International Journal of Applied Science and Engineering

Special issue: The 10th International Conference on Awareness Science and Technology (iCAST 2019)

Analysis of various transfer functions for binary owl search algorithm in feature selection problem

Ashis Kumar Mandal[1*], Rikta Sen[1], Basabi Chakraborty[2]

[1] *Graduate School of Software & Information Science, Iwate Prefectural University, Iwate, Japan*
[2] *Faculty of Software & Information Science, Iwate Prefectural University, Iwate, Japan*

ABSTRACT

Owl Search Algorithm (OSA) is a recently proposed nature-inspired meta-heuristic algorithm which is easily implementable and exhibits great potential for solving continuous optimization problems. In our earlier work, a binary version of owl search algorithm (BOSA), with transfer functions for mapping the continuous solution space into a binary one, has been developed and applied in optimal feature subset selection problem. In our preliminary simulation experiments, it was found that the performance of the solution depends on the type of transfer function used. In this work, an extensive analysis of various types of transfer functions and their respective effects on the selection of optimal feature subset has been studied by simulation experiments with multiple benchmark datasets. Transfer functions of three different families, S-shaped, V-shaped and quadratic, are used for designing eleven BOSA models, each of which is made by combining individual transfer function. The performances of the proposed wrapper based feature subset selection algorithm based on several BOSA models have been evaluated by simulation experiments with twenty datasets for finding out the best model. The best observed BOSA model has also been compared with other similar meta-heuristics algorithms for feature subset selection. Experimental results show that the feature subset selected by BOSA with quadratic transfer function produces the highest classification accuracy with the minimum number of selected features compared to other algorithms.

Keywords: Binary owl search algorithm; Meta-heuristics; Feature selection; Transfer functions.

1. INTRODUCTION

Feature selection is an important preprocessing task in many machine learning and data mining applications that selects appropriate features and removes the irrelevant and redundant features. The merits of performing feature selection are a better understanding of the characteristics of data, reduction of training time and improvement of the performance of the classifier by reducing the dimensionality of data (Chandrashekar and Sahin, 2014). Nowadays, in many real-world applications, the number of features in data has increased a lot, many of which are not so important in building the classifier or predictor model. One of the effective approaches is to find a subset of the feature set, which is the most suitable for the respective classifier model (Khushaba et al., 2011; Bolón-Canedo et al., 2015).

Feature subset selection process includes two key factors: search technique, and the evaluation criterion (Xue, 2016). The search technique explores the search space to find the optimal feature subset(s) while the evaluation criterion measures the goodness of a feature subset to guide the search. Based on the evaluation criterion, feature selection

OPEN ACCESS

Received: April 15, 2020
Revised: July 14, 2020
Accepted: August 4, 2020

Corresponding Author:
Ashis Kumar Mandal
g236r004@s.iwate-pu.ac.jp

Copyright: The Author(s). This is an open access article distributed under the terms of the Creative Commons Attribution License (CC BY 4.0), which permits unrestricted distribution provided the original author and source are cited.

Publisher:
Chaoyang University of Technology
ISSN: 1727-2394 (Print)
ISSN: 1727-7841 (Online)

https://doi.org/10.6703/IJASE.202009_17(3).281

algorithms can be classified into wrapper and filter approaches. The wrapper approach uses classification accuracy as the measurement tool for evaluation of subsets, unlike filter approaches which use specific characteristics of the data as evaluation measures. Wrapper approaches are more computationally expensive than filter approaches though they can produce more appropriate feature subset for a particular classification model. Regarding search strategy, exhaustive search bears high computational cost, especially for high dimensional data sets, while random search might not produce an acceptable solution (Dash and Liu, 2003). Finding the optimal subset of features is, in fact, a combinatorial optimization problem. To handle this optimization problem effectively, meta-heuristic search algorithms have been very popular (Fausto et al., 2019; Wang et al., 2014) as they can provide good quality solution within reasonable computational cost even if may not guarantee an optimal solution.

Till now, several meta-heuristic algorithms have been developed to solve the feature selection problem. Early research in metaheuristic-based feature selection has produced feature subset selection algorithms using Binary Genetic Algorithm (BGA), and its different variants like parallel GA (Soufan et al., 2015), fuzzy rough GA (Chakraborty, 2014), multiobjective GA (Oliveira et al., 2003) were reported. Furthermore, to perform efficient feature selection, GA was hybridized with other metaheuristics, such as GA with Particle Swarm Optimization (PSO) (Mistry et al., 2017) and GA with great deluge algorithm (Guha et al., 2019). Researchers have also attempted to solve the feature selection problem using Particle Swarm Optimization (PSO). A binary PSO (BPSO) was developed in feature selection as in Kennedy and Eberhart (1997). Xue et al. proposed an effective PSO-based multiobjective feature selection (Xue et al., 2013). Several modified PSO strategies for feature selection were also reported in Chuang et al. (2011); Liu et al., 2011; Lu et al. (2017). Other potential approaches in feature subset selection fields are Ant Colony Optimization (ACO) and Harmony Search (HS). In Aghdam et al. (2009), a feature selection method based on ACO algorithm is used for text classification. In order to find the salient features, ACO was hybridized with Artificial Neural Networks (ANN) in Kabir et al. (2012) and Manoj et al. (2019). Feature selection approach based on Harmony Search strategies such as adaptive harmony search (Dash, 2018), rough set based harmony search (Inbarani et al., 2015), and self-adjusting harmony search (Zheng, 2015) were proposed and used for finding appropriate feature subset.

Recently, newly developed nature-inspired meta-heuristics have also been shown potential for solving feature selection problem. Binary Ant Lion Optimization (BALO) in feature selection has been proposed in Emary et al. (2016). In order to explore the global search, several different chaotic maps were hybridized with Ant Lion Optimization (Zawbaa et al., 2016). Similarly, chaotic maps with other meta-heuristics, including Chaotic Dragonfly (Sayed et al., 2019), Chaotic Multi-verse (Ewees et al., 2019), Chaotic Fruit Fly (Zhang et al., 2020) are also some of the efficient feature selection approaches. Mirjalili et al. proposed an efficient feature selection approach using Whale Optimization Algorithm (WOA) (Mafarja and Mirjalili, 2018). They also combined it with Simulated Annealing in another feature section task (Mafarja and Mirjalili, 2017). Another potential approach reported in the literature is Grey Wolf Optimization (GWO) based Feature Selection (Emary et al., 2016). GWO was also hybridized with PSO and used for selecting salient features (Al-Tashi et al., 2019). Besides, Grasshopper Optimization Algorithm (GOA) (Mafarja et al., 2019), Butterfly Optimization Algorithm (BOA) (Arora and Anand, 2019), Teaching Learning-based Algorithm (TLBA) (Allam and Nandhini, 2018), Black Hole Algorithm (BHA) (Pashaei and Aydin, 2017), Salp Swarm Algorithm (SSA) (Faris et al., 2018), Social Spider Algorithm (SSA) (BAŞ and ÜLKER, 2020), Sine Cosine Algorithm (SCA) (Sindhu et al., 2017), and Symbiotic Organism Search (SOS) (Han et al., 2019) are some recent bio-inspired meta-heuristics algorithms proposed for feature subset selection.

Owl Search Algorithm (OSA) is a new population-based meta-heuristic algorithm that was initially proposed for solving continuous optimization problems (Jain et al., 2018). Recently, OSA and its modifications have been employed for solving several real-world problems. A modified OSA algorithm has been proposed to increase the efficiency in designing combined cooling, heating, and power (CCHP) systems (Cao et al., 2020). Another real-world problem where an improved version of OSA was used to generate a model that can supply guaranteed power for the base transceiver station (BTS) of telecommunication tower (Li et al., 2020). A chaotic owl search algorithm was proposed in designing an effective model of negotiation process (El-Ashmawi et al., 2020). Maximum power point tracking (MPPT) optimization was performed using combining Perturb and Observe (P&O) and OSA algorithms so that maximum efficient power can be tracked (Farhan et al., 2019). Although in all of the above cases, OSA shows better performance, the potential of this algorithm for addressing feature selection problem has not been investigated explicitly so far. In our preliminary study, we proposed a binary variant of OSA (BOSA) to solve the feature selection problem and obtained some promising results (Mandal et al., 2019). Transfer functions (Crawford et al., 2017) are used to map the continuous solution space of a meta-heuristic to discrete or binary solution space, and it is observed that these transfer functions can affect the performance of the proposed meta-heuristic algorithm.

The objective of the present work is to extend our previous work (Mandal et al., 2019) to address the feature selection problem with our proposed binary owl search algorithm (BOSA) extensively with various types of transfer functions. In this work, four S-shaped, four V-shaped, and three quadratic transfer functions are investigated, and these transfer functions are combined with the binary owl search algorithm to build several BOSA models. Simulation experiments have been done for evaluation of different BOSA models based on different transfer functions in comparison with the state-of-the-art other metaheuristic algorithms.

The paper is organized as follows. Section 2 describes the original Owl Search Algorithm (OSA) and the modification of OSA for addressing feature subset selection, a binary optimization problem. Section 3 represents the detail description of various transfer functions used with BOSA for feature selection problem, the study of their effect in the optimal feature subset selection is the main contribution of this work which is done by simulation experiments explained in section 4. Section 5 contains the experimental results of the study. Finally, conclusion and future directions are included in section 6.

2. OWL SEARCH ALGORITHM (OSA) AND FEATURE SUBSET SELECTION

The present section represents a brief description of original OSA followed by its binary version BOSA and the application of BOSA in feature subset selection problem.

2.1 Owl Search Algorithm (OSA)

Owl Search Algorithm is a nature-inspired metaheuristic algorithm for solving optimization problem in which the hunting mechanism of owls in the dark is taken as the basis (Jain et al., 2018). Owl's brain generates an auditory map of prey sound and can target the prey in the dark by their special hearing ability. For solving the global optimization problem, the position of an owl represents an individual solution and a group of owls work to find the prey or the global optimum solution. In the d dimension solution space, every owl is denoted with randomly generated positions. For n number of owls, the initial position of i^{th} owl is based on the following Equation:

$$O_i^j = O_L^j + U(0,1) \times (O_U^j - O_L^j) \qquad (1)$$

where O_i^j is the i^{th} owl in j^{th} dimension and $U(0,1)$ is a uniform random number between 0 and 1. O_U^j and O_L^j are the upper and lower bound of j^{th} dimension, respectively.

