944
Exactly2D	**0.4801**	0.504	0.5699	0.5048	0.4929	0.4884	0.5081	0.4956	0.5447	0.5816
M-of-nD	**0.0000**	0.335	0.4790	0.1419	0.0383	**0.0000**	0.0388	0.1382	0.3096	0.4955
VoteD	**0.1250**	0.196	0.4115	0.2015	0.1727	0.1626	0.1871	0.1973	0.2595	0.3431
krvskpD	0.1193	0.258	0.5281	0.1954	0.1752	**0.1192**	0.1718	0.2022	0.1578	0.3534
heartD	**0.3408**	0.368	0.5425	0.3794	0.3575	0.3471	0.3617	0.3804	0.4255	0.4969

Bold values indicate the best result

Table 4 Min measure results

MIN	SMAMPA	MPA	SMA	GA	HHO	PSO	SSA	WOA	MFO	GOA
breastWDBCD	**0.000**	**0.000**	0.084	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	0.119	0.119
ionosphereD	**0.000**	0.151	0.213	0.107	0.107	0.107	0.107	0.151	0.151	0.185
wineD	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
breastcancerD	**0.000**	0.107	0.185	0.107	**0.000**	0.107	0.107	0.107	0.107	0.213
glassD	0.117	0.113	0.099	**0.087**	**0.087**	**0.087**	**0.087**	0.095	0.117	0.129
sonarD	**0.000**	**0.000**	0.277	0.139	**0.000**	**0.000**	**0.000**	**0.000**	0.139	0.196
LymphographyD	**0.000**	0.164	0.368	0.232	0.164	0.164	0.232	0.285	0.232	0.285
tic-tac-toeD	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	0.259	**0.000**
waveformD	0.593	**0.593**	0.692	0.611	0.625	0.611	0.601	0.617	0.597	0.631
clean1dataD	**0.092**	0.225	0.275	0.159	0.159	0.159	0.183	0.225	0.159	0.243
SPECTD	0.212	0.299	0.273	0.244	**0.173**	0.212	0.273	0.273	0.273	0.367
ZooD	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
ecoliD	**0.142**	**0.142**	0.194	0.174	0.159	0.159	0.159	0.175	0.190	0.206
CongressEWD	**0.000**	0.096	0.192	0.096	0.096	**0.000**	0.096	0.096	0.096	0.096
ExactlyD	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
Exactly2D	0.447	0.465	0.465	**0.443**	**0.443**	**0.443**	**0.443**	**0.443**	0.465	0.529
M-of-nD	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
VoteD	**0.000**	**0.000**	0.163	**0.000**	0.115	**0.000**	**0.000**	0.115	0.115	0.163
krvskpD	0.087	0.162	0.221	0.142	0.137	**0.079**	0.132	0.137	0.100	0.203
heartD	**0.273**	0.299	0.367	0.323	0.299	**0.273**	**0.273**	0.299	0.299	0.346

Bold values indicate the best result

Table 5 Results of the Max measure

MAX	SMAMPA	MPA	SMA	GA	HHO	PSO	SSA	WOA	MFO	GOA
breastWDBCD	**0.1454**	0.6167	0.6766	0.2056	0.1876	0.1876	0.2374	0.2220	0.2654	0.3140
ionosphereD	**0.2384**	0.4885	0.5539	0.3371	0.3015	0.2611	0.3371	0.3536	0.4264	0.4767
wineD	**0.0000**	0.1508	0.4264	0.0754	0.0754	0.0754	0.2820	0.0754	0.2611	0.3454
breastcancerD	**0.2132**	0.4647	0.5741	0.3371	0.2611	0.2611	0.3198	0.3198	0.3989	0.4523
glassD	**0.1801**	0.1962	0.3059	0.1943	0.1903	0.1903	0.2664	0.2215	0.2855	0.3442
sonarD	**0.1961**	0.5718	0.5371	0.3922	0.3397	0.3397	0.3669	0.3669	0.4804	0.4599
LymphographyD	**0.3676**	0.7534	0.6778	0.4650	0.4027	0.4027	0.6367	0.4650	0.6778	0.7352
tic-tac-toeD	**0.0000**	0.6338	0.7516	0.4753	0.0647	0.2505	0.5786	0.4387	0.6501	0.7318
waveformD	**0.6548**	1.1335	1.1486	0.6888	0.6969	0.6573	0.6835	0.7116	0.7720	0.8686
clean1dataD	**0.2750**	0.4300	0.7101	0.3305	0.3667	0.3305	0.3889	0.3667	0.3430	0.4674
SPECTD	**0.3665**	0.5599	0.6802	0.4887	0.4887	0.4405	0.4732	0.5037	0.5325	0.5985
ZooD	**0.0000**	0.0471	0.4447	0.0577	0.0333	0.0333	0.2828	0.0471	0.2925	0.3636
ecoliD	**0.2631**	0.3489	0.5898	0.2819	0.2806	0.2806	0.3818	0.2806	0.3789	0.7703
CongressEWD	**0.1355**	0.6773	0.7103	0.2709	0.2534	0.2346	0.4064	0.2709	0.4064	0.5747
ExactlyD	0.0894	0.8509	0.7430	0.5762	0.3688	**0.0000**	0.5514	0.5477	0.6419	0.7266
Exactly2D	**0.5254**	0.6928	0.7071	0.5441	0.5441	0.5441	0.6419	0.5441	0.6000	0.7211
M-of-nD	**0.0000**	0.7975	0.6419	0.4147	0.3225	**0.0000**	0.5762	0.3633	0.6419	0.6261
VoteD	**0.1633**	0.4000	0.7916	0.2828	0.2309	0.2582	0.4000	0.3055	0.4761	0.4899
krvskpD	**0.1415**	0.7102	0.6896	0.2526	0.2209	0.1659	0.2399	0.2526	0.2293	0.5983
heartD	**0.3665**	0.4052	0.7228	0.4732	0.4405	0.4232	0.4405	0.4732	0.5464	0.6465

Bold values indicate the best result

comparative methods in 95% of the tested datasets. PSO method is also the second-best method. Except for the ExactlyD dataset, the proposed SMAMPA improved performance in all tested datasets than other comparative

approaches. The outcomes demonstrated that the proposed integration method between the SMA and MPA search processes has a powerful ability to trade with complicated feature selection problems.

Figure 2 displays the average, minimum, and maximum fitness values for the comparative methods overall used datasets. It can be seen that the developed SMAMPA reached the best results in terms of the three measures (i.e., average, minimum, and maximum fitness values). SMAMPA got the smallest values using all measures in the tested datasets, which is strong evidence regarding the ability of SMAMPA in solving the FS problems. The modification of the proposed method proved its searchability in finding better solutions than the original SMA and MPA, as well as, this modification got all the best outcomes compared to the comparative algorithms.

