



Simultaneous Feature Selection and Support Vector Machine Optimization Using the Grasshopper Optimization Algorithm

Ibrahim Aljarah¹ · Ala' M. Al-Zoubi¹ · Hossam Faris¹ · Mohammad A. Hassonah¹ · Seyedali Mirjalili² · Heba Saadeh³

Received: 9 June 2017 / Accepted: 29 December 2017
© Springer Science+Business Media, LLC, part of Springer Nature 2018

Abstract

Support vector machine (SVM) is considered to be one of the most powerful learning algorithms and is used for a wide range of real-world applications. The efficiency of SVM algorithm and its performance mainly depends on the kernel type and its parameters. Furthermore, the feature subset selection that is used to train the SVM model is another important factor that has a major influence on its classification accuracy. The feature subset selection is a very important step in machine learning, specially when dealing with high-dimensional data sets. Most of the previous researches handled these important factors separately. In this paper, we propose a hybrid approach based on the Grasshopper optimisation algorithm (GOA), which is a recent algorithm inspired by the biological behavior shown in swarms of grasshoppers. The goal of the proposed approach is to optimize the parameters of the SVM model, and locate the best features subset simultaneously. Eighteen low- and high-dimensional benchmark data sets are used to evaluate the accuracy of the proposed approach. For verification, the proposed approach is compared with seven well-regarded algorithms. Furthermore, the proposed approach is compared with grid search, which is the most popular technique for tuning SVM parameters. The experimental results show that the proposed approach outperforms all of the other techniques in most of the data sets in terms of classification accuracy, while minimizing the number of selected features.

Keywords SVM · Support vector machine · Grasshopper optimization algorithm · GOA · Optimisation · Feature selection · Metaheuristics

✉ Ibrahim Aljarah
i.aljarah@ju.edu.jo

Ala' M. Al-Zoubi
alaah14@gmail.com

Hossam Faris
hossam.faris@ju.edu.jo

Mohammad A. Hassonah
mohammad.a.hassonah@gmail.com

Seyedali Mirjalili
seyedali.mirjalili@griffithuni.edu.au

Heba Saadeh
heba.saadeh@ju.edu.jo

¹ Business Information Technology Department,
King Abdullah II School for Information Technology,
The University of Jordan, Amman, Jordan

² School of Information and Communication Technology,
Griffith University, Nathan, Brisbane, QLD 4111, Australia

³ Computer Science Department, King Abdullah II
School for Information Technology, The University
of Jordan, Amman, Jordan

Introduction

Bio-inspired systems have been intensively investigated for optimizing different cognitive and learning algorithms [1, 16, 26, 49]. One of these algorithms is the support vector machine (SVM), which is a well-known supervised classification algorithm. SVM was first proposed and implemented by Vapnik [45, 46]. The SVM algorithm is one of the most popular supervised models and is considered to be one of the strongest method in the machine learning field. SVM has some powerful characteristics compared to other methods such as excellent generalization performance, which is able to generate high-quality decision boundaries based on a small subset of training data points. In addition, the SVM has a high ability to model complex and non-linear relations [42, 51].

The basic idea behind the SVM algorithm is to find the optimal hyperplane that separates two classes by maximizing distance between the margin of the hyperplane and the data points in the given data set.

Due to these capabilities, the SVM has been applied on various applications and shown efficient performance in a wide range of challenging real-world problems, such as bio-medical applications [6, 41], pattern recognition [58], anomaly detection [32, 39], and many others [5, 34, 35, 38, 52, 57, 59].

Although the SVM algorithm has many advantages, it also suffered from some limitations such as sensitivity to the initial values of its parameters. These parameters include the cost (C) and the kernel parameters such as the gamma (γ) parameter in the radial basis function (RBF) kernel. Improper selection of these parameters can negatively impact the generalization performance of the SVM. In addition to this limitation, SVM is like many other machine learning algorithms, its performance depends on the selected data set features, which is critically important in improving the generalization performance, increasing the computational efficiency and reducing running time, and produce very accurate classification model [31, 47].

Grasshopper optimization algorithm (GOA) is a recent metaheuristic optimiser, which was proposed by Saremi et al. in [30, 37]. GOA mimics the behavior of grasshopper swarms in nature and simulates the repulsion and attraction forces between the grasshoppers. The authors in [37] showed that this algorithm can outperform other well-regarded and modern metaheuristic algorithms based on a suite of complex benchmark functions and engineering problems.

In this work, we propose a new GOA-SVM model, which applies GOA, for the first time in combination with SVM. In this mode, GOA is utilized for performing feature selection and optimizing the parameters of SVM simultaneously. The goal of the model is to maximize the classification accuracy of SVM with the smallest possible number of features. The proposed approach is evaluated based on 18 public data sets of low and high dimensions. For verification, the proposed approach is compared with seven well regarded and modern algorithms such as genetic algorithm (GA), particle swarm optimization (PSO), grey wolf optimizer (GWO), firefly algorithm (FF), bat algorithm (BA), cuckoo search (CS), and multi-verse optimizer (MVO). Furthermore, the proposed approach is compared with grid search, which is a classical technique used for tuning SVM parameters.

This paper is organized as follows: In Section “Introduction”, we review the previous related works. Section “The Grasshopper Optimization Algorithm” briefly describes the GOA algorithm. Section “Support Vector Machines” presents the SVM classifier. Section “GOA for Optimizing SVM and Feature Selection” discusses in detail the proposed approach for feature selection and SVM optimization based on the GOA algorithm. The experiments and results are implemented and analysed in Section “Experi-

ments and Results”. Finally, we summarize the conclusions and future directions of this work in Section “Conclusions”.

Related Works

In the literature, there have been different trends followed by researchers for tuning the parameters of SVM algorithm. Some researchers simply try different sets of values to tune these parameters by trial and error to reach some acceptable performance. Another traditional approach is to use a simple grid search, but this method is very slow and does not provide satisfactory results due to the large number of parameter combinations [40, 60]. A third approach is based on utilizing evolutionary or swarm intelligence algorithms. These algorithms belong to the metaheuristic search algorithms family [1, 2]. They are inspired by some theories and phenomena in nature like biological evolution or the movements of swarms of creatures in nature. Examples of such algorithms which were applied for optimizing SVM parameters are GA [36], PSO [7, 9], BA [44], FF [43], CS [33], and GWO [14].

Realizing the importance of the feature selection process on the classification accuracy of the SVM models, a distinguished line of research extended the application of evolutionary and swarm intelligence algorithms to perform feature selection concurrently with optimizing the parameters of the SVM. One of the earliest works that followed this approach was conducted by Huang and Wang in [19]. The authors proposed a GA-based approach for optimizing the kernel parameters of the SVM and feature selection with the goal of preserving the classification accuracy of the developed SVM model. Their experimental results showed that GA can outperform the grid search method with even fewer features. Another notable work was conducted by Lin et al. in [25]. They used PSO for the same task and showed that it can outperform grid search; however, it had very similar performance to GA. Other more recent examples of this approach can be found in [11, 15, 18, 23, 27]. Although this line of research is very promising, the amount of work that address optimizing the SVM parameters at the same time as performing feature selection are still relatively scarce.