The fitness value of O_i owl is evaluated by fitness function $f(.)$, which is associated with the problem to be optimized and can be represented as follows:

$$f_i = f([O_i^1, O_i^2, \ldots\ldots, O_i^d]) \qquad (2)$$

The fitness value of each owl's position is directly correlated with the intensity information received through the ears of the owl. Therefore, the best owl receives the maximum intensity as it is found closer to the prey. The normalized intensity of i^{th} owl is calculated using the following Equation.

$$I_i = \frac{(f_i - w)}{(b - w)} \qquad (3)$$

where b and w is the best and the worst intensity information respectively and defined by the following Equation.

$$b = \max_{k \in \{1,\ldots,n\}} f_k \qquad (4)$$

$$w = \min_{k \in \{1,\ldots,n\}} f_k \qquad (5)$$

The distance information between the owl O_i and the prey is calculated using Euclidean distance

$$R_i = \| O_i, V \|_2 \qquad (6)$$

where V is a prey, and the position of the fittest owl determines its location. During hunting, the change of the intensity information of i^{th} owl is calculated as follows:

$$I_{c_i} = \frac{I_i}{R_i^2} + N_s \qquad (7)$$

where N_s is random noise.

Finally, individual owl O_i changes its position based on the movement of prey. The mathematical model for updating each of the owls is as follows:

$$O_i(t+1) = \begin{cases} O_i(t) + \beta \times I_{c_i} \times |\alpha V - O_i(t)|, & \text{if } p_{vm} < 0.5 \\ O_i(t) - \beta \times I_{c_i} \times |\alpha V - O_i(t)|, & \text{if } p_{vm} \geq 0.5 \end{cases} \qquad (8)$$

where $O_i(t+1)$ is the new position of the owl at iteration $(t+1)$, α is a uniformly distributed random number in the range [0,0.5]. β is a user-defined parameter which decreases linearly from 1.9 to 0, and p_{vm} represents the probability of the movement of the prey.

2.2 Binary Owl Search Algorithm (BOSA)

In this section, the binary variant of OSA proposed in our earlier work and applied in feature subset selection problem has been described in brief. The modification of OSA for binary optimization problem has been proposed in which different transfer functions are used to convert continuous solution space to binary solution space, and the fitness function has been defined to be used in the search process. The details can be found in Mandal et al. (2019).

The original OSA, which is used for the continuous optimization problem, cannot handle binary optimization problem like feature subset selection. The original OSA has been modified for a binary problem. Usually, a feature set can be represented as a one-dimensional binary vector containing a string of 0 and 1. The binary value 1 indicates the feature is selected, while 0 indicates the feature is not selected. A feature subset is therefore equivalent to the position of an individual owl in a binary solution space. For a feature selection task, the initial position of i^{th} owl with j^{th} dimension of Equation (1) is represented in a binary solution space as follows:

$$O_i^j = R \qquad (9)$$

where R is a random number € {0,1}.

According to Tian and Shen (2005); Banka and Dara (2015), Hamming distance is the most straightforward and efficient method to measure the difference between two-bit sequences. It is observed that if feature vectors are high-dimensional, distance measure computed based on the Hamming distance may be a good choice than Euclidian distance (Jain et al., 1999). Moreover, computational cost will be lowered by the use of Hamming distance instead of Euclidian distance.

As the solution space is binary, Hamming distance function is used during measurement of the distance between a candidate owl and the prey. The new Equation is as follows:

$$R_i = \sum_{j=1}^{d} | O_i^j - V^j | \tag{10}$$

where $O_i^j, V^j \in \{0,1\}$.

The control parameter β need to be decided. As reported previously, the initial value of β should be relatively high and then it decreases gradually with the progress of the optimization step. We define the following linear function for the parameter to fulfil the requirement.

$$\beta = \theta - (t \times \theta)/T \tag{11}$$

where t is the current iteration, T is the total number of iterations, and θ is a constant. We choose the value of θ as 2.0 with some preliminary experiment such that it can assist exploration in the search space.

Finally, Equation (8), which updates the position vector of each owl has been redefined. In BOSA, individual owl O_i at t^{th} iteration is updated and produces a step vector $\Delta O_i(t+1)$, which contains the continuous value of O_i at iteration $(t+1)$. Each continuous value is then mapped into binary value using transfer functions. The following Equation describes how this step vector is calculated.

$$\Delta O_i(\mathrm{t}+1) = \begin{cases} O_i(t) + \beta \times I_{c_i} \times |\alpha V - O_i(t)|, & \text{if } p_{vm} < 0.5 \\ O_i(t) - \beta \times I_{c_i} \times |\alpha V - O_i(t)|, & \text{if } p_{vm} \geq 0.5 \end{cases} \tag{12}$$

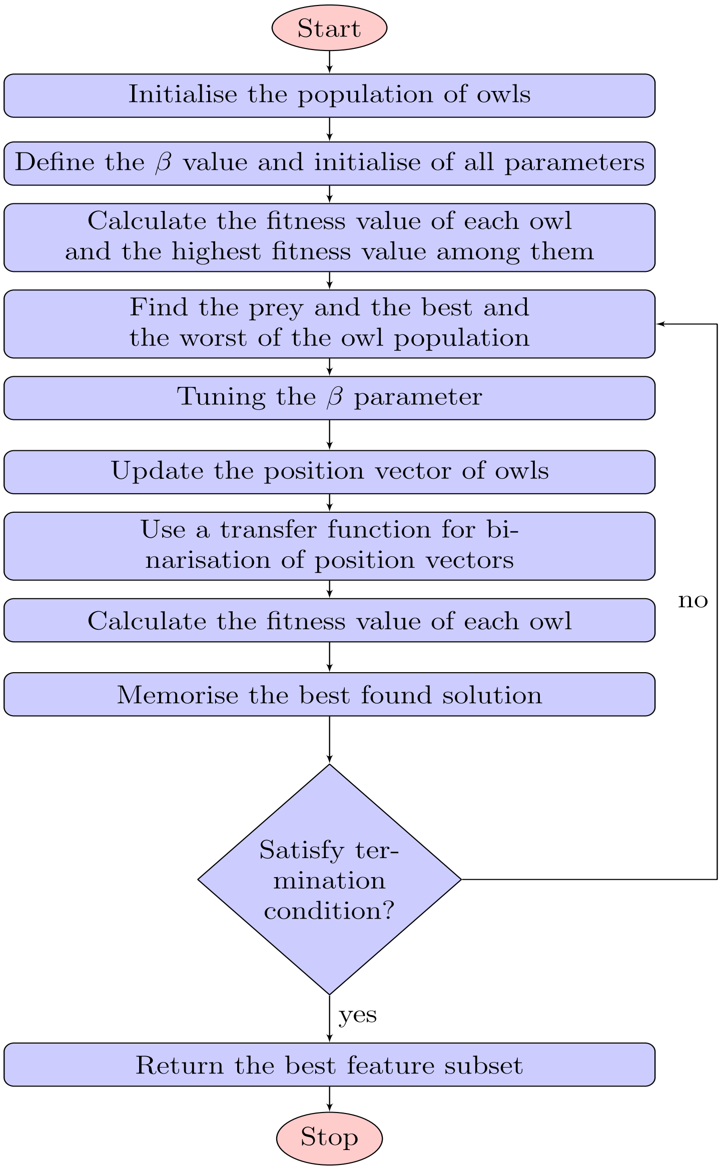

Fig. 1. Proposed BOSA for feature selection task.

Algorithm 1: BOSA with different transfer functions for Feature Subset selection

```
Input: Testing dataset with full features
Output: Best feature subset
/* Start of parameter definition                                    */
$T$ is the maximum number of iteration
$n$ is number of owls
$d$ is the dimension of each owl
$V$ is the prey's position
$t$ is the current iteration
$\beta$ is controlling parameter
$\Upsilon$ is a transfer function
/* End of parameter definition                                      */
Initialize $\beta$ function
$\Upsilon \leftarrow$ select a transfer function from Table 1
Initialize each owl's position randomly using Eq. (9)
Calculate the fitness of owls by Eq. (13)
Calculate the highest fitness value from the owls by Eq. (4)
$t \leftarrow 0$
while $t < T$ do
    $V \leftarrow$ position of highest fitness value so far
    Evaluate the best fitness values of the owls by Eq. (4)
    Evaluate the worst fitness values of the owls by Eq. (5)
    Evaluate the $\beta$ by Eq.(11)
    foreach owl do
        Calculate the normalized intensity of owl by Eq.(3)
        Calculate the Hamming distance between owl and prey by Eq.(10)
        Calculate the intensity difference of owl by Eq.(7)
        Evaluate the step vector by Eq.(12)
        if type of $\Upsilon = S - shaped$ then
            Evaluate binary position by Eq.(14)
        else
            Evaluate binary position by Eq. (15)
        Evaluate the fitness of owl by Eq. (13)
    Best feature subset $\leftarrow$ the fittest owl found so far
    $t \leftarrow t + 1$
```

2.3 Procedure for Feature Subset Selection with BOSA

Fig. 1 shows the general procedure for feature subset selection with BOSA. The proposed approach takes dataset with the full feature set as the input and returns the best feature subset as output. As mentioned previously, a solution of a feature subset represents the position of an owl in the binary space. At the beginning of the procedure, the initial owl population representing the initial set of solutions is randomly generated, and the best among them is considered as the initial prey. In the following step, β value is defined, and the parameters are initialized. After the initialization, the iterative process is repeated until stopping criteria (maximum number of iteration) is satisfied. In each iteration, the fitness value of the owl population is evaluated, and then prey location and the best and worst owl locations are calculated. In order to balance between exploration and exploitation, β is tuned. Afterwards, the position of owls is updated, and a transfer function maps the continuous position values of each owl into binary ones. The best solution obtained so far is also tracked in each iteration, and when the termination condition is satisfied, the best solution is returned as the best feature subset. In this paper, we have used 11 different transfer functions (detailed presentation in the next section) which combined with BOSA has produced eleven different algorithms.