Table 6 displays each algorithm's accuracy measure values overall the used datasets. The proposed SMAMPA gathered the best high accuracy values in 95% of the tested datasets, pursued by PSO. However, the PSO obtained the best values in three datasets (i.e., ExactlyD, Exactly2, and M-of-n). In general, the SMAMPA exhibited an excellent ability to select the most vital features in the selection stage and produce the highest accuracy values in the classification stage. Figure 3 illustrates the average of the accuracy values for the all methods. We can recognise that the proposed method got the highest accuracy values compared to all comparative techniques; this supports our claim regarding the proposed SMAMPA; it works more efficiently than traditional methods and is also more efficient than other comparative algorithms. The second best method is the PSO algorithm; it got more reliable results than the rest of the comparison techniques in solving these widespread problems.

Table 7 displays each algorithm's Std measure of the fitness function assessments using all the given datasets. The proposed SMAMPA obtained stable results according to Std values in 50% of the tested datasets, pursued by PSO, WOA, HHO, GA, and finally, the MPA. This result declares that the SMAMPA's stability is better than other comparative methods according to its performance. The obtained results' distribution is excellent and smaller than other comparative methods overall, the tested datasets. Figure 4 illustrates the average of the Std of the fitness function values for all compared methods. We can see obviously that the suggested SMAMPA got the smallest Std values compared to all comparative techniques; this supports our claim regarding the performance of the proposed SMAMPA again; it achieves promising results compared to other methods by giving low distribution and similar outcomes across a wide range of executions. The following best method is the PSO algorithm, pursued by HHO.

Table 8 lists the numbers of the selected features for all the tested methods. Table 8 shows the shorter length of the obtained optimal subset of features acquired by the comparative techniques. Investigating the results, SMA produced the nominal feature size in ten datasets, pursued by SMAMPA (six datasets). Compared with MPA, SMA, GA, HHO, PSO, SSA, WOA, MFO, and GOA, the SMAMPA can typically find the nominal subset of selected features that can adequately represent the main idea, as shown in Fig. 5. Owing to the MPA method, SMAMPA can override the local optima problem and thoroughly recognize the most helpful feature selection solution.

According to the computational time given in Table 9 and Fig. 6, the proposed SMAMPA got comparable computational time to solve the given problems. The main important thing in these experiments to tackle the FS problem is the evaluation measures, like the accuracy, because the given problem needs to be solved one time and not more.

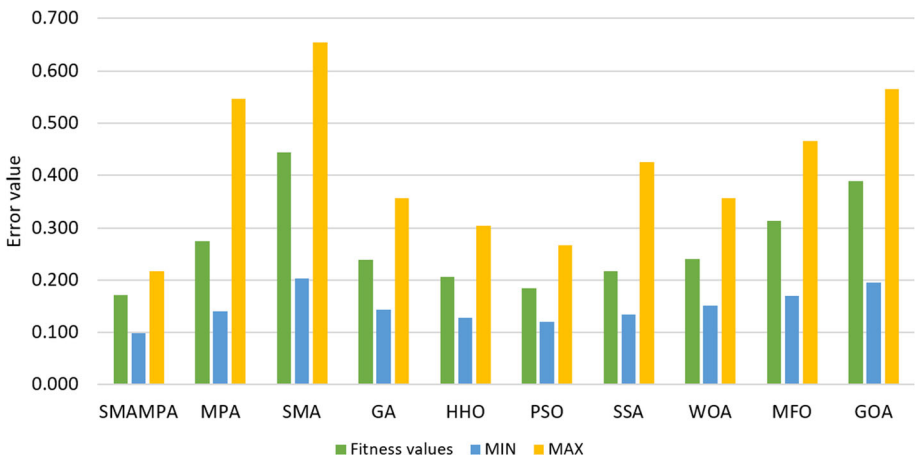

Fig. 2 Error values average for all algorithms

Table 6 Results of the Accuracy measure

ACC	SMAMPA	MPA	SMA	GA	HHO	PSO	SSA	WOA	MFO	GOA
breastWDBCD	**0.990**	0.949	0.856	0.982	0.984	0.988	0.985	0.979	0.968	0.952
ionosphereD	**0.974**	0.909	0.826	0.943	0.952	0.962	0.945	0.938	0.920	0.890
wineD	**1.000**	0.993	0.872	0.995	0.998	0.999	0.967	0.994	0.929	0.903
breastcancerD	**0.974**	0.923	0.829	0.952	0.961	0.970	0.954	0.949	0.928	0.890
glassD	**0.798**	0.636	0.657	0.624	0.663	0.687	0.677	0.613	0.579	0.515
sonarD	**0.989**	0.907	0.825	0.953	0.955	0.978	0.959	0.935	0.919	0.889
LymphographyD	**0.932**	0.397	0.708	0.703	0.570	0.752	0.729	0.654	0.495	0.364
tic-tac-toeD	**1.000**	0.941	0.683	0.939	1.000	0.998	0.990	0.992	0.779	0.696
waveformD	**0.794**	0.767	0.604	0.783	0.784	0.793	0.787	0.786	0.774	0.738
clean1dataD	**0.961**	0.917	0.800	0.933	0.929	0.948	0.926	0.911	0.924	0.879
SPECTD	**0.903**	0.669	0.758	0.859	0.874	0.885	0.870	0.853	0.832	0.766
ZooD	**1.000**	0.994	0.929	0.910	0.996	0.995	0.708	0.966	0.398	0.254
ecoliD	**0.862**	0.849	0.655	0.836	0.835	0.837	0.839	0.832	0.786	0.818
CongressEWD	**0.987**	0.896	0.832	0.964	0.971	0.978	0.962	0.966	0.942	0.897
ExactlyD	**1.000**	0.791	0.636	0.892	0.990	**1.000**	0.983	0.886	0.770	0.633
Exactly2D	0.465	0.325	0.636	0.737	0.750	**0.759**	0.729	0.746	0.699	0.660
M-of-nD	**1.000**	0.751	0.734	0.957	0.994	**1.000**	0.986	0.965	0.870	0.738
VoteD	**0.982**	0.955	0.808	0.956	0.969	0.969	0.960	0.959	0.927	0.875
krvskpD	**0.986**	0.913	0.707	0.961	0.969	0.985	0.970	0.958	0.974	0.862
heartD	**0.883**	0.864	0.681	0.854	0.871	0.878	0.868	0.853	0.815	0.747