The No Free Lunch theorem in optimization [48] states that there is and will never be an optimization algorithm to solve all optimization problems. Therefore, new algorithms have the potential to outperform the existing ones on a set of problems. The GOA algorithm is a newcomer and potentially able to show better results with or without modification as compared to the existing algorithms. Saremi et al. [37] showed that this algorithm outperformed most of the existing algorithms on challenging problems with a

large number of local solutions (multi-modal). This justifies the use of this algorithm in this work since we intend to take advantage of its new operators to solve classification problems.

The Grasshopper Optimization Algorithm

The GOA algorithm was proposed in 2016 by Saremi et al. [37]. This algorithm mimics the swarming behavior of grasshoppers in nature. The flying path of a grasshopper in a swarm is affected by three components: social interaction (S_i), gravity (G_i), and wind advection (A_i). In the GOA algorithm, the social interaction is the main mechanism of search defined as follows:

$$S_i = \sum_{j=1, j \neq i}^N s(d_{ij}) \hat{d}_{ij} \quad (1)$$

where d_{ij} is the distance between i -th and j -th grasshopper and it is calculated as $d_{ij} = |x_j - x_i|$, s is a function to define the strength of social forces, and $\hat{d}_{ij} = \frac{x_j - x_i}{d_{ij}}$ is a unit vector from i -th grasshopper to the j -th grasshopper.

It may be seen in this equation that the main component of the social interaction is the function s . This function defines the movement direction of a grasshopper in the swarm and defined as follows:

$$s(r) = f e^{-\frac{r}{l}} - e^{-r} \quad (2)$$

where f indicates the intensity of attraction and l is the attractive length scale.

This function causes two types of forces between the grasshoppers: attraction and repulsion. When the distance between two grasshoppers is between $[0, 2.079]$, they repel each other to avoid collision. The attraction force increases when the distance is in $[2.079, 4]$ to maintain the cohesion of the swarm. When the distance is exactly 2.079, there is no force, and this area is called comfort zone. A conceptual model of attraction, repulsion, and comfort zone is visualized in Fig. 1.

If the distance becomes equal to 2.079, there is no attraction and repulsion. The attraction force increases from 2.079 unit of distance to nearly 4 and then it gradually decreases. Changing the parameters of the equation for the function s (l and f) changes the swarming behavior significantly. To show the interaction between grasshoppers with respect to comfort area, Fig. 1 shows a conceptual schematic.

The swarm model is efficient in simulating the interactions between grasshoppers. However, it must be adjusted to design an optimization algorithm. Saremi et al. [37] proposed the following mathematical model search while

grasshoppers are interacting. The mathematical model is represented by the following equation:

$$X_i^d = c \left(\sum_{j=1, j \neq i}^N c \frac{ub_d - lb_d}{s} s(|x_j^d - x_i^d|) \frac{x_j^d - x_i^d}{d_{ij}} \right) + \hat{T}_d \quad (3)$$

where ub_d is the upper bound in the d -th dimension, lb_d is the lower bound in the d -th dimension, \hat{T}_d is the value of d -th dimension in the target (best solution found so far), and c is a decreasing coefficient to shrink the comfort area, repulsion area, and attraction area.

It may be seen in this equation that the swarm updates the position around a target (\hat{T}_d). The parameter c converges the swarm towards the target. In the GOA algorithm, it is assumed that the target is the best solution obtained so far. While grasshoppers are interacting and chasing the target, the best solution is updated if a better solution is encountered.

The parameter c is the main controlling parameter in the GOA algorithm and, is updated using the following equation:

$$c = c_{\max} - l \frac{c_{\max} - c_{\min}}{L} \quad (4)$$

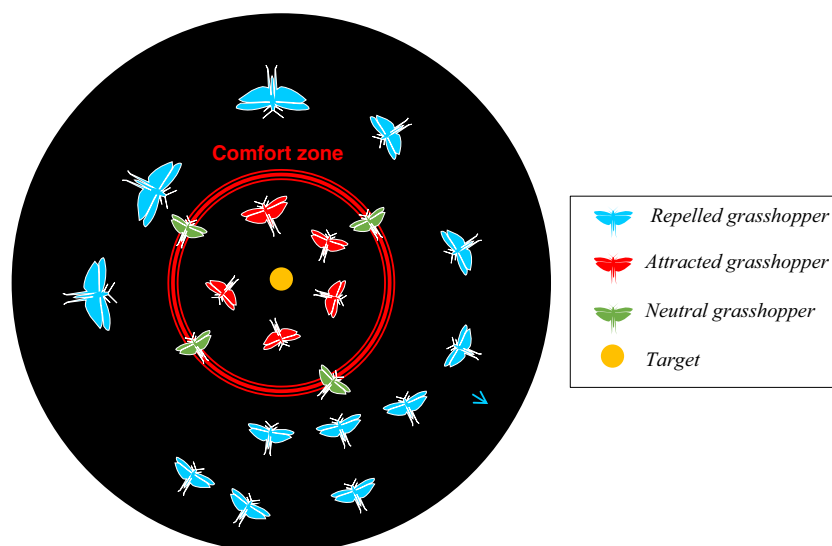
where L shows the maximum number of iteration, l is the current iteration, $c_{\max} = 1$, and $c_{\min} = 0.00001$.

Figure 2 illustrates the effect of the previous equation. This shows the position of the target in 3D space over 100 iterations of the algorithm. The figure illustrates how the mathematical model is employed to implement the movement of a swarm around a target in a 3D space. Due to the use of vectors in the equations, this can be extended to any number of dimensions.

It has been proven that the GOA is very efficient in finding optimal solutions for challenging problems. Despite the simplicity, the algorithm properly gravitates solutions towards to best regions of the search space. The initial population is very important in GOA since the number of random components is very limited in this algorithm.

The computational complexity of the GOA algorithm (excluding SVM in the proposed algorithm) is of $O(t * d * n * n)$ where n is the number of solutions, t is the number of iterations, and d is the number of variables. This is because, the distance of a grasshopper to all the others should be calculated in each dimension over the course of iterations. Note that this computational complexity does not consider the cost of objective function since it varies for every problem.

Fig. 1 Attraction, repulsion, and comfort zone in GOA



Support Vector Machines

SVM is a non-linear binary classifier that was developed by Vladimir Vapnik [13, 45, 46]. They are an extension of the support vector classifier (SVC) to accommodate the non-linear boundaries between classes, which in turn is an extension of the maximal margin classifier (MMC) that is limited to simple linearly separable data [10]. SVMs are used in supervised learning and work for both classification and regression.