2.4 Fitness function for Evaluation of Feature Subset

Fitness function evaluates the individual feature subset during the meta-heuristic search process. In this study, the fitness function considers both the classification accuracy and the number of selected features, which form a single objective of the minimization problem. A good fitness value is obtained when classification accuracy is high with low feature cardinality. Accuracy of the classification function is calculated by K-nearest neighbor (KNN) where k is set to five. The fitness function is defined as follows.

$$f(O_i) = \omega \times A(O_i) + (1-\omega) \times (1 - \frac{S_T}{L_T}) \quad (13)$$

where $f(O_i)$ is the fitness function of the i^{th} feature subset O_i, $A(O_i)$ is the classifier accuracy, S_T is the number of features of the subset O_i, and L_T is the total number of the original feature. The parameter ω is used to controls the trade-offs between classification accuracy and feature cardinality. In this research, we set the value of ω to 0.90 for providing more weight on accuracy term.

3. VARIOUS TRANSFER FUNCTIONS USED WITH BOSA

A transfer function is used for mapping continuous space into binary space. It is observed that transfer function is an effective and straightforward approach to convert continuous values into binary one (Mafarja et al., 2018; Engelbrecht and Pampara, 2007). In this research, for mapping continuous values into binary values, three categories of transfer function namely, S-shaped (Kennedy and Eberhart, 1997; Mirjalili and Lewis, 2013), V-shaped (Mirjalili and Lewis, 2013), and quadratic (Jordehi, 2019) have been used. Each of the categories of V-shaped and S-shaped functions contains four independent functions, and Quadratic shaped transfer function contains three independent functions, making a total of 11 transfer functions. Throughout this paper, we will use the term quadratic to Q-shaped. The mathematical formulation of the transfer functions is illustrated in Table 1, and their shapes are shown in Fig. 2.

At iteration $(t+1)$ each step vector of binary owl $\Delta O_i(t+1) = [\Delta O_i^1, \Delta O_i^2, \dots \Delta O_i^d]$, in Equation (12) is in continuous d^{th} dimensional space. A transfer function $T(x)$ maps this continuous vector to binary vector $O_i(t+1) = [O_i^1, O_i^2, \dots O_i^d]$. Mapping method can be varied based on transfer functions. Therefore, for different type of transfer functions, the transformation of the real value of j^{th} dimension of i^{th} owl into binary value for iteration $(t+1)$ is as follows.

Table 1. Different transfer functions used for building BOSA model

Name	Transfer Functions	Functions Type
S1	$T(x) = \frac{1}{1+e^{-2x}}$	S-shaped
S2	$T(x) = \frac{1}{1+e^{-x}}$	S-shaped
S3	$T(x) = \frac{1}{1+e^{(-\frac{x}{2})}}$	S-shaped
S4	$T(x) = \frac{1}{1+e^{(-\frac{x}{3})}}$	S-shaped
V1	$T(x) = \lvert \mathrm{erf}\left(\frac{\sqrt{\pi}}{2}x\right) \rvert = \left\lvert \frac{\sqrt{2}}{\pi} \int_0^{\frac{\sqrt{\pi}}{2}x} e^{-t^2}\, dx \right\rvert$	V-shaped
V2	$T(x) = \lvert \tanh(x) \rvert$	V-shaped
V3	$T(x) = \lvert \frac{x}{\sqrt{1+x^2}} \rvert$	V-shaped
V4	$T(x) = \lvert \frac{2}{\pi} \arctan(\frac{\pi}{2}\mathrm{x}) \rvert$	V-shaped
Q1	$T(x) = \begin{cases} \left\lvert \left(\frac{x}{0.5x_{max}}\right) \right\rvert & if\ \lvert x \rvert < 0.5x_{max} \\ 1 & if\ \lvert x \rvert \geq 0.5x_{max} \end{cases}$	Quadratic
Q2	$T(x) = \begin{cases} \left\lvert \left(\frac{x}{0.5x_{max}}\right)^2 \right\rvert & if\ \lvert x \rvert < 0.5x_{max} \\ 1 & if\ \lvert x \rvert \geq 0.5x_{max} \end{cases}$	Quadratic
Q3	$T(x) = \begin{cases} \left\lvert \left(\frac{x}{0.5x_{max}}\right)^3 \right\rvert & if\ \lvert x \rvert < 0.5x_{max} \\ 1 & if\ \lvert x \rvert \geq 0.5x_{max} \end{cases}$	Quadratic

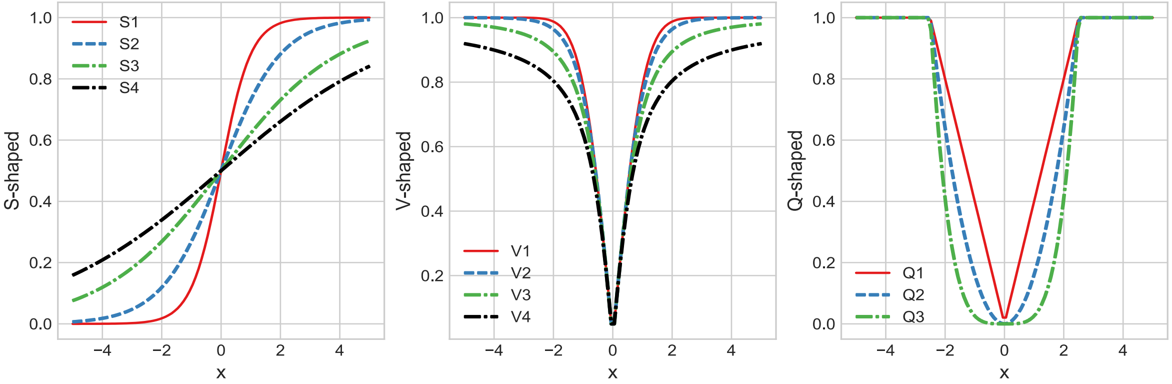

Fig. 2. The shape of different transfer functions.

3.1 Mapping methods for S-type function

S-type (S1 to S4) are all sigmoid functions with different slopes that transfer continuous values to probability values in the interval [0,1]. Then the corresponding binary value is determined based on the following Equation, where B is a uniform random number between 0 and 1. S-type is a commonly-used transfer function, and it forces the search agent to move to 0 or 1.

$$O_i^j(t+1) = \begin{cases} 1, & \text{if } B < T(\Delta O_i^j(t+1)) \\ 0, & \text{if } B \geq T(\Delta O_i^j(t+1)) \end{cases} \tag{14}$$

3.2 Mapping methods for V-type and Q-type function

V-type (V1-V4) functions are different, respectively based on their slope values. Like S-type, V-type function converts continuous values to probability values in the interval [0,1]. The binary value is determined using the following Equation, where B is a uniform random number between 0 and 1. Unlike S-type, in V-type, search agents are not forcefully moved to 0 or 1.

$$O_i^j(t+1) = \begin{cases} \neg O_i^j(t), & \text{if } B < T(\Delta O_i^j(t+1)) \\ O_i^j(t), & \text{if } B \geq T(\Delta O_i^j(t+1)) \end{cases} \tag{15}$$

Q-type (Q1 to Q3) are all quadratic functions. Their mapping procedure is similar to the V-type function. It is noted that the parameter x_{max} of each quadratic transfer function is assigned in this study to 5.0.

4. SIMULATION EXPERIMENT

In this work, the effect of different transfer functions on BOSA models is studied by simulation experiments with several benchmark data sets. This section describes the details of data sets and the parameter setting of the experimental study.

4.1 Datasets

Table 2 shows the details of the datasets considered in this study. Twenty datasets with the different number of features, instances and classes from the University of California, Irvine (UCI) repository (Dua and Graff, 2019) are selected. Since there are some missing values in some datasets, we replaced missing values of each continuous feature with means and missing values of each discrete feature with median values. Wrapper based feature selection approach is adopted here; all the datasets are partitioned into three sets using stratified random sampling. These partitions are 60% for training, 20% for testing, and the remaining 20% for the validation set. The training set is used to train the classification model, while the validation set is used to evaluate the model in the optimization phase. The testing set is kept aside during the feature selection process and is used for measuring final classification accuracy.

4.2 Parameter Settings

Eleven different types of transfer functions are used for comparison of respective BOSA models in this study. Besides, the best-proposed model is compared with three other state-of-the-art meta-heuristics algorithms including Harmony Search (HS) (Dash, 2018) algorithm, Binary Particle Swarm Optimization (BPSO) (Kumar, and Inbarani, 2017) and Binary Genetic Algorithm (BGA) (Guha et al., 2019) for feature selection problem. The number of search agents and the number of iterations of each algorithm is assigned to 20 and 100, respectively, in order to obtain fair comparison. Optimizer-specific parameters used in this experiment are reported in Table 3. It is noted that all of the settings are set based on trial and error on small empirical study. We use popular classification algorithm Support Vector Machine (SVM) with the linear kernel to evaluate the final classification performance of individual approaches. All of the experiments are repeated 20 times with different random seeds, and average statistical measures are recorded. We implemented all the algorithms with Python 3.7 and executed on a PC 2.30 GHz core-i5 CPU, 8.00 GB RAM and Windows 8 operating system.

Table 2. Datasets used

Datasets	No. of features	No. of instances	No. of classes
Arrhythmia	279	452	16
Breast-w	9	683	2
Clean1	166	476	2
Dermatology	34	358	6
Hepatitis	19	155	2
Ilpd	10	583	2
Libras-move	90	360	15
Lung-cancer	56	32	3
Parkinsons	22	195	2
Pendigits	16	10992	10
Promoters	57	106	2
Qsar-biodeg	41	1055	2
Semeion	256	1593	10
Sonar	60	208	2
Spambase	58	4601	2
Spect	22	267	2
Spectf	45	349	2
Vehicle	18	846	4
Wine	13	178	3
Wisconsin	30	569	2

Table 3. Same block size B and different embedded bit compare with block folding

Algorithm	Parameter	value
HS	Harmony memory consideration rate	0.3
	Pitch adjustment rate	0.9
BPSO	Inertia weight, w	0.1
	Acceleration coefficients, $C_1$and C_1	2
BGA	Mutation ratio	0.1
	Crossover ratio	0.9
BOSA	θ	2.0

5. EXPERIMENTAL RESULTS

To study the influence of eleven transfer functions on the performance of BOSA, in this section, we compare BOSA with eleven transfer functions on feature selection task. Then the performance of the best approach among the BOSA variants is compared with various state-of-the-art approaches. In Table 4 to Table 9, the best-reported result of each instance is highlighted in bold.