Bold values indicate the best result

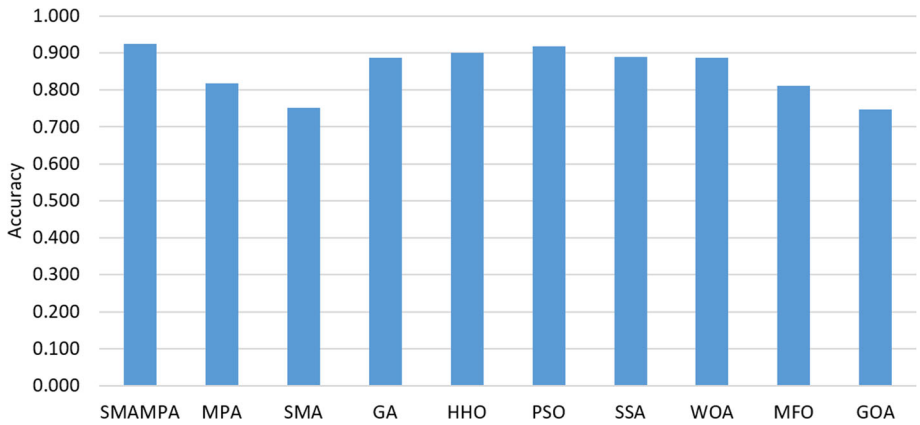

Fig. 3 Accuracy average for all algorithms

5.4 Comparison with the state-of-the-art

This part evaluates the SMAMPA and compares further with different advanced and well-known published methods in the literature. These methods are BDA [50], BSSAS3 [14], bGWO2 [12], GLR [51], SbBOA [45], BGOAM [52], Das [53], and S-bBOA [45].

Table 10 shows all the tested methods using versions benchmark datasets. The given values of the comparative methods in this table are taken from their original papers. The ''–'' sign denotes no given results for this case. The proposed SMAMPA obtained better results in 70% of the tested datasets according to given values. It got the most high-grade results in almost all the tested datasets except ionosphereD, BreastcancerD, LymphographyD, ExactlyD, Exactly2D, and VoteD. The following best method is BDA, which got the most beneficial results in 53%, as this method has results for 15 datasets, followed by BSSAS3.

Recap, SMAMPA has a more trustworthy exploration experience than other comparative optimization techniques. This result is confirmed because the other tested algorithms did not allow SMAMPA to investigate other search areas in the search regions. Moreover, this proved the proposed SMAMPA to sustain solutions heterogeneity remarkably better than other feature selection methods. Besides, SMAMPA always got superior fitness values than other algorithms, proving its ability to evade restricted optima. In comparison, the other methods may quickly fall

Table 7 Results of the Std measure

STD	SMAMPA	MPA	SMA	GA	HHO	PSO	SSA	WOA	MFO	GOA
breastWDBCD	**0.036**	0.136	0.166	0.045	0.037	0.038	0.055	0.040	0.041	0.045
ionosphereD	0.050	0.115	0.094	0.048	0.048	**0.047**	0.052	0.048	0.065	0.066
wineD	**0.000**	0.038	0.111	0.030	0.020	0.016	0.082	0.034	0.053	0.065
breastcancerD	0.047	0.098	0.103	**0.044**	0.047	0.045	0.047	0.050	0.066	0.056
glassD	**0.017**	0.022	0.049	0.025	0.021	0.023	0.033	0.025	0.041	0.046
sonarD	0.070	0.132	0.077	**0.064**	0.082	0.094	0.084	0.086	0.083	0.068
LymphographyD	0.087	0.178	0.084	0.062	0.055	0.062	0.076	**0.053**	0.105	0.116
tic-tac-toeD	**0.000**	0.224	0.248	0.183	0.011	0.040	0.098	0.087	0.098	0.190
waveformD	0.016	0.106	0.140	0.017	**0.013**	0.014	0.019	0.019	0.036	0.055
clean1dataD	0.051	0.047	0.093	0.038	0.042	**0.037**	0.045	0.042	0.046	0.054
SPECTD	**0.042**	0.065	0.096	0.055	0.060	0.048	0.052	0.048	0.065	0.065
ZooD	**0.000**	0.012	0.157	0.019	0.010	0.011	0.069	0.015	0.061	0.092
ecoliD	0.032	0.046	0.109	0.028	0.025	0.025	0.038	**0.024**	0.047	0.104
CongressEWD	**0.035**	0.169	0.155	0.043	0.043	0.061	0.069	0.046	0.070	0.109
ExactlyD	0.020	0.274	0.134	0.224	0.083	**0.000**	0.118	0.181	0.207	0.116
Exactly2D	**0.018**	0.048	0.069	0.024	0.025	0.026	0.041	0.026	0.031	0.041
M-of-nD	**0.000**	0.314	0.201	0.151	0.068	**0.000**	0.110	0.125	0.184	0.127
VoteD	0.052	0.083	0.150	0.058	**0.038**	0.065	0.067	0.044	0.078	0.087
krvskpD	**0.016**	0.144	0.121	0.029	0.021	0.018	0.027	0.028	0.031	0.116
heartD	0.031	**0.027**	0.093	0.043	0.039	0.038	0.039	0.045	0.063	0.079

Bold values indicate the best result

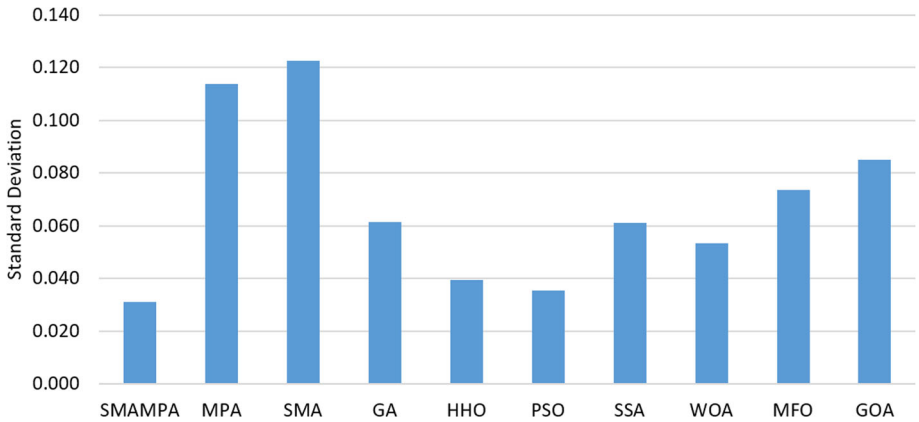

Fig. 4 Standard deviation average for all algorithms

into the local optima problem. Investigating the selected number of optimal features by SMAMPA has sufficient exploration energy than other comparative algorithms, proved by selecting fewer features over the tested benchmark datasets.