Formally, an SVM constructs linear separating hyperplanes in high-dimensional vector spaces in which each data point can be viewed as (x_i, y_i) pair, where x_i is the feature vector $(x_{i1}, x_{i2}, \dots, x_{ip})$; p is the number of features, $i = 1, \dots, n$, where n is the number of training observations, and y_i is the class label; usually $\{-1, 1\}$ [20]. If the feature space is well separated, several hyperplanes will exist that separate the data points belonging to the two labelled classes into separated regions in the space. The best hyperplane is

defined as the one that maximizes the distance to the nearest training data points; this distance is called the margin. The observations that are far from the hyperplane by a distance equal to the margin are called support vectors (SV). These vectors are located in the feature space. The hyperplane location depends directly and only on these observations SV, such that if any of these observations change their locations, the location of the hyperplane will also be changed accordingly [20]. Figure 3 shows the illustration of the idea of the optimal hyperplane and support vectors in SVM.

The linear separating hyperplanes will lead to optimal classification. However, in most of the cases the data points that belong to different classes are not clearly separated, and hence, linear classification will lead to substantial misclassification. One way that SVMs can handle such frequent cases is the ability to map the original feature space (most likely non-linear) to a higher-dimensional space ϕ , in which the data points that belong to different

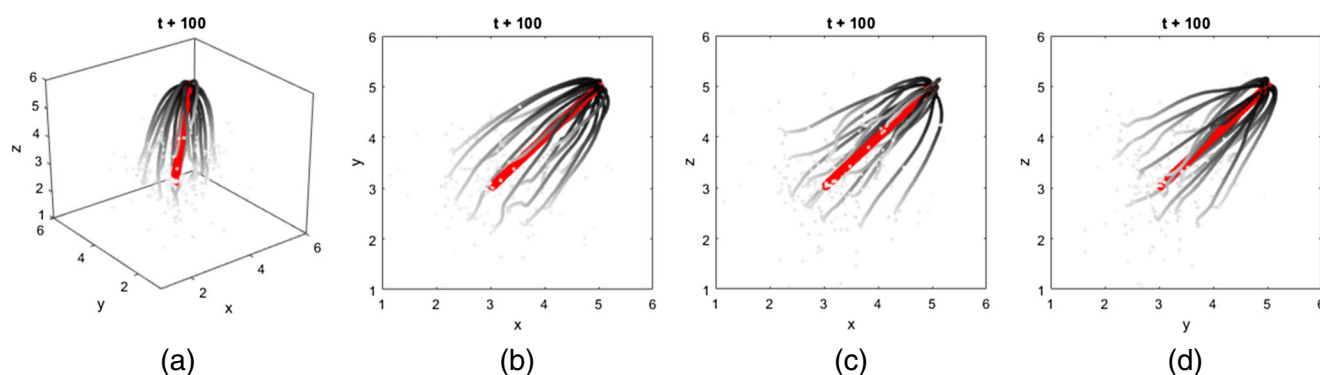


Fig. 2 Trajectory of swarm **a** in 3D space **b** x-y view **c** x-z view **d** y-z view

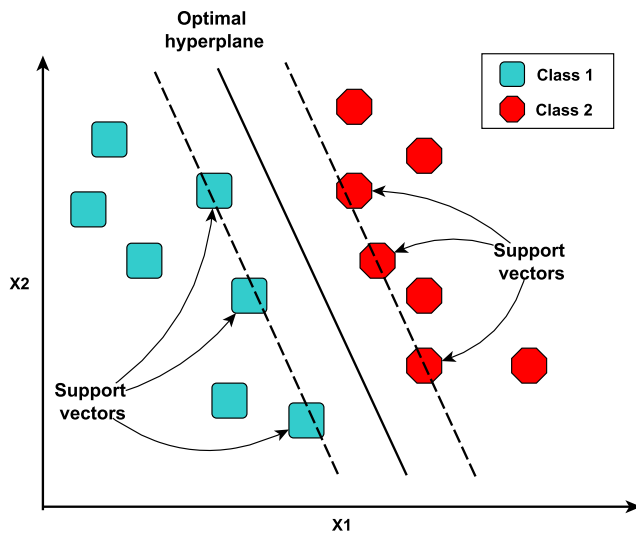


Fig. 3 Illustration of the idea of the optimal hyperplane and support vectors in SVM

classes have clearer separation boundaries (become linearly separable). The high-dimensional space results from non-linear enlarging of the original space using kernel functions. Kernel functions have several forms such as:

- Linear kernels with the following form:

$$k(x_i, x'_i) = \sum_{j=1}^p x_{ij}x'_{ij},$$

- Polynomial kernels with the following form:

$$k(x_i, x'_i) = \left(1 + \sum_{j=1}^p x_{ij}x'_{ij}\right)^d,$$

- RBF kernels with the following form:

$$k(x_i, x'_i) = \exp\left(-\gamma \sum_{j=1}^p (x_{ij} - x'_{ij})^2\right),$$

Where $k(\cdot)$ is the kernel function and its value is the inner product of the two observation vectors x_i and x'_i . In the transformed feature space φ , the inner product of two vectors can be represented as $\varphi(x_i) \cdot \varphi(x'_i)$. A SVM is a support vector classifier with non-linear kernel function.

A major determinant of the performance of the SVM is the selection of its kernel function and their parameters. In this paper, we used the GOA algorithm for optimizing these parameters.

GOA for Optimizing SVM and Feature Selection

In this section, we describe the methodology followed to use GOA for feature selection and optimizing the parameters of SVM simultaneously. The methodology adopted is based on the wrapper-based approach for feature selection. Wrapper

is a type of feature selection method, which evaluates the selected subset of features based on their predictive measure. The main advantage of this approach is that it incorporates a learning algorithm to evaluate the searched subset of features. Wrapper-based feature selection consists of three main components: a search method, an inductive algorithm, and an evaluation measurement [22]. In our implementation, GOA is used as a search algorithm for finding a good subset of features, SVM is used as an inductive algorithm, while classification accuracy is utilized as an evaluation measure. Figure 4 shows the high level of the wrapper-based approach that is used for feature selection based on GOA and SVM.

In order to apply any metaheuristic algorithm for optimizing a problem, there are two formulation issues that should be addressed. The first is how the individual (i.e., solution) is represented, while the second is the selection of a fitness function used for evaluating the quality of the generated solutions. In our proposed GOA-SVM, we address both points as follows.