Table 4 shows the mean and standard deviation of fitness values of eleven BOSA variants on twenty datasets. As it is observed, 15 out of 20 datasets the mean fitness value of BOSA-Q3 is better than other variants of BOSA. The next successful BOSA variant regarding fitness value is BOSA-Q2 along with BOSA-S4. Both approaches show the best performances for four out of twenty datasets. It is clearly observed from the table that BOSA with both S-type and V-type does not produce better fitness value than Q-type transfer functions.

Table 5 shows the average classification accuracy and standard deviation results for eleven versions of BOSA algorithms. As it is observed, BOSA-Q3 outperforms other algorithms on five datasets, followed by both BOSA-V2 and BOSA-S1 on four datasets. There are two cases where BOSA-Q2 produces the best results. BOSA-V3 could not produce the best result for any dataset, whereas the rest of the approaches (BOSA-S2, BOSA-S3, BOSA-S4, BOSA-Q1, BOSA-Q2) show superiority only in one or two cases. It is evident that no method is able to produce the best classification accuracy on most of the datasets, but it can be said that in general, V-type and Q-type exhibit superiority over S-type.

Table 6 shows the average number of the selected features of the BOSA variations. Clearly, BOSA-Q3 outperforms other approaches in terms of reducing the number of features, as it shows high performance at 80% of datasets (16 out of 20 datasets). BOSA-S4 produces less number of features for *Libras-move, Qsar-biodeg* and *Spectf* datasets and BOSA-Q2 produces the best results for *Sonar* and *Lung-cancer* results. Overall, BOSA with Q-type produces less number of feature almost for every dataset, indicating its superiority over other competitors.

Table 7 shows the average CPU time (in seconds) of BOSA variants. BOSA-Q3 has the best computational time on nine datasets, while BOSA-S4 is in the second position that has the best computational time for six datasets. BOSA-S2 produces the best results in terms of CPU time only for 3 cases. V-type function takes more time than other approaches. S-type, in general, takes less computational time than Q3 and Q2.

From the above comparison between different variations of BOSA, it is clear that BOSA-Q3 has demonstrated high success regarding reducing the number of features for most of the datasets and it takes less CPU time than its other variants for around half of the datasets. Although BOSA-Q3 is not able to produce the best classification accuracy for most of the datasets, considering all of the performance criteria, BOSA with transfer function Q3 is the best model for feature section task. Therefore, in the rest of the paper, BOSA-Q3 is used to benchmark with three other state-of-the-art algorithms, namely BPSO, BGA, and HS.

Table 8 displays the results of average fitness values of BOSA-Q3 and the other three algorithms. According to the fitness values, in 9 out of 20 datasets, BOSA-Q3 produces the best performance. Similarly, BPSO also produces the best performances on nine datasets. Both HS and BGA exhibit superior performances only for two datasets, respectively. Clearly, BOSA-Q3 outperforms both BGA and HS and is competitive with BPSO when fitness results are considered.

Table 9 shows the comparison of BOSA-Q3 with other algorithms in terms of the average classification accuracy. BOSA-Q3 obtains better classification accuracy than other approaches since it produces the best results for 60% of the

Table 4. Mean (avg) and the standard deviation (std) of fitness values

Dataset	Measure	BOSA-S1	BOSA-S2	BOSA-S3	BOSA-S4	BOSA-V1	BOSA-V2	BOSA-V3	BOSA-V4	BOSA-Q1	BOSA-Q2	BOSA-Q3
Arrhythmia	avg	0.700	0.700	0.711	0.711	0.695	0.701	0.699	0.702	0.710	0.709	**0.718**
	std	0.007	0.014	0.014	0.019	0.013	0.016	0.016	0.020	0.020	0.011	0.013
Breast-w	avg	0.967	0.972	0.975	0.969	0.975	0.969	0.970	0.971	0.971	0.973	**0.979**
	std	0.003	0.004	0.005	0.003	0.007	0.003	0.005	0.005	0.003	0.003	0.004
Clean1	avg	0.898	0.910	0.908	0.910	0.906	0.903	0.906	0.910	0.911	0.910	**0.914**
	std	0.020	0.014	0.019	0.018	0.018	0.017	0.018	0.017	0.018	0.018	0.015
Dermatology	avg	0.967	0.970	0.974	**0.975**	0.965	0.971	0.966	0.968	0.969	0.973	**0.975**
	std	0.003	0.003	0.003	0.003	0.003	0.004	0.003	0.004	0.004	0.004	0.006
Hepatitis	avg	0.952	0.958	0.955	0.952	0.947	0.951	0.947	0.950	0.950	0.951	**0.961**
	std	0.013	0.012	0.011	0.008	0.012	0.013	0.014	0.013	0.009	0.015	0.012
Ilpd	avg	0.765	0.773	**0.782**	**0.782**	0.769	0.771	0.771	0.770	0.778	0.780	**0.782**
	std	0.011	0.017	0.010	0.013	0.014	0.014	0.013	0.009	0.007	0.010	0.012
Libras-move	avg	0.801	0.809	0.815	0.815	0.804	0.809	0.804	0.807	0.810	**0.818**	**0.818**
	std	0.020	0.023	0.032	0.032	0.025	0.025	0.023	0.032	0.027	0.019	0.019
Lung-cancer	avg	0.936	0.957	0.974	0.950	0.925	0.951	0.953	0.956	0.945	**0.975**	0.963
	std	0.057	0.042	0.003	0.055	0.062	0.043	0.043	0.043	0.056	0.003	0.042
Parkinsons	avg	0.979	0.976	0.967	0.979	0.971	0.971	0.974	0.972	0.977	**0.980**	**0.980**
	std	0.003	0.005	0.009	0.003	0.005	0.006	0.005	0.006	0.003	0.004	0.004
Pendigits	avg	0.958	0.958	0.958	0.958	0.958	**0.959**	0.958	0.958	0.958	0.958	0.958
	std	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.002
Promoters	avg	0.947	0.953	0.951	0.946	0.942	0.948	0.952	0.943	0.943	0.946	**0.973**
	std	0.022	0.023	0.029	0.026	0.021	0.022	0.021	0.022	0.027	0.018	0.012
Qsar-biodeg	avg	0.858	0.866	0.871	**0.872**	0.857	0.859	0.859	0.862	0.868	0.868	0.871
	std	0.007	0.009	0.005	0.005	0.006	0.006	0.007	0.007	0.007	0.008	0.006
Semeion	avg	0.899	0.898	0.899	0.899	0.900	0.899	0.902	0.898	0.907	0.899	**0.908**
	std	0.005	0.004	0.006	0.006	0.006	0.004	0.004	0.007	0.011	0.004	0.006
Sonar	avg	0.908	0.909	0.922	0.923	0.905	0.909	0.914	0.914	0.921	0.911	**0.924**
	std	0.019	0.019	0.020	0.015	0.016	0.019	0.013	0.018	0.024	0.009	0.018
Spambase	avg	0.891	0.896	0.899	0.898	0.894	0.893	0.893	0.896	0.901	**0.910**	0.897
	std	0.004	0.003	0.003	0.004	0.004	0.006	0.006	0.005	0.005	0.005	0.006
Spect	avg	0.816	0.817	0.826	0.827	0.812	0.811	0.810	0.817	0.818	0.825	**0.837**
	std	0.019	0.019	0.022	0.016	0.016	0.015	0.017	0.017	0.017	0.020	0.014
Spectf	avg	0.854	0.869	**0.873**	**0.873**	0.851	0.853	0.863	0.858	0.867	0.866	0.868
	std	0.022	0.009	0.019	0.011	0.018	0.025	0.023	0.019	0.010	0.013	0.015
Vehicle	avg	0.723	0.734	0.736	0.737	0.725	0.726	0.727	0.725	0.737	0.739	**0.743**
	std	0.013	0.011	0.013	0.014	0.009	0.010	0.010	0.014	0.015	0.018	0.016
Wine	avg	0.967	0.975	0.981	0.982	0.963	0.960	0.965	0.966	0.975	0.982	**0.987**
	std	0.010	0.009	0.004	0.004	0.011	0.013	0.012	0.010	0.008	0.004	0.004
Wisconsin	avg	0.951	0.959	0.962	0.963	0.952	0.952	0.953	0.955	0.960	0.960	**0.964**
	std	0.006	0.006	0.007	0.007	0.005	0.005	0.004	0.005	0.005	0.006	0.004

datasets. BOSA-Q3 also outperforms HS for around 80% of datasets, BPSO and BGA for around 75% of datasets. Besides, the standard deviation of classification accuracy of BOSA-Q3 for many datasets is minimum, indicating the robustness of BOSA-Q3.