5.5 Experiment series 2: real−world quantitative structure-activity relationship application

In this section, we evaluate ability of the proposed method in selecting the most relevant features using real−world problems. Quantitative structure-activity relationship (QSAR) models is a mathematical framework in chemometrics to explain the structural relationship between chemical compounds and biological activity [62–65]. The QSAR modelling has been conducted to study the proposed algorithm and verify its effectiveness. Six high-dimensional datasets are adopted. The first dataset is the inhibitors of influenza A viruses (H1N1). An RNA virus called influenza causes a respiratory infection. It is a highly dangerous illness that is associated with high rates of mortality and morbidity. The influenza virus has two main glycoproteins on its surface: neuraminidase and haemagglutinin. Thus, utilizing compounds that block neuraminidase can prevent host cells from becoming infected with viruses and prevent the virus from spreading across cells. According to IC50, this dataset contained two

Table 8 The evaluation of the selected features number of all benchmark

NF	SMAMPA	MPA	SMA	GA	HHO	PSO	SSA	WOA	MFO	GOA
breastWDBCD	12	13	**8**	16	16	15	16	18	16	15
ionosphereD	8	6	**6**	15	12	14	15	12	16	17
wineD	7	7	**5**	8	8	7	7	7	7	6
breastcancerD	10	**8**	9	16	11	15	15	12	16	16
glassD	**4**	5	5	5	5	5	5	5	5	5
sonarD	**20**	23	20	30	28	29	30	31	30	30
LymphographyD	9	**7**	8	10	9	9	9	10	10	9
tic-tac-toeD	9	8	**4**	8	9	9	9	9	6	5
waveformD	14	15	**6**	13	15	14	13	17	14	12
clean1dataD	52	56	**42**	82	73	81	82	73	85	82
SPECTD	**8**	9	9	11	9	10	11	10	12	11
ZooD	9	9	**6**	9	10	9	9	10	9	8
ecoliD	5	5	**4**	5	5	5	5	5	4	4
CongressEWD	7	5	**4**	8	7	7	7	7	8	8
ExactlyD	**6**	7	6	8	7	7	7	9	8	7
Exactly2D	5	**4**	4	6	4	7	6	4	6	7
M-of-nD	7	**5**	6	8	8	7	7	8	8	7
VoteD	**4**	4	5	8	6	8	7	6	8	8
krvskpD	21	19	**10**	20	20	20	20	24	21	19
heartD	**6**	8	6	8	7	7	7	8	7	7

Bold values indicate the best result

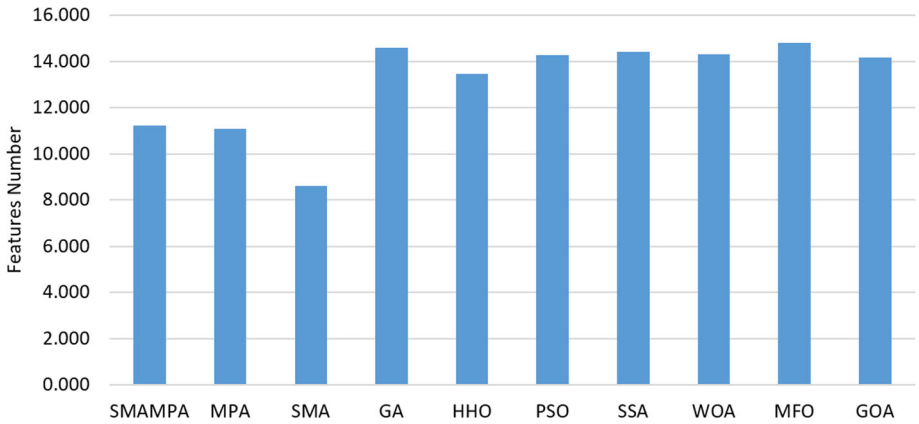

Fig. 5 Selected features of each optimization method

classes of active compound (IC50 < 20 μM) and weakly active compound (IC50 > 20 μM). This data consists of 2644 features and 479 instances [66].

The second dataset represents the anti-hepatitis C virus (hepatitis). Hepatitis C virus (HCV)-related liver conditions are among the most prevalent medical issues in the world today. The compounds employed have anti-hepatitis C virus action and were thiourea derivatives. This dataset containing 2952 features and 121 instances. According to EC50, the compounds were split into two sets: active and inactive compounds when EC50 < 0.1 μM and EC50 $\geq$ 0.1 μM, respectively [67].

The third dataset, called Chalcone, relates to a wide range of antibiotics with unique bioactivities against Candida albicans. The minimum inhibitory concentration (MIC) against C. albicans in mM/L was used to measure the antibacterial activities, which were expressed as pMIC, or the logarithm of the reciprocal of MIC. The median, or 1.30, of all 212 pMICs was taken into consideration as the cut-off to categorize these antimicrobial drugs into two groups based on the bioactivity distribution over the entire datasets. The first group consisted of 108 active compounds with pMIC values more than 1.30, and the remaining 104 inactive compounds made up the second group. The fourth, fifth, and the sixth datasets were publicly available in the UCI repository [61].

The proposed algorithm results, SMAMPA, are evaluated in terms of classification accuracy, selected features,

Table 9 Results of the computational time

Time	SMAMPA	MPA	SMA	GA	HHO	PSO	SSA	WOA	MFO	GOA
breastWDBCD	15.31	16.93	6.60	7.06	15.64	6.30	6.34	6.34	**6.29**	6.95
ionosphereD	14.92	16.63	6.40	6.89	15.40	6.17	6.17	6.29	**6.13**	6.80
wineD	14.68	16.42	**6.23**	7.17	15.41	6.44	6.41	6.39	6.43	6.72
breastcancerD	13.75	15.30	6.26	6.75	15.05	**6.03**	6.04	6.15	6.04	6.69
glassD	14.74	15.52	**5.18**	7.36	12.11	6.56	6.64	6.18	6.57	6.76
sonarD	13.38	14.74	6.29	6.71	14.59	6.05	6.04	6.03	**6.03**	7.14
LymphographyD	9.48	11.68	**5.27**	6.61	13.16	5.67	5.87	5.63	5.82	6.11
tic-tac-toeD	15.83	16.50	**6.99**	8.16	15.92	7.27	7.37	7.39	7.38	7.36
waveformD	41.40	61.29	**12.34**	20.30	40.94	18.07	18.09	19.70	18.56	18.41
clean1dataD	16.31	17.89	**6.68**	7.83	16.75	7.01	7.04	6.96	7.07	10.07
SPECTD	13.58	13.76	5.88	6.71	14.61	6.06	6.09	**5.87**	6.03	6.48
ZooD	10.76	12.97	**4.93**	7.29	14.70	6.17	6.38	6.31	6.44	6.52
ecoliD	10.22	10.51	**4.62**	5.79	11.67	5.18	5.19	5.12	5.24	5.23
CongressEWD	14.20	15.61	6.45	7.25	15.53	6.49	6.47	**6.37**	6.45	6.83
ExactlyD	15.28	16.51	**6.69**	7.89	16.48	7.11	7.06	6.96	7.03	7.35
Exactly2D	13.33	14.54	6.94	7.77	16.19	7.03	6.89	**6.30**	6.99	7.58
M-of-nD	15.44	16.55	**6.89**	8.05	16.76	7.20	7.23	7.13	7.23	7.55
VoteD	13.74	15.14	6.44	7.25	15.56	6.56	6.47	**6.43**	6.54	6.73
krvskpD	16.11	43.54	**10.45**	14.12	29.58	12.57	12.55	13.85	12.70	13.12
heartD	15.23	21.67	10.30	11.53	24.90	**10.28**	10.32	10.35	10.32	11.04