- Individual/population representation: To address this point, we must decide which decision variables to represent in the individual. In our case, these variables are the SVM parameters and the input features. Therefore, in our implementation, a single solution is represented as a one-dimensional vector of real numbers as shown in Eq. 3. The first two elements of the vector are the candidate hyperparameters of the SVM, which are the cost C and the kernel parameter γ . Other elements of the vector represent a sequence of flags, where each flag corresponds to a feature. That is, if the value of the flag is larger or equals to 0.5 then the feature is selected, otherwise the feature is excluded.

$$I_i = [C \ \gamma \ F_1 \ F_2 \ \dots \ F_n] \quad (5)$$

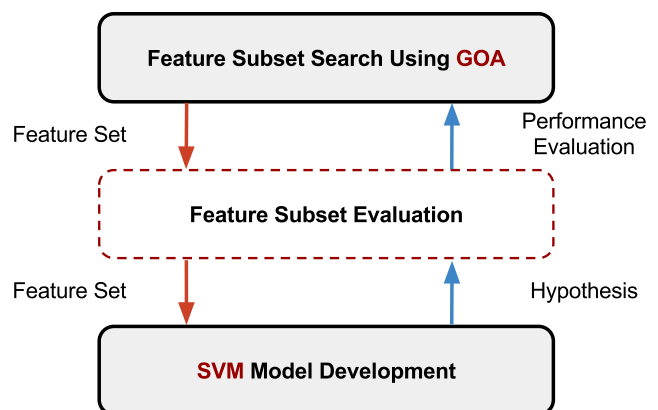


Fig. 4 Wrapper-based approach for feature selection based on GOA and SVM

- **Fitness evaluation:** The quality of every generated individual is assessed using a predefined measurement. For this purpose, a fitness function should be defined. In the proposed model, the fitness function relies on the average classification accuracy of the SVM models trained using the parameters and features indicated by the individual. This fitness function can be formulated as shown in Eq. 6 and will be used to evaluate the quality of individual I_i^t at iteration t , where $c(x_j)$ is the classification result of the j th instance in the testing data set, y_j is the actual class label of the j th instance, δ is the relation between $c(x_j)$, and y_j that is if $c(x_j) = y_j$ then $\delta = 1$ otherwise $\delta = 0$. N is the number of instances in the testing data set. Note that the fitness evaluation is measured based on K -folds cross-validation where K denotes the number of folds. The latter step is important to avoid the problem of overfitting and consequently to obtain more robust results. This cross-validation will be referred to as an *internal cross-validation*.

$$\text{Fitness}(I_i^t) = \frac{1}{K} \sum_{k=1}^K \frac{1}{N} \sum_{j=1}^N \delta(c(x_j), y_j) \quad (6)$$

After deciding on these two design issues, the processes of running the GOA-SVM model can be described as follows:

1. **Initialization:** The GOA-SVM model starts with an initial random population of candidate solutions where each solution consists of the C and γ parameters and a number of flags corresponding to the features in the data set as was shown in Eq. 3.
2. **Update and reproduction:** Reproduction operators of the GOA are applied in this step on all individuals to form a new population of candidate solutions.
3. **Fitness evaluation:** The quality of every generated individual is assessed using the fitness function, which was described previously in Eq. 6.
4. **Termination:** The search process for the best solution ends when the maximum number of iterations is reached. Then, the GOA-SVM model reports the best SVM parameters along with the subset of features that was found to have the highest classification accuracy.
5. **Testing:** For verification, the final step is to test the developed SVM model with the selected parameters and features based on an unseen and new part of the original data set.

The overall process of the proposed GOA-SVM model can be summarized as depicted in the schematic diagram

shown in Fig. 5. The diagram shows that the proposed approach starts by specifying the input data set. After that, some of the preprocessing operations are applied such as data normalization and missing value handling. After this step, the training and testing parts are performed based on a cross-validation methodology. The training part is then used in the SVM parameters optimization and the feature subset selection. After that, the optimization process is started using the GOA algorithm to find the best SVM parameters and the most relevant feature subset of the SVM model. After this iterative process, the final SVM model is used to evaluate the testing data using the classification accuracy evaluation measure.

Experiments and Results

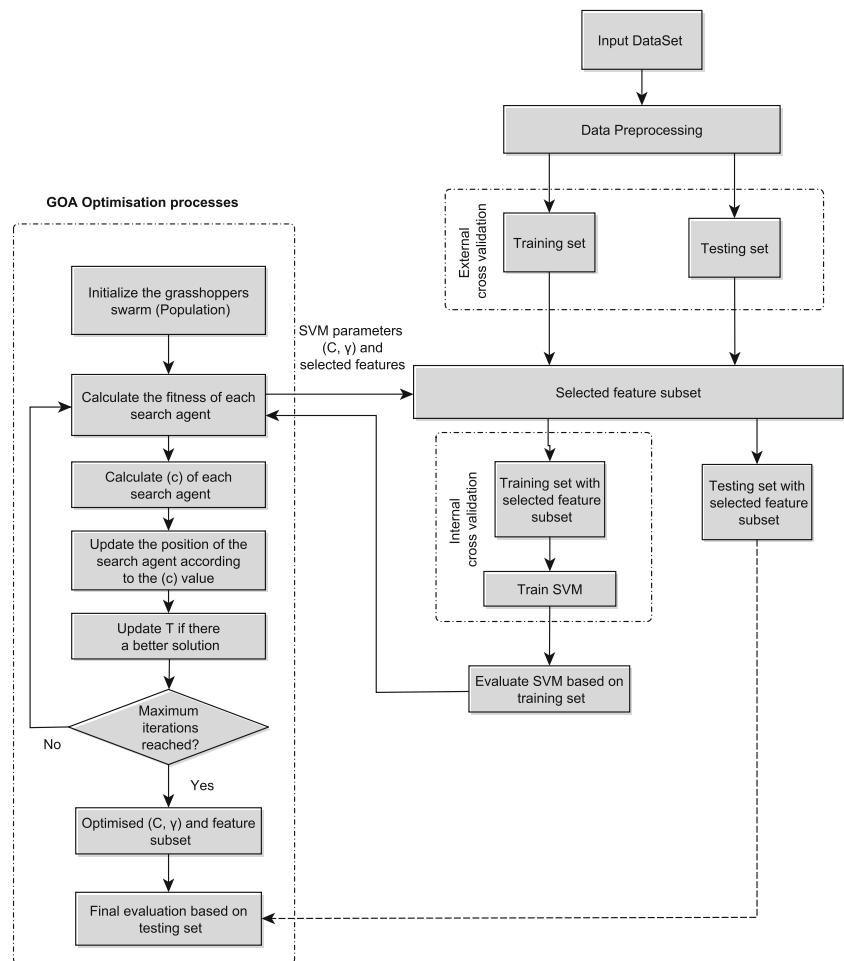
To evaluate and benchmark the proposed GOA-SVM approach, 18 binary and multi-class data sets were obtained from the University of California at Irvine (UCI) machine learning repository¹ [24]. The description of the data sets in terms of number of instances, number of features, and number of classes is given in Table 1. As can be seen, the data sets are varied in their size, and number of dimensions, in order to test the proposed approach on different scales of problem complexity. Moreover, most of the data sets (14 out of 18) are considered high-dimensional data sets, as they have more than 10 features/dimensions [8, 21]. It is worth mentioning that many of these data sets have been used in the literature for testing SVM based classifiers.

To give an equal weight for all features, it is important to convert them all to the same scale. Therefore, all data sets are normalized before the training process. In this work, we apply the min-max normalization, which can be defined as given in Eq. 7, where F_{new} is the newly transformed feature, \min_F and \max_F are the minimum and maximum values of the targeted feature F . Using this equation, all features are scaled to the interval $[0,1]$.

$$F_{\text{new}} = \frac{F_i - \min_F}{\max_F - \min_F} \quad (7)$$

As a training/testing methodology, we apply a 10-folds cross-validation, where each data set is divided into ten equal parts. Nine parts are used for training and one for testing. Then, another nine parts are used for training and the tenth part is used for testing and so on. This processes is repeated ten times. Finally, the average of the classification accuracy on testing parts is reported.