Table 5. Mean (avg) and the standard deviation (std) of classification accuracy

Dataset	Measure	BOSA-S1	BOSA-S2	BOSA-S3	BOSA-S4	BOSA-V1	BOSA-V2	BOSA-V3	BOSA-V4	BOSA-Q1	BOSA-Q2	BOSA-Q3
Arrhythmia	avg	63.676	63.309	62.500	60.882	63.235	64.118	62.059	62.353	60.294	62.868	**64.282**
	std	5.120	4.858	3.325	2.495	3.937	5.044	4.482	4.327	5.810	4.333	2.664
Breast-w	avg	96.390	96.195	95.610	95.707	**96.537**	95.951	96.049	96.049	96.244	95.122	96.488
	std	1.177	1.655	1.781	2.015	1.426	1.869	1.817	1.269	1.302	1.380	1.145
Clean1	avg	78.951	78.671	76.783	75.664	79.580	77.832	78.741	**80.140**	76.434	75.594	76.084
	std	3.135	2.841	3.756	3.359	4.296	3.915	3.931	3.689	3.058	3.851	3.593
Dermatology	avg	96.481	96.574	95.556	95.556	96.574	**97.593**	96.667	96.759	96.389	95.833	95.833
	std	1.499	2.096	2.822	2.719	2.547	1.461	1.324	1.813	1.269	1.964	1.915
Hepatitis	avg	80.638	**81.489**	79.362	80.213	80.213	78.085	79.574	79.574	79.362	78.936	80.000
	std	4.536	5.853	5.312	4.261	5.766	3.180	5.779	2.497	3.760	2.116	1.488
Ilpd	avg	**71.429**	**71.429**	**71.429**	**71.429**	**71.429**	71.371	71.371	**71.429**	**71.429**	**71.429**	**71.429**
	std	0.000	0.000	0.000	0.000	0.000	0.181	0.181	0.000	0.000	0.000	0.000
Libras-move	avg	64.352	63.889	62.685	62.870	64.815	64.722	64.815	64.722	**65.833**	63.704	63.704
	std	4.147	4.860	4.920	5.004	5.556	4.287	4.187	4.353	4.770	5.966	5.413
Lung-cancer	avg	49.000	47.000	52.000	43.000	53.000	42.000	45.000	48.000	**56.000**	52.000	51.000
	std	7.379	13.375	13.984	10.593	9.487	18.738	14.337	9.189	12.649	11.353	11.972
Parkinsons	avg	**87.288**	85.763	86.780	85.593	85.763	86.271	86.610	85.085	85.593	84.915	85.932
	std	3.769	5.312	3.815	2.680	5.372	4.757	3.786	4.136	6.552	5.445	5.181
Pendigits	avg	95.385	95.018	94.688	94.257	95.079	**95.503**	95.421	94.870	94.842	94.491	94.873
	std	0.493	1.208	1.250	1.555	0.736	0.643	1.314	1.436	1.026	0.920	1.088
Promoters	avg	73.750	70.000	73.125	74.375	73.750	72.188	78.750	75.000	73.750	73.438	**79.375**
	std	5.929	7.247	7.683	9.638	6.110	6.823	7.483	7.933	10.013	5.938	7.683
Qsar-biodeg	avg	84.448	83.596	83.407	82.902	**85.331**	85.110	85.110	83.565	83.817	83.659	82.776
	std	1.822	1.271	1.940	1.558	1.540	1.880	1.530	2.031	1.093	1.572	2.135
Semeion	avg	**92.343**	91.632	90.795	90.586	92.176	92.008	92.113	91.674	91.255	90.941	91.423
	std	0.947	1.098	1.154	1.240	0.900	1.251	0.996	1.333	0.955	1.175	0.946
Sonar	avg	74.444	73.333	71.587	71.746	73.175	**75.238**	73.016	74.603	73.810	72.857	74.603
	std	7.574	5.333	3.991	4.016	7.116	7.185	7.989	6.776	6.136	6.236	4.030
Spambase	avg	91.390	90.956	90.891	90.565	91.644	91.513	91.723	91.383	91.665	**91.818**	91.253
	std	1.574	1.023	1.091	0.866	0.984	1.141	0.842	0.826	0.963	0.791	1.385
Spect	avg	70.000	70.617	71.852	70.000	70.617	69.877	71.975	70.988	72.099	72.099	**73.210**
	std	3.590	5.422	5.929	5.239	5.231	4.164	3.949	4.287	4.040	4.478	4.034
Spectf	avg	81.810	80.667	81.905	81.238	80.571	81.143	81.714	80.762	81.333	81.238	**82.190**
	std	4.293	3.361	2.618	3.047	2.298	3.909	3.231	3.231	4.334	2.696	2.502
Vehicle	avg	73.465	73.504	72.283	71.378	74.921	**75.236**	73.465	74.843	71.614	72.992	72.441
	std	2.684	1.929	3.813	3.623	2.851	3.369	4.024	2.925	3.787	2.035	3.209
Wine	avg	92.963	92.407	89.444	90.926	94.074	93.333	93.704	**94.259**	93.333	91.111	92.037
	std	3.123	3.948	2.626	2.538	3.785	3.172	3.825	3.202	3.172	3.884	3.497
Wisconsin	avg	**95.731**	**95.731**	95.322	95.439	95.380	95.497	95.497	95.439	95.263	95.673	95.380
	std	1.104	1.561	1.322	1.257	1.335	1.486	0.996	1.454	1.184	1.002	0.700

Table 6. Mean (avg) and the standard deviation (std) of the cardinality of the feature subset

Dataset	Measure	BOSA-S1	BOSA-S2	BOSA-S3	BOSA-S4	BOSA-V1	BOSA-V2	BOSA-V3	BOSA-V4	BOSA-Q1	BOSA-Q2	BOSA-Q3
Arrhythmia	avg	163.600	129.400	108.600	105.500	154.400	151.400	149.800	140.500	124.200	107.300	**104.100**
	std	6.769	10.658	4.600	6.223	20.326	17.702	14.117	11.424	6.339	8.551	5.425
Breast-w	avg	5.300	4.500	3.600	3.600	4.800	4.500	4.700	4.300	3.600	3.400	**3.100**
	std	0.949	0.850	1.430	1.265	0.789	1.179	1.567	1.059	0.843	0.699	0.738
Clean1	avg	99.200	101.100	91.200	87.400	123.600	122.300	116.300	113.100	98.900	85.200	**84.900**
	std	10.560	11.445	8.954	7.734	19.529	18.270	16.813	16.059	10.429	5.996	6.557
Dermatology	avg	15.200	20.100	17.400	17.000	23.800	22.800	22.200	21.000	19.900	18.200	**14.400**
	std	1.398	1.729	1.713	1.826	2.201	2.573	2.201	2.789	2.132	2.658	3.273
Hepatitis	avg	14.000	11.400	10.600	9.400	14.700	13.000	13.600	12.500	10.200	8.700	**7.700**
	std	1.764	1.430	2.171	2.011	1.567	1.764	1.430	2.173	1.874	2.214	1.418
Ilpd	avg	6.000	4.900	4.100	4.300	6.900	6.400	5.700	5.800	5.200	4.400	**3.300**
	std	1.886	1.663	1.449	1.059	1.853	2.221	2.263	1.549	1.814	1.506	1.337
Libras-move	avg	67.100	56.500	45.700	**45.000**	64.800	65.500	63.200	61.100	56.600	51.500	**45.000**
	std	9.134	7.619	2.830	5.270	9.750	8.489	5.116	5.705	5.522	6.948	1.826
Lung-cancer	avg	40.800	32.800	29.200	27.500	38.700	39.300	37.000	34.300	31.100	**25.500**	25.900
	std	2.974	2.781	3.011	3.375	6.147	3.622	3.055	1.947	1.969	3.837	3.843
Parkinsons	avg	13.500	10.700	9.100	9.300	12.700	12.700	11.600	12.300	10.300	9.000	**7.800**
	std	2.415	2.163	1.449	1.418	2.003	2.710	2.066	2.584	1.252	1.563	1.398
Pendigits	avg	10.300	10.200	9.700	9.500	9.900	10.400	10.500	10.200	9.900	9.600	**9.200**
	std	0.675	0.632	0.675	0.707	0.738	0.699	0.707	1.033	0.568	0.516	0.632
Promoters	avg	36.900	35.300	32.500	28.200	38.300	38.100	40.100	36.400	31.700	28.300	**26.000**
	std	5.915	2.983	2.461	3.553	6.093	5.065	4.228	4.377	2.830	3.368	2.867
Qsar-biodeg	avg	29.200	23.600	21.800	**18.500**	30.000	29.400	29.000	24.700	21.600	20.400	18.800
	std	4.590	3.373	2.898	3.064	2.357	1.955	2.309	3.713	2.413	3.062	3.048
Semeion	avg	203.500	164.000	147.300	145.300	204.900	201.300	194.700	183.300	159.200	144.300	**132.500**
	std	5.104	6.532	6.993	5.964	9.632	9.978	8.744	12.437	7.829	4.296	9.095
Sonar	avg	45.300	37.000	31.700	29.800	43.500	44.200	41.100	41.500	33.100	**28.800**	29.400
	std	4.762	2.906	2.830	4.237	6.329	5.789	3.784	3.749	2.767	4.417	3.950
Spambase	avg	35.300	33.600	31.400	31.400	36.800	36.700	36.700	35.700	33.000	29.900	**28.700**
	std	6.961	3.062	4.248	5.317	3.994	5.736	5.926	4.620	4.082	3.446	2.791
Spect	avg	16.900	12.500	**11.000**	11.400	16.200	16.700	16.400	14.700	13.100	11.200	**11.000**
	std	1.853	3.100	2.749	1.647	3.190	2.163	1.430	1.889	2.424	2.150	2.055
Spectf	avg	29.700	25.000	23.400	**20.400**	28.300	29.600	29.100	27.900	24.800	20.900	21.400
	std	5.438	4.346	2.011	4.006	4.423	5.147	4.818	4.841	3.360	3.381	3.565
Vehicle	avg	11.500	10.000	8.900	8.900	12.500	12.100	11.200	11.500	10.000	9.400	**7.800**
	std	1.780	1.826	1.101	1.370	1.650	1.197	2.150	1.509	1.886	1.838	0.919
Wine	avg	7.200	5.900	4.900	4.800	8.100	7.200	6.900	7.300	5.800	4.800	**4.400**
	std	1.229	1.197	1.101	1.033	1.101	1.398	1.524	1.059	1.549	0.919	0.843
Wisconsin	avg	13.300	15.600	13.000	13.400	20.300	19.500	18.800	18.300	14.900	14.800	**12.200**
	std	2.669	2.591	1.944	2.221	2.627	3.028	3.521	2.003	1.853	2.201	2.440

Table 7. Mean (avg) and the standard deviation (std) of CPU time (in seconds)