Bold values indicate the best result

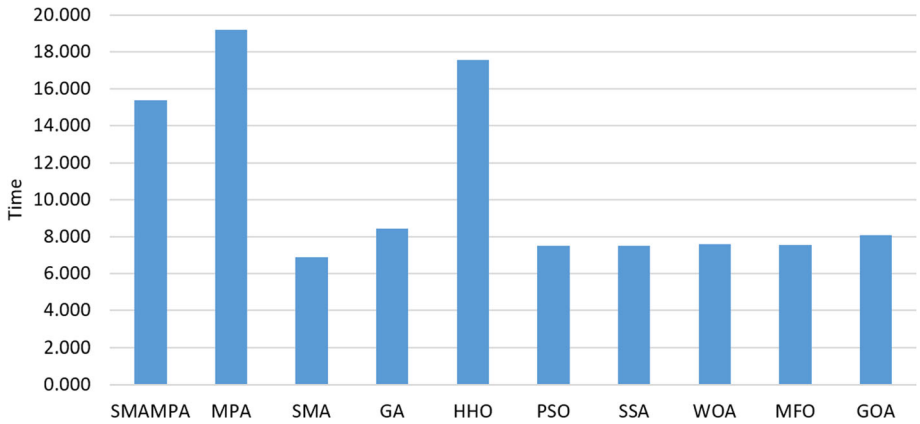

Fig. 6 Computational time Average of all algorithms

and standard deviation (Std). All results are summarized in Tables 13, 14 and 15.

From the Table 12, we assess the superiority of proposed algorithms, compared to others well-known algorithms. However, SMAMPA can be described as stable methods in most of all datasets except Biodeg dataset in which GA algorithm is better.

As can be seen from Table 13, the proposed algorithm, SMAMPA, has a significantly larger accuracy. These results demonstrated that the reduction in features contributes to the improvement of the accuracy resulting from the other algorithms. In terms of the selected features (Table 14), it can see that the SMAMPA obtained better values than the compared methods. It selected fewer features with high classification accuracy. Related to Std in Table 15, the SMAMPA algorithm achieved the low Std results in the H1N1, OralToxicity, and AndrogenReceptor datasets and was considered the most stable algorithm than the other algorithms. Furthermore, the hepatitis and Chalcone datasets presented competitive results for the SMAMPA with other algorithms. In general, SMAMPA algorithm can be considered as stable algorithm. From the above analysis, the SMAMPA method showed a high selecting ability for the essential features with high accuracy and good stability.

Table 10 Accuracy compression between SMAMPA and the other methods in the literature

Name	SMAMPA	BDA	BSSAS3	bGWO2	GLR	SbBOA	BGOAM	Das	S-bBOA
breastWDBCD	**0.990**	0.979	0.948	0.935	–	0.971	0.970	–	0.971
ionosphereD	0.974	**0.991**	0.918	0.834	0.000	0.907	0.946	0.865	0.907
wineD	**1.000**	**1.000**	0.993	0.920	0.978	0.984	0.989	0.961	0.984
breastcancerD	0.974	–	**0.976**	0.975	–	0.969	0.974	0.971	0.969
glassD	**0.798**	–	–	–	0.730	–	–	0.692	–
sonarD	**0.989**	0.980	0.937	0.729	0.829	0.936	0.915	0.793	0.936
LymphographyD	0.932	**0.992**	0.890	0.700	–	0.868	0.912	–	0.868
tic−tac−toeD	**1.000**	–	0.821	–	–	0.798	0.791	–	0.798
waveformD	**0.794**	0.758	0.733	0.789	–	0.743	0.751	–	0.743
clean1dataD	**0.961**	–	0.880	0.727	–	0.883	–	–	0.883
SPECTD	**0.903**	0.850	0.836	0.822	–	0.846	0.826	–	0.846
ZooD	**1.000**	**1.000**	**1.000**	0.879	–	0.978	0.958	0.960	0.978
ecoliD	**0.862**	–	–	–	0.852	–	–	0.789	–
CongressEWD	**0.987**	0.987	0.963	0.938	–	0.959	0.976	0.526	0.959
ExactlyD	1.000	**1.000**	0.980	0.776	–	0.972	**1.000**	–	0.972
Exactly2D	0.465	**0.773**	0.758	0.750	–	0.760	0.736	–	0.760
M−of−nD	**1.000**	**1.000**	0.991	0.963	–	0.972	**1.000**	–	0.972
VoteD	0.982	**0.989**	0.951	0.920	–	0.965	0.963	–	0.965
krvskpD	**0.986**	0.979	0.964	0.956	–	0.966	0.974	–	0.966
heartD	**0.883**	0.876	0.860	0.776	–	0.824	0.836	0.784	0.824

Bold values indicate the best result

Table 11 Description of the real-world datasets

Dataset	Features	Instances	Classes
H1N1	2644	479	2
hepatitis	2952	121	2
Chalcone	2821	100	2
biodeg	41	1055	2
OralToxicity	1024	8992	2
AndrogenReceptor	1024	1687	2

Table 12 Real application: Average of the fitness functions values

Name	SMAMPA	MPA	SMA	GA	HHO	PSO	SSA	WOA	MFO	GOA
H1N1	**0.3714**	0.3724	0.4898	0.4160	0.3897	0.4011	0.4110	0.3973	0.4363	0.5443
hepatitis	**0.1235**	0.2023	0.3982	0.3818	0.2719	0.3679	0.3919	0.2400	0.4362	0.4514
Chalcone	**0.2906**	0.3166	0.5244	0.4837	0.3823	0.4820	0.5135	0.3636	0.5472	0.5352
Biodeg	0.3310	0.3229	0.4204	**0.3168**	0.3482	0.3312	0.3427	0.3614	0.3711	0.3561
OralToxicity	**0.2504**	0.2530	0.2645	0.2559	0.2576	0.2581	0.2593	0.2572	0.2639	0.2624
AndrogenReceptor	**0.2758**	0.2819	0.3194	0.2860	0.2858	0.2870	0.2877	0.2923	0.3087	0.3107