¹<http://archive.ics.uci.edu/ml/>.

Fig. 5 Schematic diagram of the proposed GOA-SVM model**Table 1** List of data sets

Data set	Features	Instances	Classes
Breast cancer	10	683	2
Wine	13	178	3
Heart	13	270	2
Parkinsons	22	195	2
Sonar	60	208	2
Vowel	10	528	11
Spectft	44	276	2
Ionosphere	34	351	2
German	24	1000	2
Vehicle	18	846	4
Australian	15	653	2
Diabetes	8	768	2
Liver	6	345	2
Glass	9	214	6
Arrhythmia	297	452	16
Cleveland	13	303	4
Libras	90	360	15
Teaching Assistant Evaluation (TAE)	5	151	3

The GOA-SVM approach is compared to MVO, GA, PSO, GWO, FF, BA, and CS approaches using the 18 data sets in order to verify its performance. Furthermore, the comparison with the grid search method is discussed. To make the comparisons fair, the experiments are repeated for ten times/runs to minimize random effects. Each run is set to 100 iterations as stopping criteria. Furthermore, all settings and parameters that are used in the experiments for all algorithms are presented in Table 2.

The implementation of the proposed model is developed on Matlab R2015a environment. For SVM, we used Libsvm [12]. The experimental evaluation was performed on a PC with an Intel Core i5-6400, 2.70 GHz, 8 GB RAM, and Windows 7.

Classification rate or accuracy is the main evaluation measure that is used to assess the performance of the GOA-SVM as well as the other algorithms. Accuracy

measures the ratio of the correctly classified instances to the actual classes. For each experiment, average (Avg) and standard deviation (Std) of the classification results are reported. In addition, the Wilcoxon's test is used to check the statistical significance of the difference of the given classification results. In this paper, a 5% significance level is used in the Wilcoxon's test to threshold the calculated P values.

In this paper, we performed three different experiments, the first experiment is applied on the whole set of features for every data set (without feature selection), the second experiment is applied on a reduced set of features for every data set (with feature selection), and in the last experiment, the GOA-SVM is compared with six other algorithms (grid search, MVO, GA, PSO, and GSA), for which the results were obtained directly from previous studies.

Table 2 Initial parameters of the metaheuristic algorithms

Algorithm	Parameter	Value
GOA [37]	cMin	0.00001
	cMax	1
	Number of search agents	50
	Number of iterations	100
GA [17]	Crossover ratio	0.9
	Mutation ratio	0.1
	Selection mechanism	Roulette wheel
	Population size	50
PSO [50, 54]	Generations	100
	Acceleration constants	[2.1, 2.1]
	Inertia w	[0.9, 0.6]
	Number of particles	50
MVO [3, 28]	Generations	100
	Min wormhole existence ratio	0.2
	Max wormhole existence ratio	1
	Universes	50
BA [56]	Iterations	100
	Loudness	0.5
	Pulse rate	0.5
	Frequency minimum	0
FF [55]	Frequency maximum	2
	Alpha	0.5
	Beta	0.2
	Gamma	1
Cuckoo Search [53]	Discovery rate $P\alpha$	0.25
	Number of nests	50
	Number of generations	100
GWO [4, 29]	α	Min = 0 and max = 2

Experiment I: Results Without Applying Feature Selection

In this experiment, we apply the GOA and other algorithms on 18 data sets and employing the complete set of features

for each data set. The accuracy results of the GOA and other algorithms are presented in Table 3. The average accuracy values show that GOA outperforms the other metaheuristics and obtains the best performance on 11 data sets out of 18 data sets (60% of the data sets). For the other 7 data sets, the

Table 3 Classification accuracy and standard deviation results for all data sets without applying feature selection

Algorithm	GOA	MVO	GA	PSO
Data set	Avg \pm std	Avg \pm Std	Avg \pm Std	Avg \pm Std
Breast cancer	97.23 \pm 1.88	96.92 \pm 2.13	96.78 \pm 1.94	96.63 \pm 1.15
Wine	97.77 \pm 2.72	97.75 \pm 2.76	97.71 \pm 2.80	96.60 \pm 2.78
Heart	82.22 \pm 6.24	82.59 \pm 5.51	82.59 \pm 7.23	83.70 \pm 2.46
Parkinsons	94.95 \pm 5.00	94.89 \pm 5.05	94.92 \pm 4.47	93.37 \pm 4.52
Sonar	88.55 \pm 8.01	88.50 \pm 6.14	87.52 \pm 8.57	87.98 \pm 5.34
Vowel	99.81 \pm 0.57	99.43 \pm 0.88	99.43 \pm 0.88	98.86 \pm 0.93
Spectft	78.73 \pm 7.82	76.45 \pm 4.81	76.79 \pm 6.45	75.31 \pm 5.91
Ionosphere	95.16 \pm 3.14	94.59 \pm 3.93	93.18 \pm 3.09	94.04 \pm 4.05
German	75.80 \pm 5.31	75.70 \pm 4.45	76.80 \pm 5.27	76.30 \pm 4.94
Vehicle	85.35 \pm 3.48	84.76 \pm 3.93	82.96 \pm 5.02	85.10 \pm 3.63
Australian	84.93 \pm 4.06	84.35 \pm 3.54	85.07 \pm 4.72	84.78 \pm 2.27
Diabetes	76.05 \pm 3.72	76.17 \pm 3.81	76.44 \pm 6.43	76.83 \pm 6.48
Liver	71.60 \pm 6.17	68.94 \pm 4.02	72.13 \pm 6.28	73.32 \pm 5.24
Glass	70.06 \pm 11.84	65.97 \pm 11.10	65.48 \pm 8.34	67.77 \pm 8.37
Arrhythmia	65.24 \pm 10.91	64.24 \pm 13.26	68.78 \pm 5.95	71.21 \pm 6.67
Cleveland	55.08 \pm 12.07	58.01 \pm 8.93	59.01 \pm 8.61	57.72 \pm 7.81
Libras	87.78 \pm 5.30	87.22 \pm 5.00	87.22 \pm 3.97	87.50 \pm 3.34
TAE	58.25 \pm 10.42	56.29 \pm 13.03	54.92 \pm 16.37	55.54 \pm 10.59
Algorithm	GWO	FF	BA	CS
Data set	Avg \pm Std	Avg \pm Std	Avg \pm Std	Avg \pm Std
Breast Cancer	96.64 \pm 1.30	96.64 \pm 1.30	96.93 \pm 1.37	96.49 \pm 1.47
Wine	96.63 \pm 2.75	96.08 \pm 4.35	97.22 \pm 5.12	97.75 \pm 3.72
Heart	82.96 \pm 9.54	84.44 \pm 4.91	82.96 \pm 6.67	83.33 \pm 4.46
Parkinsons	94.89 \pm 5.62	93.89 \pm 4.93	92.79 \pm 6.63	93.82 \pm 5.55
Sonar	85.50 \pm 5.93	88.45 \pm 3.85	88.02 \pm 7.48	85.10 \pm 6.88
Vowel	99.05 \pm 1.28	99.24 \pm 0.93	99.43 \pm 0.86	99.24 \pm 0.93
Spectft	77.54 \pm 8.12	77.19 \pm 10.75	77.54 \pm 5.46	77.25 \pm 8.71
Ionosphere	94.30 \pm 3.13	94.57 \pm 5.34	94.86 \pm 4.75	94.29 \pm 3.14
German	76.10 \pm 3.78	76.90 \pm 4.20	76.30 \pm 3.49	75.80 \pm 4.19
Vehicle	83.93 \pm 3.02	84.87 \pm 5.45	84.75 \pm 2.19	84.64 \pm 3.39
Australian	86.38 \pm 3.50	85.36 \pm 5.36	85.07 \pm 4.25	84.06 \pm 4.85
Diabetes	75.91 \pm 4.55	75.77 \pm 4.83	76.31 \pm 6.50	76.70 \pm 3.54
Liver	71.27 \pm 6.04	71.91 \pm 7.37	71.00 \pm 6.55	72.48 \pm 7.79
Glass	64.48 \pm 8.23	67.88 \pm 9.97	67.27 \pm 8.67	69.09 \pm 10.83
Arrhythmia	71.92 \pm 7.23	59.06 \pm 8.88	71.24 \pm 6.37	71.72 \pm 7.97
Cleveland	56.44 \pm 6.17	57.09 \pm 5.19	59.78 \pm 11.28	57.73 \pm 7.88
Libras	84.72 \pm 4.17	86.39 \pm 3.61	87.22 \pm 7.26	86.67 \pm 6.89
TAE	57.71 \pm 14.57	52.96 \pm 15.08	55.50 \pm 11.45	58.13 \pm 9.75