Dataset	Measurese	BOSA-S1	BOSA-S2	BOSA-S3	BOSA-S4	BOSA-V1	BOSA-V2	BOSA-V3	BOSA-V4	BOSA-Q1	BOSA-Q2	BOSA-Q3
Arrhythmia	avg	23.040	20.416	18.685	**18.235**	23.929	23.211	22.547	21.746	20.888	19.448	18.966
	std	0.118	0.387	0.033	0.039	0.152	0.111	0.148	0.069	0.076	0.065	0.066
Breast-w	avg	9.146	8.981	8.879	8.872	9.171	9.150	9.104	9.102	8.994	8.890	**8.678**
	std	0.030	0.046	0.040	0.051	0.033	0.031	0.021	0.058	0.048	0.020	0.020
Clean1	avg	15.726	14.079	13.511	**13.349**	16.458	16.117	15.632	15.122	14.810	14.107	13.841
	std	0.092	0.050	0.199	0.420	0.045	0.054	0.045	0.053	0.054	0.047	0.056
Dermatology	avg	5.789	**5.688**	5.847	5.703	5.924	5.867	5.833	5.790	5.885	5.858	5.862
	std	0.051	0.020	0.167	0.021	0.088	0.011	0.015	0.015	0.020	0.026	0.045
Hepatitis	avg	3.648	**3.612**	3.616	3.654	3.730	3.702	3.682	3.669	3.733	3.718	3.718
	std	0.034	0.010	0.013	0.052	0.029	0.016	0.012	0.008	0.026	0.012	0.016
Ilpd	avg	7.377	7.326	7.248	**7.206**	7.456	7.455	7.382	7.368	7.316	7.276	7.270
	std	0.061	0.097	0.033	0.021	0.078	0.167	0.031	0.076	0.026	0.034	0.022
Libras-move	avg	8.951	8.414	**8.221**	8.295	9.404	9.255	9.133	8.963	8.949	8.690	8.663
	std	0.062	0.030	0.058	0.088	0.053	0.024	0.065	0.060	0.085	0.035	0.055
Lung-cancer	avg	3.237	**2.900**	2.912	2.909	3.129	3.085	3.066	3.029	3.248	3.255	2.915
	std	0.018	0.048	0.008	0.017	0.023	0.016	0.024	0.039	0.031	0.046	0.033
Parkinsons	avg	4.084	4.035	**4.034**	4.036	4.202	4.273	4.135	4.101	4.195	4.147	4.136
	std	0.018	0.015	0.023	0.018	0.057	0.143	0.018	0.008	0.065	0.014	0.010
Pendigits	avg	204.586	175.241	159.685	155.299	205.658	202.373	195.152	187.541	169.338	155.824	**151.123**
	std	1.901	0.961	0.779	1.077	1.810	1.839	1.406	1.102	1.052	0.900	0.715
Promoters	avg	4.273	4.168	4.160	4.161	4.499	4.460	4.408	4.332	4.505	4.475	**3.688**
	std	0.014	0.011	0.015	0.026	0.020	0.033	0.019	0.009	0.017	0.011	0.032
Qsar-biodeg	avg	17.215	16.066	15.465	**15.242**	17.514	17.264	16.935	16.698	16.085	15.551	15.315
	std	0.407	0.283	0.189	0.192	0.440	0.404	0.359	0.332	0.249	0.188	0.161
Semeion	avg	156.644	130.990	126.505	112.906	158.517	155.123	149.256	142.581	134.055	119.548	**112.028**
	std	0.411	0.208	0.262	0.612	0.320	0.155	0.348	0.244	6.631	1.544	0.172
Sonar	avg	6.074	5.852	6.355	5.896	6.234	6.768	6.270	6.192	6.223	6.669	**5.556**
	std	0.153	0.145	0.557	0.310	0.111	0.488	0.200	0.384	0.193	0.447	0.121
Spambase	avg	86.085	85.183	85.565	85.477	85.602	87.562	85.713	87.474	85.425	89.540	**54.015**
	std	1.312	1.165	1.269	1.110	0.506	1.239	1.014	1.972	0.608	4.551	1.619
Spect	avg	6.450	4.917	5.087	**4.765**	5.163	4.891	5.046	5.114	5.065	5.387	5.215
	std	1.050	0.283	0.499	0.281	0.320	0.238	0.335	0.369	0.211	0.303	0.260
Spectf	avg	7.731	7.327	7.488	7.418	7.953	7.731	6.756	6.635	6.618	6.505	**6.250**
	std	0.428	0.396	0.270	0.158	0.671	0.565	0.110	0.115	0.071	0.041	0.031
Vehicle	avg	13.117	12.446	11.673	11.903	13.290	12.296	12.666	12.304	12.038	11.662	**11.197**
	std	0.737	0.537	0.302	0.442	0.782	0.579	0.634	0.546	0.245	0.577	0.173
Wine	avg	4.405	3.894	3.844	4.652	4.167	4.369	4.227	4.411	4.718	4.329	**3.541**
	std	0.377	0.222	0.158	0.480	0.441	0.418	0.151	0.127	0.466	0.253	0.024
Wisconsin	avg	10.411	10.313	9.573	**9.248**	10.458	9.897	9.649	9.983	10.060	10.433	9.934
	std	0.641	1.069	0.450	0.331	0.998	0.252	0.412	0.553	0.688	0.612	0.454

Table 8. Average fitness and standard deviation of BOSA-Q3 compared to three other algorithms

Dataset	HS		BPSO		BGA		BOSA-Q3	
	Avg	std	Avg	std	Avg	std	Avg	std
Arrhythmia	0.692	0.018	0.705	0.016	0.705	0.013	**0.718**	0.013
Breast-w	0.965	0.006	**0.979**	0.006	0.975	0.004	**0.979**	0.004
Clean1	0.862	0.027	**0.919**	0.013	0.914	0.016	0.914	0.015
Dermatology	0.951	0.008	0.977	0.004	**0.978**	0.002	0.975	0.006
Hepatitis	0.913	0.023	0.951	0.010	0.950	0.015	**0.961**	0.012
Ilpd	0.750	0.013	**0.787**	0.013	0.781	0.017	0.782	0.012
Libras-move	0.738	0.029	0.814	0.030	0.813	0.020	**0.818**	0.019
Lung-cancer	0.803	0.090	**0.968**	0.043	0.963	0.042	0.963	0.042
Parkinsons	0.950	0.011	**0.982**	0.003	0.978	0.003	0.980	0.004
Pendigits	**0.960**	0.001	0.959	0.001	0.952	0.002	0.958	0.002
Promoters	0.899	0.012	0.968	0.016	0.951	0.022	**0.973**	0.012
Qsar-biodeg	0.846	0.007	**0.878**	0.006	0.875	0.010	0.871	0.006
Semeion	0.872	0.009	0.901	0.009	0.901	0.006	**0.908**	0.006
Sonar	0.878	0.033	**0.929**	0.014	0.922	0.022	0.924	0.018
Spambase	0.896	0.005	0.907	0.005	0.905	0.009	**0.897**	0.006
Spect	0.768	0.022	0.826	0.014	0.825	0.010	**0.837**	0.014
Spectf	**0.887**	0.018	0.874	0.020	0.876	0.016	0.868	0.015
Vehicle	0.737	0.014	0.752	0.013	**0.754**	0.004	0.743	0.016
Wine	0.970	0.012	**0.987**	0.005	0.985	0.003	**0.987**	0.004
Wisconsin	0.954	0.008	**0.969**	0.004	0.965	0.006	0.964	0.004

Table 9. Average classification accuracies and standard deviation of BOSA-Q3 compared to three other algorithms

Dataset	HS		BPSO		BGA		BOSA-Q3	
	Avg	std	Avg	std	Avg	std	Avg	std
Arrhythmia	60.053	1.974	60.735	2.552	61.985	4.013	**64.282**	2.664
Breast-w	95.854	1.709	**96.488**	0.970	96.439	1.437	**96.488**	1.145
Clean1	75.944	5.625	75.105	3.233	**77.762**	3.926	76.084	3.593
Dermatology	95.000	1.525	**96.296**	2.138	**96.296**	1.691	95.833	1.915
Hepatitis	78.298	4.789	78.936	3.242	78.936	2.547	**80.000**	1.488
Ilpd	**71.429**	0.000	**71.429**	0.000	**71.429**	0.000	**71.429**	0.000
Libras-move	61.944	4.770	63.611	4.920	**64.444**	5.420	63.704	5.413
Lung-cancer	48.000	12.293	43.000	15.670	44.000	17.127	**51.000**	11.972
Parkinsons	85.763	3.591	**86.610**	2.704	85.932	4.156	85.932	5.181
Pendigits	**95.024**	1.081	94.867	0.917	95.361	0.729	94.873	1.088
Promoters	70.625	5.743	77.813	7.134	74.063	5.522	**79.375**	7.683
Qsar-biodeg	82.965	2.379	83.880	1.221	**84.006**	1.828	82.776	2.135
Semeion	89.854	1.671	90.251	1.108	90.669	1.480	**91.423**	0.946
Sonar	73.492	5.701	71.270	5.100	73.333	4.780	**74.603**	4.030
Spambase	89.768	1.632	90.905	1.099	91.195	1.164	**91.253**	1.385
Spect	69.877	4.627	70.370	6.886	72.346	3.502	**73.210**	4.034
Spectf	80.000	1.489	81.429	4.473	81.524	5.083	**82.190**	2.502
Vehicle	72.008	3.620	71.890	3.831	**72.913**	3.054	72.441	3.209
Wine	**92.778**	2.217	91.667	3.519	92.037	3.272	92.037	3.497
Wisconsin	95.263	1.444	95.146	1.830	95.205	1.804	**95.380**	0.700

Fig. 3 presents a comparison of BOSA-Q3 with HS, BPSO and BGA in terms of the average number of selected features. We estimate the feature selection in percentage using the following Equation:

$$Feature\ \ Selection = \frac{number\ \ of\ \ selected\ \ features}{Total\ \ Number\ \ of\ \ features} \times 100 \tag{16}$$

From the Fig., it is found that for 11 out of 20 datasets, BOSA-Q3 can select less percentage of features than the other three methods. The next best approach is BPSO that manages to find the least percentage of features for six datasets. Besides, BOSA-Q3 has shown a noteworthy superiority over BGA algorithms on the majority of the datasets.