Bold values indicate the best result

Table 13 Real application: The accuracy percentage

Name	SMAMPA	MPA	SMA	GA	HHO	PSO	SSA	WOA	MFO	GOA
H1N1	**86.09**	85.89	70.56	82.54	84.68	83.77	82.98	84.09	78.06	50.63
hepatitis	**96.41**	95.21	83.33	85.00	91.82	85.91	84.09	93.18	80.61	79.24
Chalcone	**0.9043**	0.8869	0.7166	0.7623	0.8503	0.7634	0.7314	0.8594	0.6971	0.7097
Biodeg	0.8902	0.8958	0.8201	**0.8996**	0.8788	0.8902	0.8826	0.8693	0.8617	0.8731
OralToxicity	**0.9373**	0.9358	0.9298	0.9344	0.9335	0.9333	0.9326	0.9337	0.9302	0.9310
AndrogenReceptor	**0.9234**	0.9194	0.8969	0.9171	0.9177	0.9165	0.9159	0.9135	0.9034	0.9023

Bold values indicate the best result

Table 14 Real application: Selected features number

Name	SMAMPA	MPA	SMA	GA	HHO	PSO	SSA	WOA	MFO	GOA
H1N1	**1023**	1212	1060	1323	1324	1310	1315	1259	1398	1313
hepatitis	**1019**	1843	1329	1480	1347	1475	1476	1222	1514	1470
Chalcone	**634**	850	657	1310	661	1329	1338	639	1377	1327
Biodeg	**12**	19	16	20	28	19	21	27	23	26
OralToxicity	**381**	431	516	512	507	524	508	659	513	510
AndrogenReceptor	460	**387**	494	523	559	514	509	557	501	517

Bold values indicate the best result

Table 15 Real application: The standard deviation values

Dataset	SMAMPA	MPA	SMA	GA	HHO	PSO	SSA	WOA	MFO	GOA
H1N1	**0.0345**	0.0348	0.0921	0.0394	0.0363	0.0374	0.0368	0.0356	0.0727	0.1536
hepatitis	0.0658	0.0836	0.0902	0.0649	0.0889	0.0746	0.0743	0.1029	**0.0605**	0.0619
Chalcone	0.1060	0.1135	0.0917	0.0615	0.0597	0.0648	0.0697	0.0915	**0.0587**	0.0618
Biodeg	0.0172	0.0085	0.0563	0.0030	**0.0000**	0.0114	**0.0000**	0.0079	0.0230	0.0080
OralToxicity	**0.0062**	0.0119	0.0154	0.0107	0.0120	0.0117	0.0116	0.0102	0.0136	0.0098
AndrogenReceptor	**0.0232**	0.0328	0.0323	0.0340	0.0259	0.0337	0.0365	0.0326	0.0355	0.0347

Bold values indicate the best result

6 Conclusion and future work

This study developed a new feature selection (FS) method by enhancing the original style of the slime mould algorithm (SMA). We leverage the exploration ability of the marine predators algorithm (MPA) to work as a local search method for the proposed method. The modified version, namely, SMAMPA, was evaluated on twenty well-known UCI benchmark datasets, using different evaluation metrics. Moreover, it was compared to the traditional SMA, MPA, and several state-of-art optimization methods. The developed SMAMPA showed superior performance over several optimization algorithms and several modified optimization algorithms. Furthermore, to verify the efficiency of the SMAMPA on more complicated and high-dimensional real-world problems, six datasets related to chemometrics, were used. Evaluation outcomes also showed the high performance of the SMAMPA, and it

obtained the best results compared to other optimization algorithms. According to the superior results of the developed SMAMPA, in future work, it could be further investigated in more complicated problems, such as multi-optimization problems, big data mining, and medical image processing.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant No. 62150410434) and in part by LIESMARS Special Research Funding.

Data Availability The datasets generated during and/or analysed during the current study are available in the UCI repository [61].

Declarations

Conflict of interest The authors declare that there is no conflict of interest.