(The best values are marked in italic)

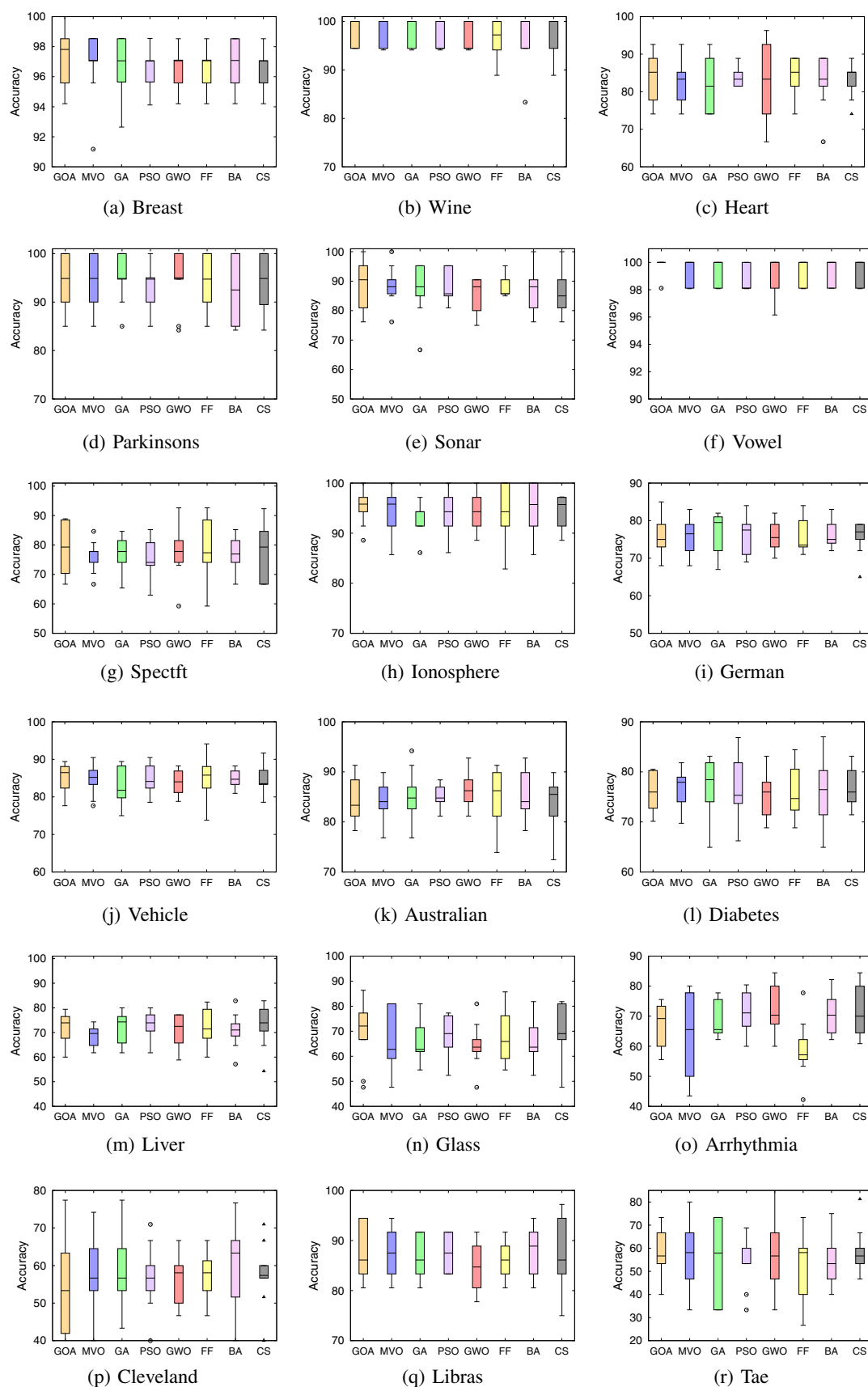


Fig. 6 Boxplot charts for GOA and other algorithms based on 18 data sets (without features selection)

Table 4 *P* values of the Wilcoxon test of GOA vs. other algorithms before applying features selection (*P* ≥ 0.05 are underlined)