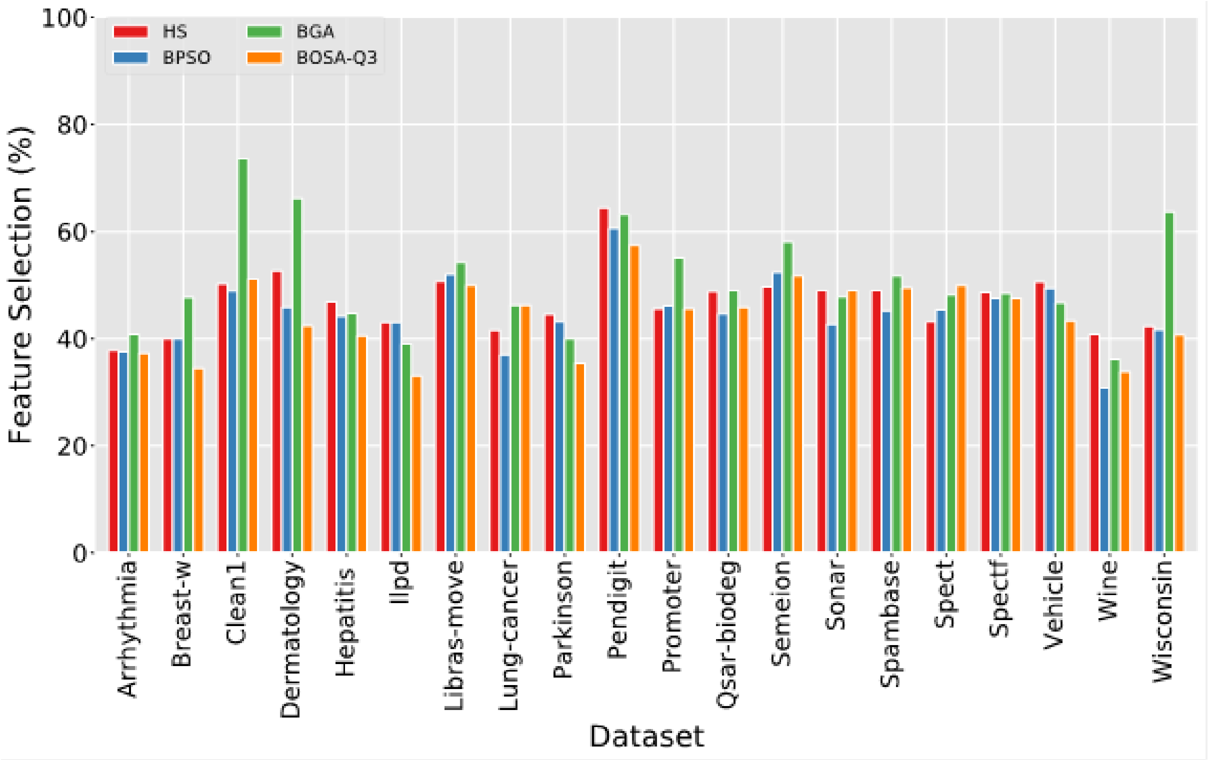

Fig. 3 Average feature selection rate of BOSA-Q3 compared to other algorithms

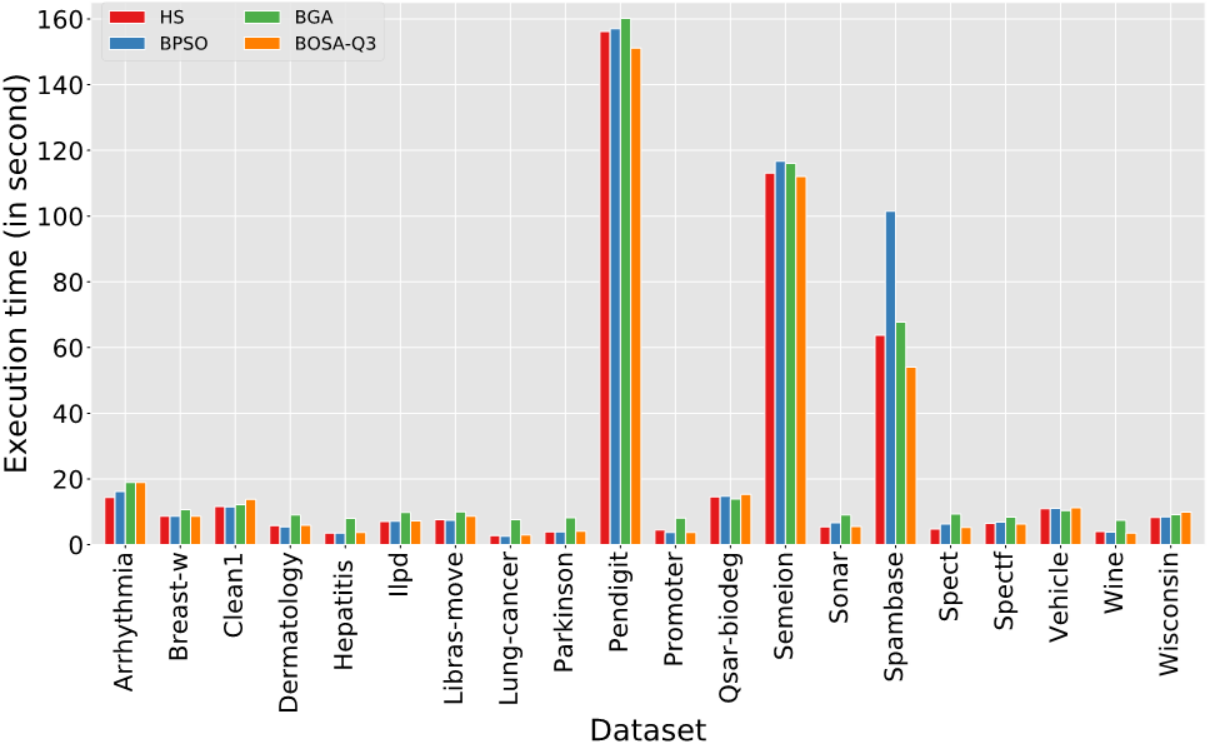

Fig. 4. Average CPU time of BOSA-Q3 compared to other algorithms

Table 10. Overall comparison among the methods

Methods	Average fitness	Average classification accuracy	Average no. of feature selected
HS	0.865	79.20 %	44.73%
BPSO	0.901	79.58 %	44.68%
BGA	0.898	79.99 %	50.77%
BOSA-Q3	**0.901**	**80.71 %**	**44%**

Table 11. p-values of Wilcoxon's matched-pairs test results

Comparison	Average fitness	Average classification accuracy	Average no. of feature selected
BOSA-Q3 vs HS	**0.00034**	**0.00148**	**0.02491**
BOSA-Q3 vs BPSO	0.63036	**0.00956**	0.51966
BOSA-Q3 vs BGA	0.16319	0.19299	**0.00029**

Fig. 4 illustrates the average computational time of BOSA-Q3 compared with HS, GA, and BPSO. As per results in Fig. 4, BOSA-Q3 produces less time than other approaches in solving datasets *Penigit*, *Semeion*, and *Spambase*. They all have either a large number of features or instances and usually take more CPU time. It is also observed that, regarding CPU time, BOSA-Q3 outperforms BGA for most of the datasets. When BOSA-Q3 is compared with BPSO and HS, the time difference between the algorithms is not vast on the majority of the datasets.

Table 10 shows the summary results for the datasets using the BOSA-Q3 and other three approaches, with the best results being in boldface. Based on the results, we can conclude that BOSA-Q3 performs better than other meta-heuristic approaches in terms of average fitness, average classification accuracy, and the average no. of feature selected. Moreover, Wilcoxon's matched-pairs signed-ranks test for all the methods regarding average fitness, average classification accuracy, and the average feature selection are reported in Table 11. The significance level is considered to be 0.05, and the significant results ($P<0.05$) are highlighted in bold. It is observed that in the case of average fitness BOSA-Q3 is significantly better than HS while BOSA-Q3 shows statistically better average classification accuracy compared to both HS and PBSO. BOSA-Q3 is also significantly better than BGA and HS as regards average feature selection rate. We can also observe that, for any of the performance measure, the results of BOSA-Q3 is not inferior to that of other approaches.

6. CONCLUSION

Feature subset selection is an essential preprocessing task for any pattern recognition or data mining application. But optimal feature subset selection from a large number of possibilities is a computation-intensive problem, and the computational complexity increases with the increase of dimensionality of the data. Meta-heuristic algorithms are extensively used for finding out near-optimal solutions in case of the difficult optimization problem and popularly used for feature subset selection.

In this paper, we have intensively studied the performance of our proposed Binary Owl Search Algorithm (BOSA), modification of a recently developed metaheuristic algorithm named Owl Search Algorithm, for feature subset selection problem. Eleven transfer functions of three varieties are used to map the continuous solution space to binary solution space, and those functions have been combined to develop 11 BOSA models. The models are studied for evaluation of their performances in terms of accuracy of classification, reduction of features and computational cost by simulation experiments with benchmark data sets from UCI repository.

Experimental results show that BOSA with Q-type and V-type functions can perform well compared to S-type functions in terms of classification accuracy. Besides Q-type function can reduce the feature a lot. The better performance of V-type and Q-type functions over S-type can be attributed to the fact that they do not force the search element to take the value 0 or 1 unlike S-type function. S-type function is, however, comparatively faster than the other two variants. It induces the complement of the position only if the step value is comparatively high. Among all of the eleven BOSA models, BOSA with Q3 performed the best based on all the performance metrics. Finally, BOSA-Q3 has been compared with state-of-the-art population-based meta-heuristics approaches BPSO, HS, and BGA. Results show that for most of the datasets BOSA-Q3 is able to reduce the feature cardinality, improve the classification accuracy, and reduce the computational complexity in comparison with other popular meta-heuristics feature selection approaches. The reason for BOSA-Q3's better performance is likely to be attributed to the proper balancing between exploration and exploitation capabilities of the proposed approach, which intensively examines the promising region of the solution space.

In future investigations, it might be possible to use a multiobjective BOSA for wrapper based or filter-based feature selection. Another possible extension of this work could be examining the behavior of BOSA in addressing feature selection for high dimensional datasets such as in the area of text and biomedical data.

ACKNOWLEDGEMENT

This work was supported by JSPS Bilateral Open Partnership Joint Research Project.

REFERENCES

Aghdam, M.H., Ghasem-Aghaee, N., Basiri, M.E. 2009. Text feature selection using ant colony optimization. Expert Systems with Applications, 36, part 2, 6843–6853.

Allam, M. Nandhini, M. 2018. Optimal feature selection using binary teaching learning based optimization algorithm. Journal of King Saud University - Computer and Information Sciences. doi:10.1016/j.jksuci. 2018. 12.001.

Al-Tashi, Q., Abdul Kadir, S.J., Rais, H.M., Mirjalili, S., Alhussian, H. 2019. Binary optimization using hybrid grey wolf optimization for feature selection. IEEE Access, 7, 39496–39508.

Arora, S., Anand, P. 2019. Binary butterfly optimization approaches for feature selection. Expert Systems with Applications, 116, 147–160.

Banka, H., Dara, S. 2015. A hamming distance based binary particle swarm optimization (HDBPSO) algorithm for high dimensional feature selection, classification and validation. Pattern Recognition Letters, 52, 94–100.