References

1. Quiroz Juan C, Amit B, Dascalu Sergiu M, Lun Lau S (2017) Feature selection for activity recognition from smartphone accelerometer data. Intelli Autom Soft Comput 87:1–9
2. Han C, Zhou G, Zhou Y (2019) Binary symbiotic organism search algorithm for feature selection and analysis. IEEE Access 7:166833–166859
3. Han J, Pei J, Kamber M (2011) Data mining: concepts and techniques. Elsevier, Amsterdam
4. Hancer E, Xue B, Karaboga D, Zhang M (2015) A binary abc algorithm based on advanced similarity scheme for feature selection. Appl Soft Comput 36:334–348
5. Hua J, Tembe Waibhav D, Dougherty Edward R (2009) Performance of feature-selection methods in the classification of high-dimension data. Pattern Recogn 42(3):409–424
6. Jingwei T, Seyedali M (2020) A hyper learning binary dragonfly algorithm for feature selection: a covid-19 case study. Knowl-Based Syst 87:106553
7. Zhong C, Chen Y, Jian P (2020) Feature selection based on a novel improved tree growth algorithm. Int J Comput Intell Syst 13(1):247–258
8. Xue B, Zhang M, Browne WN (2014) Novel initialisation and updating mechanisms. particle swarm optimisation for feature selection in classification. Appl Soft Comput 18:261–276
9. Tan F, Xuezheng F, Zhang Y, Bourgeois Anu G (2008) A genetic algorithm-based method for feature subset selection. Soft Comput 12(2):111–120
10. Mustafa Serter U, Nihat Y, Onur I(2013) Feature selection method based on artificial bee colony algorithm and support vector machines for medical datasets classification. The Scientific World Journal 2013
11. Selvakumar B, Muneeswaran K (2019) Firefly algorithm based feature selection for network intrusion detection. Computers Secur 81:148–155
12. Emary E, Zawbaa Hossam M, Hassanien Aboul E (2016) Binary grey wolf optimization approaches for feature selection. Neurocomputing 172:371–381
13. Sindhu R, Ngadiran R, Yacob YM, Zahri Nik Adilah H, Hariharan M (2017) Sine-cosine algorithm for feature selection with elitism strategy and new updating mechanism. Neural Comput Appl 28(10):2947–2958
14. Faris H, Mafarja Majdi M, Heidari Ali A, Aljarah I, Ala'M A-Z, Mirjalili S, Fujita H (2018) An efficient binary salp swarm algorithm with crossover scheme for feature selection problems. Knowl-Based Syst 154:43–67
15. Ewees Ahmed A, Aziz Mohamed AE, Hassanien Aboul E (2019) Chaotic multi-verse optimizer-based feature selection. Neural Comput Appl 31(4):991–1006
16. Abualigah L, Diabat A, Mirjalili S, Elaziz MA, Gandomi AH (2021) The arithmetic optimization algorithm. Computer Methods Appl Mech Eng 376:113609
17. Laith A, Ali D (2021) Advances in sine cosine algorithm: a comprehensive survey. Artif Intell Rev 25:1–42
18. Ewees Ahmed A, Al-qaness Mohammed AA, Abualigah L, Oliva D, Algamal ZY, Anter AM, Ibrahim RA, Ghoniem RM, Elaziz MA (2021) Boosting arithmetic optimization algorithm with genetic algorithm operators for feature selection: case study on cox proportional hazards model. Mathematics 9(18):2321
19. Ibrahim Rehab A, Ewees Ahmed A, Oliva D, Elaziz MA, Songfeng L (2019) Improved salp swarm algorithm based on particle swarm optimization for feature selection. J Amb Intell Humanized Comput 10(8):3155–3169
20. Zorarpaci E, Aycseozel S (2016) A hybrid approach of differential evolution and artificial bee colony for feature selection. Expert Syst Appl 62:91–103
21. Arora S, Singh H, Sharma M, Sharma S, Anand P (2019) A new hybrid algorithm based on grey wolf optimization and crow search algorithm for unconstrained function optimization and feature selection. IEEE Access 7:26343–26361
22. Abd Mohamed E, Elaziz Ahmed A, Diego Oliva E, Pengfei D, Shengwu X (2017) A hybrid method of sine cosine algorithm and differential evolution for feature selection. In International conference on neural information processing, 145–155 Springer,
23. Elaziz MA, Ewees Ahmed A, Ibrahim RA, Songfeng L (2020) Opposition-based moth-flame optimization improved by differential evolution for feature selection. Math Computers Simul 168:48–75
24. Neggaz N, Ewees Ahmed A, Elaziz MA, Mafarja M (2020) Boosting salp swarm algorithm by sine cosine algorithm and disrupt operator for feature selection. Expert Syst Appl 145:113103
25. Laith A, Ali D (2020) A comprehensive survey of the grasshopper optimization algorithm: results, variants, and applications. Neural Comput Appl 25:1–24
26. Shimin L, Huiling C, Mingjing W, Asghar Heidari A, and Mirjalili S (2020) A new method for stochastic optimization. future generation computer systems, Slime mould algorithm
27. Kumar C, Dharma Raj T, Premkumar M, Dhanesh Raj T (2020) A new stochastic slime mould optimization algorithm for the estimation of solar photovoltaic cell parameters. Optik 223:165277
28. Chen Z, Liu W (2020) An efficient parameter adaptive support vector regression using k-means clustering and chaotic slime mould algorithm. IEEE Access 8:156851–156862
29. Al-Qaness Mohammed AA, Hong F, Ewees Ahmed A, Dalia Y, Mohammed Abd E (2020) Improved anfis model for forecasting Wuhan city air quality and analysis covid-19 lockdown impacts on air quality. Environ Res 871:110607
30. Ali D (2020) The optimal synthesis of thinned concentric circular antenna arrays using slime mold algorithm. Electromagnetics 58:1–13
31. Sun K, Jia H, Li Y, Jiang Z (2021) Hybrid improved slime mould algorithm with adaptive β hill climbing for numerical optimization. J Intell Fuzzy Syst (Preprint) 14:1667–1679
32. Ewees Ahmed A, Laith A, Dalia Y, Zakariya Yahya A, Al-Ganess Mohammed AA, Rehab Ali I, Mohamed Abd E (2021) Improved slime mould algorithm based on firefly algorithm for feature selection: a case study on qsar model. Eng Computers 69:1–15