Data set/algorithm	MVO	GA	PSO	GWO	FF	BA	CS
Breast cancer	<u>9.02E-01</u>	<u>1.67E-01</u>	7.72E-04	2.74E-03	2.74E-03	7.85E-02	2.75E-04
Wine	<u>5.71E-01</u>	2.64E-02	1.45E-04	5.72E-09	5.40E-05	<u>7.74E-01</u>	<u>7.74E-01</u>
Heart	<u>6.17E-01</u>	<u>8.05E-01</u>	<u>1.00E+00</u>	<u>7.12E-01</u>	<u>1.01E-01</u>	<u>5.33E-01</u>	<u>1.00E+00</u>
Parkinsons	<u>8.00E-01</u>	<u>9.00E-01</u>	5.78E-03	<u>4.46E-01</u>	<u>7.78E-02</u>	2.26E-02	<u>1.67E-01</u>
Sonar	<u>4.57E-01</u>	<u>6.21E-01</u>	<u>3.86E-01</u>	1.69E-04	<u>2.62E-01</u>	<u>3.86E-01</u>	5.39E-04
Vowel	1.16E-04	2.37E-04	9.00E-14	1.40E-06	6.44E-06	2.86E-03	6.44E-06
Spectft	1.89E-02	<u>1.10E-01</u>	3.11E-03	<u>5.39E-01</u>	<u>6.24E-01</u>	<u>2.19E-01</u>	<u>1.40E-01</u>
Ionosphere	<u>6.13E-01</u>	8.67E-07	3.21E-02	1.74E-02	<u>7.09E-01</u>	<u>9.01E-01</u>	<u>2.04E-01</u>
German	<u>6.25E-01</u>	<u>6.61E-02</u>	<u>3.27E-01</u>	<u>6.23E-01</u>	<u>3.26E-01</u>	<u>4.60E-01</u>	<u>6.41E-02</u>
Vehicle	<u>2.20E-01</u>	3.27E-03	<u>6.24E-01</u>	1.45E-03	<u>4.63E-01</u>	3.28E-03	4.98E-02
Australian	<u>8.06E-01</u>	<u>3.89E-01</u>	<u>4.59E-01</u>	1.37E-03	<u>1.09E-01</u>	<u>2.67E-01</u>	<u>9.03E-01</u>
Diabetes	<u>8.07E-01</u>	4.99E-02	<u>2.70E-01</u>	<u>6.25E-01</u>	<u>6.24E-01</u>	<u>9.03E-01</u>	<u>3.27E-01</u>
Liver	2.86E-05	<u>2.19E-01</u>	<u>8.58E-02</u>	<u>6.24E-01</u>	<u>3.91E-01</u>	4.98E-02	<u>3.25E-01</u>
Glass	6.87E-03	1.40E-04	2.70E-02	4.83E-07	2.74E-02	2.10E-03	<u>3.25E-01</u>
Arrhythmia	<u>8.07E-01</u>	<u>1.11E-01</u>	2.34E-04	9.37E-04	2.66E-07	9.37E-04	1.45E-03
Cleveland	<u>6.59E-02</u>	1.41E-02	<u>1.41E-01</u>	<u>6.24E-01</u>	<u>3.26E-01</u>	2.93E-03	<u>1.77E-01</u>
Libras	<u>3.81E-01</u>	<u>5.32E-01</u>	<u>9.01E-01</u>	1.12E-04	<u>1.03E-01</u>	<u>1.00E+00</u>	<u>7.09E-01</u>
TAE	<u>3.89E-01</u>	<u>6.53E-02</u>	<u>2.14E-01</u>	<u>1.41E-01</u>	1.39E-02	8.39E-02	<u>1.76E-01</u>

average accuracy results of the GOA are very competitive. For example, the performance of the GOA on the liver data set was better than three of other algorithms (MVO, GWO, and BA). Moreover, the proposed approach obtains the highest classification accuracies on high-dimensional data sets such as Sonar, Spectft, Libras, and Ionosphere. These results support the capability of the proposed approach to handle high-dimensional data sets. In addition, GOA shows smaller Std values for most of the data sets, which supports the robustness of the algorithm.

Figure 6 presents the boxplot charts of the accuracy for all used data sets. The boxplots are shown for the ten values of classification accuracy that are given by each algorithm at the end of the SVM training. In this plot, the box represents the interquartile range (IQR), the whiskers represent the maximum accuracy values, the bar in the box represents the median accuracy values, and outliers of the accuracy values are expressed as small circles. The boxplots indicate the improved performance of GOA for optimizing the SVM.

To assess the overall performance of GOA and other algorithms, and to confirm the significance and robustness of the results, we apply the Wilcoxon's statistical test with 5% significance level on the obtained average accuracy results. Table 4 lists Wilcoxon's test *P* values between the GOA and other approaches. The *P* values less than 0.05 indicate that the GOA results have statistically significant differences compared with other approaches. The *P* values that are greater than 0.05 (underlined) are not significant compared with other approaches. According to this table, the GOA results are statistically significant in most of the data sets.

As an overall summary for all data sets, Table 5 shows summarized values which represent the number of data sets each algorithm won/tied/lost on accuracy, and the associated *P* values. It appears that the GOA algorithm is superior on 11 data sets out of 18 in terms of accuracy. Moreover, in terms of significance, the results show that the GOA is better on 4, 5, 7, 8, 6, 8, and 4 data sets

Table 5 Summary statistical results on a variety of measures (without feature selection)

Algorithm	GOA			MVO			GA			PSO			GWO			FF			BA			CS		
	W	T	L	W	T	L	W	T	L	W	T	L	W	T	L	W	T	L	W	T	L	W	T	L
Accuracy	11	0	7	0	0	18	0	0	18	2	0	16	2	0	16	2	0	16	1	0	17	0	0	18
<i>P</i> value	N/A	N/A	N/A	0	14	4	2	11	5	1	10	7	2	8	8	0	12	6	1	9	8	1	13	4
Rank	3.08			4.81			4.67			4.69			5.22			4.83			4.06			4.64		

The table values represent the number of data sets each algorithm won/losses/ties on a variety of measures along with their rank using *F* test

compared with MVO, GA, PSO, GWO, FF, BA, and CS, respectively. In addition, the Friedman test was used to rank the different algorithms applied to 18 data sets. The results of the Friedman test show that the GOA obtains the best rank (lower is better). This confirms the ability of the GOA algorithm to optimize the parameters of the SVM classifier.

Experiment II: Results with Applying Feature Selection

In this experiment, we apply the GOA and other algorithms on 18 data sets, when extending the GOA for simultaneously optimizing the SVM parameters and performing feature selection. Table 6 shows the results when applying GOA with feature selection compared with GOA without feature selection. The results show that the effect of feature selection improves the results. As we can see in the table, the accuracy results are improved for ten data sets with clear reduction of the original features (32 to 65% reduction rate). In addition, the results for the remaining eight data sets with feature selection are very close to the results with all features. These results indicate that the proposed

feature selection approach, used simultaneously with SVM parameter optimization, is able to get better accuracy results with a considerable reduction in the number of original features.

For high-dimensional data sets such as Libras, Arrhythmia, Ionosphere, Spectft, and Sonar, their results show that the features are reduced to 52.22, 53.09, 55.59, 55.00, and 47.00%, respectively. In addition, the accuracy rates for these data sets are higher than the results with all features. These results reveal that the proposed approach not only locates the optimal model parameters and reduces the number of features but also can find the optimal and most relevant features.

Table 7 shows the comparison of GOA and other algorithms after applying the feature selection. It is noted that GOA outperforms the other algorithms in 12 data sets in terms of classification accuracy. Moreover, it shows competitive results in terms of minimizing the number of selected features.