BAŞ, E., ÜLKER, E. 2020. An efficient binary social spider algorithm for feature selection problem. Expert Systems with Applications, 146, 113185.

Bolón-Canedo, V., Sánchez-Maroño, N., Alonso-Betanzos, A. 2015. Recent advances and emerging challenges of feature selection in the context of big data. Knowledge-Based Systems, 86, 33–45.

Cao, Y., Wang, Q., Wang, Z., Jermsittiparsert, K., Shafiee, M. 2020. A new optimized configuration for capacity and operation improvement of CCHP system based on developed owl search algorithm. Energy Reports, 6, 315–324.

Chakraborty, B. 2014. Rough fuzzy consistency measure with evolutionary algorithm for attribute reduction. In 2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 723–728.

Chandrashekar, G., Sahin, F. 2014. A survey on feature selection methods. Computers and Electrical Engineering, 40, 16–28.

Chuang, L.Y., Tsai, S.W., Yang, C.H. 2011. Improved binary particle swarm optimization using catfish effect for feature selection. Expert Systems with Applications, 38, 12699–12707.

Crawford, B., Soto, R., Astorga, G., García, J., Castro, C., Paredes, F. 2017. Putting continuous metaheuristics to work in binary search spaces. Complexity, ed. Hao, J., 2017, 8404231.

Dash, M., Liu, H. 2003. Consistency-based search in feature selection. Artificial Intelligence, 151, 155–176.

Dash, R. 2018. An adaptive harmony search approach for gene selection and classification of high dimensional medical data. Journal of King Saud University - Computer and Information Sciences. doi:10.1016/j.jksuci. 2018.02.013.

Dua, D., Graff, C. 2019. UCI machine learning repository, 2017. URL http://archive. ics. uci. edu/ml, 37.

El-Ashmawi, W.H., Abd Elminaam, D.S., Nabil, A.M., Eldesouky, E. 2020. A chaotic owl search algorithm based bilateral negotiation model. Ain Shams Engineering Journal. doi:10.1016/j.asej. 2020.01.005.

Emary, E., Zawbaa, H.M., Hassanien, A.E. 2016. Binary ant lion approaches for feature selection. Neurocomputing, 213, 54–65.

Emary, E., Zawbaa, H.M., Hassanien, A.E. 2016. Binary grey wolf optimization approaches for feature selection. Neurocomputing, 172, 371–381.

Engelbrecht, A.P., Pampara, G. 2007. Binary differential evolution strategies. In 2007 IEEE Congress on Evolutionary Computation, 1942–1947.

Ewees, A.A., El Aziz, M.A., Hassanien, A.E. 2019. Chaotic multi-verse optimizer-based feature selection. Neural Computing and Applications, 31, 991–1006.

Farhan, A.F., Feilat, E.A., Al-Salaymeh, A.S. 2019. Maximum power point tracking technique using combined perturb & observe and owl search algorithms. In 2019 International Conference on Electrical and Computing Technologies and Applications, ICECTA 2019, 4–8. doi:10.1109/ICECTA48151.2019.8959738.

Faris, H., Mafarja, M.M., Heidari, A.A., Aljarah, I., Al-Zoubi, A.M., Mirjalili, S., Fujita, H. 2018. An efficient binary Salp Swarm Algorithm with crossover scheme for feature selection problems. Knowledge-Based Systems, 154, 43–67.

Fausto, F., Reyna-Orta, A., Cuevas, E., Andrade, Á.G., Perez-Cisneros, M. 2019. From ants to whales: metaheuristics for all tastes. Artificial Intelligence Review, Springer Netherlands. doi:10.1007/s10462-018-09676-2.

Guha, R., Ghosh, M., Kapri, S., Shaw, S., Mutsuddi, S., Bhateja, V., Sarkar, R. 2019. Deluge based Genetic Algorithm for feature selection. Evolutionary Intelligence. doi:10.1007/s12065-019-00218-5.

Han, C., Zhou, G., Zhou, Y. 2019. Binary Symbiotic Organism Search Algorithm for Feature Selection and Analysis. IEEE Access, 7, 166833–166859.

Inbarani, H.H., Bagyamathi, M., Azar, A.T. 2015. A novel hybrid feature selection method based on rough set and improved harmony search. Neural Computing and Applications, 26, 1859–1880.

Jain, A.K., Murty, M.N., Flynn, P.J. 1999. Data clustering: a review. ACM computing surveys (CSUR), 31, 264–323.

Jain, M., Maurya, S., Rani, A., Singh, V. 2018. Owl search algorithm: A novel nature-inspired heuristic paradigm for global optimization. Journal of Intelligent and Fuzzy Systems, 34, 1573–1582.

Jordehi, A.R. 2019. Binary particle swarm optimization with quadratic transfer function: A new binary optimization algorithm for optimal scheduling of appliances in smart homes. Applied Soft Computing, 78, 465–480.

Kabir, M.M., Shahjahan, M., Murase, K. 2012. A new hybrid ant colony optimization algorithm for feature selection. Expert Systems with Applications, 39, 3747–3763.

Kennedy, J., Eberhart, R.C. 1997. A discrete binary version of the particle swarm algorithm. In 1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation, 4104–4108.

Khushaba, R.N., Al-Ani, A., Al-Jumaily, A. 2011. Feature subset selection using differential evolution and a statistical repair mechanism. Expert Systems with Applications, 38, 11515–11526.

Kumar, U.S., Inbarani, H.H. 2017. PSO-based feature selection and neighborhood rough set-based classification for BCI multiclass motor imagery task. Neural Computing and Applications, 28, 3239–3258.

Li, D., Deng, L., Su, Q., Song, Y. 2020. Providing a guaranteed power for the BTS in telecom tower based on improved balanced owl search algorithm. Energy Reports, 6, 297–307.

Liu, Y., Wang, G., Chen, H., Dong, H., Zhu, X., Wang, S. 2011. An improved particle swarm optimization for feature selection. Journal of Bionic Engineering, 8, 191–200.

Lu, H., Chen, J., Yan, K., Jin, Q., Xue, Y., Gao, Z. 2017. A hybrid feature selection algorithm for gene expression data classification. Neurocomputing, 256, 56–62.

Mafarja, M., Aljarah, I., Faris, H., Hammouri, A.I., Al-Zoubi, A.M., Mirjalili, S. 2019. Binary grasshopper optimization algorithm approaches for feature selection problems. Expert Systems with Applications, 117, 267–286.

Mafarja, M., Aljarah, I., Heidari, A.A., Faris, H., Fournier-Viger, P., Li, X., Mirjalili, S. 2018. Binary dragonfly optimization for feature selection using time-varying transfer functions. Knowledge-Based Systems, 161, August, 185–204.

Mafarja, M., Mirjalili, S. 2018. Whale optimization approaches for wrapper feature selection. Applied Soft Computing Journal, 62, 441–453.

Mafarja, M.M., Mirjalili, S. 2017. Hybrid Whale Optimization Algorithm with simulated annealing for feature selection. Neurocomputing, 260, 302–312.

Mandal, A.K., Sen, R., Chakraborty, B. 2019. Binary owl search algorithm for feature subset selection. In 2019 IEEE 10th International Conference on Awareness Science and Technology (iCAST), 1–6. doi:10.1109/ICAwST.2019.8923486.

Manoj, R.J., Anto Praveena, M.D., Vijayakumar, K. 2019. An ACO–ANN based feature selection algorithm for big data. Cluster Computing, 22, 3953–3960.

Mirjalili, S., Lewis, A. 2013. S-shaped versus V-shaped transfer functions for binary Particle Swarm Optimization. Swarm and Evolutionary Computation, 9, 1–14.

Mistry, K., Zhang, L., Neoh, S.C., Lim, C.P., Fielding, B. 2017. A Micro-GA embedded PSO feature selection approach to intelligent facial emotion recognition. IEEE transactions on cybernetics, 47, 1496–1509.

Oliveira, L.S., Sabourin, R., Bortolozzi, F., Suen, C.Y. 2003. A methodology for feature selection using multiobjective genetic algorithms for handwritten digit string recognition. International Journal of Pattern Recognition and Artificial Intelligence, 17, 903–929.

Pashaei, E., Aydin, N. 2017. Binary black hole algorithm for feature selection and classification on biological data. Applied Soft Computing Journal, 56, 94–106.

Sayed, G.I., Tharwat, A., Hassanien, A.E. 2019. Chaotic dragonfly algorithm: an improved metaheuristic algorithm for feature selection. Applied Intelligence, 49, 188–205.

Sindhu, R., Ngadiran, R., Yacob, Y.M., Zahri, N.A.H., Hariharan, M. 2017. Sine–cosine algorithm for feature selection with elitism strategy and new updating mechanism. Neural Computing and Applications, 28, 2947–2958.

Soufan, O., Kleftogiannis, D., Kalnis, P., Bajic, V.B. 2015. DWFS: A wrapper feature selection tool based on a parallel Genetic Algorithm. PLoS ONE, 10, 1–23.

Tian, H., Shen, H. 2005. Hamming distance and hop count based classification for multicast network topology inference. In 19th International Conference on Advanced Information Networking and Applications, 267–272.

Wang, L., Ni, H., Yang, R., Pappu, V., Fenn, M.B., Pardalos, P.M. 2014. Feature selection based on meta-heuristics for biomedicine. Optimization Methods and Software, 29, 703–719.

Xue, B., Zhang, M., Browne, W.N. 2013. Particle swarm optimization for feature selection in classification: A multiobjective approach. IEEE Transactions on Cybernetics, 43, 1656–1671.

Xue, B., Zhang, M., Browne, W.N., Yao, X. 2016. A survey on evolutionary computation approaches to feature selection. IEEE Transactions on Evolutionary Computation, 20, 606–626.

Zawbaa, H.M., Emary, E., Grosan, C. 2016. Feature selection via chaotic antlion optimization. PLoS ONE, 11, 1–21.

Zhang, X., Xu, Y., Yu, C., Heidari, A.A., Li, S., Chen, H., Li, C. 2020. Gaussian mutational chaotic fruit fly-built optimization and feature selection. Expert Systems with Applications. doi:10.1016/j.eswa. 2019.112976.

Zheng, L., Diao, R., Shen, Q. 2015. Self-adjusting harmony search-based feature selection. Soft Computing, 19, 1567–1579.