33. Afshin F, Mohammad H, Seyedali M, Gandomi Amir H (2020) Marine predators algorithm: a nature-inspired metaheuristic. Expert Syst Appl 5:113377
34. Al-Qaness Mohammed AA, Saba Amal I, Elsheikh Ammar H, Elaziz MA, Ibrahim Rehab A, Songfeng L, Hemedan Ahmed A, Shanmugan S, Ewees Ahmed A (2020) Efficient artificial intelligence forecasting models for covid-19 outbreak in russia and brazil. Process Safety and Environmental Protection
35. Al-Qaness Mohammed AA, Ewees Ahmed A, Fan H, Abualigah L, Elaziz MA (2020) Marine predators algorithm for forecasting confirmed cases of Covid-19 in Italy, USA, Iran and Korea. Int J Environ Res Publ Health 17(10):3520
36. Elaziz MA, Ewees Ahmed A, Yousri D, Naji Husein S, Alwerfali Qamar A, Awad Songfeng L, Al-Qaness Mohammed AA (2020) An improved marine predators algorithm with fuzzy entropy for multi-level thresholding: Real world example of Covid-19 ct image segmentation. IEEE Access 8:125306–125330
37. Sahlol Ahmed T, Yousri D, Ewees Ahmed A, Al-Qaness MAA, Damasevicius R, Elaziz MA (2020) Covid-19 image classification using deep features and fractional-order marine predators algorithm. Scientif Rep 10(1):1–15
38. Yousri D, Hasanien Hany M, Fathy A (2020) Parameters identification of solid oxide fuel cell for static and dynamic simulation using comprehensive learning dynamic multi-swarm marine predators algorithm. Energy Conver Manage 228:113692
39. Elaziz MA, Shehabeldeen Taher A, Elsheikh Ammar H, Zhou J, Ewees Ahmed A, Al-qaness Mohammed AA (2020) Utilization of random vector functional link integrated with marine predators algorithm for tensile behavior prediction of dissimilar friction stir welded aluminum alloy joints. J Mater Res Technol 9(5):11370–11381
40. Al-qaness MAA, Ewees AA, Fan H, Abualigah L, Elaziz MA (2022) Boosted anfis model using augmented marine predator algorithm with mutation operators for wind power forecasting. Appl Energy 314:118851
41. Yousri D, Babu TS, Beshr E, Eteiba Magdy B, Allam D (2020) A robust strategy based on marine predators algorithm for large scale photovoltaic array reconfiguration to mitigate the partial shading effect on the performance of pv system. IEEE Access 8:112407–112426
42. Sayed Safinaz A-F, Nabil E, Badr A (2016) A binary clonal flower pollination algorithm for feature selection. Pattern Recognit Lett 77:21–27
43. Mafarja M, Mirjalili S (2018) Whale optimization approaches for wrapper feature selection. Appl Soft Comput 62:441–453
44. Sayed GI, Hassanien AE, Azar AT (2019) Feature selection via a novel chaotic crow search algorithm. Neural Comput Appl 31(1):171–188
45. Arora S, Anand P (2019) Binary butterfly optimization approaches for feature selection. Expert Syst Appl 116:147–160
46. Jingwei T, Abdul Rahim A (2020) A new and fast rival genetic algorithm for feature selection. J Supercomput 58:1–31
47. Zhang Y, Liu R, Wang X, Chen H, Li C (2020) Boosted binary harris hawks optimizer and feature selection. Structure 58(25):26
48. Elgamal Zenab M, Yasin Norizan BM, Tubishat M, Alswaitti M, Mirjalili S (2020) An improved harris hawks optimization algorithm with simulated annealing for feature selection in the medical field. IEEE Access 8:186638–186652
49. Salima O, Mohamed AE (2020) Enhanced crow search algorithm for feature selection. Expert Syst Appl 25:113572
50. Mafarja M, Aljarah I, Heidari AA, Faris H, Fournier-Viger P, Li X, Mirjalili S (2018) Binary dragonfly optimization for feature selection using time-varying transfer functions. Knowl-Based Syst 161:185–204
51. Zhang H, Wang J, Sun Z, Zurada Jacek M, Pal Nikhil R (2019) Feature selection for neural networks using group lasso regularization. IEEE Trans Knowl Data Eng 32(4):659–673
52. Mafarja M, Aljarah I, Faris H, Hammouri Abdelaziz I, Ala'M A-Z, Mirjalili S (2019) Binary grasshopper optimisation algorithm approaches for feature selection problems. Expert Syst Appl 117:267–286
53. Das A, Das S (2017) Feature weighting and selection with a pareto-optimal trade-off between relevancy and redundancy. Pattern Recognit Lett 88:12–19
54. Whitley D (1994) A genetic algorithm tutorial. Stat Comput 4(2):65–85
55. Ali Asghar H, Seyedali M, Hossam F, Ibrahim A, Majdi M, Huiling C (2019) Algorithm and applications, Harris hawks optimization. Fut Gener Computer Syst 97:849–872
56. Eberhart R, Kennedy J (1995) A new optimizer using particle swarm theory. In *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pp. 39–43. Ieee,
57. Mirjalili S, Gandomi Amir H, Mirjalili Seyedeh Z, Saremi S, Faris H, Mirjalili SM (2017) Salp swarm algorithm: a bio-inspired optimizer for engineering design problems. Adv Eng Softw 114:163–191
58. Mirjalili S, Lewis A (2016) The whale optimization algorithm. Adv Eng Softwa 95:51–67
59. Mirjalili S (2015) Moth-flame optimization algorithm: a novel nature-inspired heuristic paradigm. Knowl-Based Syst 89:228–249
60. Saremi S, Mirjalili S, Lewis A (2017) Grasshopper optimisation algorithm: theory and application. Adv Eng Softw 105:30–47
61. Dua D, Graff C (2017) UCI machine learning repository,
62. Algamal ZY, Alhamzawi R, Ali Haithem TM (2018) Gene selection for microarray gene expression classification using bayesian lasso quantile regression. Computers Biol Med 97:145–152
63. Algamal Zakariya Y, Lee MH, Al-Fakih AM (2016) High-dimensional quantitative structure-activity relationship modeling of influenza neuraminidase a/pr/8/34 (h1n1) inhibitors based on a two-stage adaptive penalized rank regression. J Chemometr 30(2):50–57
64. Algamal ZY, Lee MH, Al-Fakih AM, Aziz M (2017) High-dimensional qsar classification model for anti-hepatitis c virus activity of thiourea derivatives based on the sparse logistic regression model with a bridge penalty. J Chemometr 31(6):e2889
65. Algamal ZY, Qasim MK, Ali HTM (2017) A qsar classification model for neuraminidase inhibitors of influenza a viruses (h1n1) based on weighted penalized support vector machine. SAR and QSAR Environ Res 28(5):415–426
66. Al-Thanoon Niam A, Qasim Omar S, Algamal ZY (2019) A new hybrid firefly algorithm and particle swarm optimization for tuning parameter estimation in penalized support vector machine with application in chemometrics. Chemometr Intell Lab Syst 184:142–152
67. Al-Dabbagh ZT, Algamal ZY (2019) A robust quantitative structure-activity relationship modelling of influenza neuraminidase a/pr/8/34 (h1n1) inhibitors based on the rank-bridge estimator. SAR and QSAR Environ Res 30(6):417–428

Authors and Affiliations

Ahmed A. Ewees[1,2] · Mohammed A. A. Al-qaness[3] · Laith Abualigah[4,5] · Zakariya Yahya Algamal[6,7] · Diego Oliva[8] · Dalia Yousri[9] · Mohamed Abd Elaziz[10,11,12,13]

✉ Ahmed A. Ewees
ewees@du.edu.eg

Mohammed A. A. Al-qaness
alqaness@zjnu.edu.cn

Laith Abualigah
Aligah.2020@gmail.com

Zakariya Yahya Algamal
zakariya.algamal@uomosul.edu.iq

Diego Oliva
diego.oliva@cucei.udg.mx

Dalia Yousri
day01@fayoum.edu.eg

Mohamed Abd Elaziz
abd_el_aziz_m@yahoo.com

[1] Department of Information Systems, College of Computing and Information Technology, University of Bisha, Bisha 61922, Saudi Arabia

[2] Department of Computer, Damietta University, Damietta 34517, Egypt

[3] College of Physics and Electronic Information Engineering, Zhejiang Normal University, Jinhua 321004, China

[4] Hourani Center for Applied Scientific Research, Al-Ahliyya Amman University, Amman 19328, Jordan

[5] Faculty of Information Technology, Middle East University, Amman 11831, Jordan

[6] Department of Statistics and Informatics, University of Mosul, Mosul, Iraq

[7] College of Engineering, University of Warith Al-Anbiyaa, Karbala, Iraq

[8] Depto. de Ciencias Computacionales, Universidad de Guadalajara, CUCEI, Av. Revolución 1500, Guadalajara, Jal, Mexico

[9] Department of Electrical Engineering, Faculty of Engineering, Fayoum University, Fayoum, Egypt

[10] Department of Mathematics, Faculty of Science, Zagazig University, Zagazig 44519, Egypt

[11] Faculty of Computer Science and Engineering, Galala University, Suez, Egypt

[12] Artificial Intelligence Research Center (AIRC), Ajman University, Ajman, UAE

[13] Department of Electrical and Computer Engineering, Lebanese American University, Byblos, Lebanon