Figure 7 shows the boxplots of the accuracy results with feature selection for all of the data sets used. The boxplots confirm that the results of the GOA approach has higher median while maintaining smaller interquartile ranges in

Table 6 Classification accuracy and standard deviation results of GOA for all data sets with and without applying feature selection

Algorithm	Without feature selection		With feature selection		
	Accuracy (Avg \pm Std)	Original No. of features	Accuracy (Avg \pm Std)	No. of selected features (Avg \pm Std)	Reduction rate (%)
Breast cancer	97.23 \pm 1.88	10	96.49 \pm 1.17	6.70 \pm 1.19	33.00
Wine	97.77 \pm 2.72	13	97.77 \pm 2.72	7.70 \pm 1.35	40.77
Heart	82.22 \pm 6.24	13	82.59 \pm 6.20	7.10 \pm 1.22	45.38
Parkinsons	94.95 \pm 5.00	22	95.97 \pm 4.37	12.90 \pm 1.92	41.36
Sonar	88.55 \pm 8.01	60	88.86 \pm 5.93	31.80 \pm 3.54	47.00
Vowel	99.81 \pm 0.57	10	99.05 \pm 1.27	7.40 \pm 0.92	26.00
Spectft	78.73 \pm 7.82	44	80.52 \pm 5.44	19.80 \pm 3.16	55.00
Ionosphere	95.16 \pm 3.14	34	93.44 \pm 2.88	16.80 \pm 2.27	50.59
German	75.80 \pm 5.31	24	73.4 \pm 4.45	10.00 \pm 3.03	58.33
Vehicle	85.35 \pm 3.48	18	82.50 \pm 2.12	11.40 \pm 1.28	36.67
Australian	84.93 \pm 4.06	15	85.07 \pm 5.27	5.40 \pm 2.00	64.00
Diabetes	76.05 \pm 3.72	8	74.35 \pm 5.15	4.10 \pm 1.58	48.75
Liver	71.60 \pm 6.17	6	67.51 \pm 8.05	4.20 \pm 1.47	30.00
Glass	70.06 \pm 11.84	9	69.19 \pm 10.72	5.20 \pm 1.17	42.22
Arrhythmia	65.24 \pm 10.91	297	70.35 \pm 4.75	142.30 \pm 4.69	52.09
Cleveland	55.08 \pm 12.07	13	55.47 \pm 7.06	6.90 \pm 2.12	46.92
Libras	87.78 \pm 5.30	90	89.44 \pm 4.44	43.00 \pm 4.31	52.22
TAE	58.25 \pm 10.42	5	58.91 \pm 7.85	3.40 \pm 1.02	32.00

(The best values are marked in italic)

Table 7 Classification accuracy and standard deviation results of GOA and other approaches for all data sets after applying feature selection

Algorithm	GOA		MVO		GA		PSO	
Data set	Avg \pm Std	NSF	Avg \pm Std	NSF	Avg \pm Std	NSF	Avg \pm Std	NSF
Breast cancer	96.49 \pm 1.17	6.70 \pm 1.19	96.05 \pm 1.73	5.10 \pm 1.45	96.04 \pm 1.86	6.40 \pm 1.56	95.61 \pm 1.83	6.40 \pm 1.43
Wine	97.77 \pm 2.72	7.70 \pm 1.35	97.19 \pm 3.75	8.10 \pm 1.58	97.19 \pm 2.81	7.90 \pm 1.7	95.55 \pm 3.33	7.60 \pm 1.02
Heart	82.59 \pm 6.20	7.10 \pm 1.22	81.85 \pm 5.60	8.60 \pm 1.69	80.74 \pm 6.37	6.50 \pm 2.38	81.85 \pm 7.49	8.40 \pm 1.20
Parkinsons	95.97 \pm 4.37	12.90 \pm 1.92	93.87 \pm 4.98	12.10 \pm 2.84	94.34 \pm 4.26	8.70 \pm 1.79	94.37 \pm 5.86	10.40 \pm 2.11
Sonar	88.86 \pm 5.93	31.80 \pm 3.54	86.12 \pm 8.32	29.90 \pm 2.43	85.14 \pm 12.09	21.90 \pm 4.97	84.16 \pm 5.59	29.70 \pm 2.97
Vowel	99.05 \pm 1.27	7.40 \pm 0.92	97.17 \pm 2.83	8.20 \pm 0.75	98.49 \pm 2.36	8.20 \pm 0.60	98.68 \pm 1.47	8.00 \pm 1.26
Spectft	80.52 \pm 5.44	19.80 \pm 3.16	80.14 \pm 7.56	20.20 \pm 3.54	77.95 \pm 11.25	14.30 \pm 2.00	78.28 \pm 7.67	20.40 \pm 4.86
Ionosphere	93.44 \pm 2.88	16.80 \pm 2.27	90.87 \pm 4.40	15.50 \pm 3.44	92.60 \pm 4.07	14.00 \pm 3.49	92.88 \pm 3.88	18.70 \pm 1.79
German	73.40 \pm 4.45	10.00 \pm 3.03	73.90 \pm 6.42	13.00 \pm 3.16	72.60 \pm 4.82	9.10 \pm 3.56	74.80 \pm 5.65	15.80 \pm 1.17
Vehicle	82.50 \pm 2.12	11.40 \pm 1.28	81.43 \pm 3.27	13.00 \pm 2.10	81.56 \pm 3.32	11.70 \pm 2.19	81.79 \pm 2.75	13.20 \pm 1.25
Australian	85.07 \pm 5.27	5.40 \pm 2.00	86.81 \pm 5.04	5.30 \pm 1.73	84.64 \pm 5.39	5.20 \pm 0.87	84.06 \pm 4.44	4.90 \pm 1.04
Diabetes	74.35 \pm 5.15	4.10 \pm 1.58	76.56 \pm 2.97	4.80 \pm 1.78	76.69 \pm 3.63	6.30 \pm 1.62	76.43 \pm 4.25	5.40 \pm 1.28
Liver	67.51 \pm 8.05	4.20 \pm 1.47	62.87 \pm 8.34	3.40 \pm 1.28	66.83 \pm 10.10	4.40 \pm 1.20	66.18 \pm 1.28	3.70 \pm 1.35
Glass	69.19 \pm 10.72	5.20 \pm 1.17	66.82 \pm 9.93	5.00 \pm 0.89	66.84 \pm 9.87	5.20 \pm 1.40	66.90 \pm 7.31	5.10 \pm 0.94
Arrhythmia	70.35 \pm 4.75	142.30 \pm 4.69	70.80 \pm 7.84	142.90 \pm 9.18	70.83 \pm 6.44	70.10 \pm 23.30	70.14 \pm 5.39	142.20 \pm 9.09
Cleveland	55.47 \pm 7.06	6.90 \pm 2.12	48.17 \pm 10.85	6.30 \pm 1.79	55.82 \pm 6.68	5.30 \pm 2.00	57.03 \pm 11.83	6.20 \pm 1.66
Libras	89.44 \pm 4.44	43.00 \pm 4.31	88.33 \pm 5.09	44.10 \pm 3.88	88.61 \pm 4.20	30.20 \pm 7.32	89.17 \pm 4.56	44.50 \pm 6.07
TAE	58.91 \pm 7.85	3.40 \pm 1.02	53.13 \pm 17.12	3.90 \pm 0.94	51.83 \pm 14.54	4.60 \pm 0.49	54.38 \pm 14.11	4.50 \pm 0.67

Algorithm	GWO		FF		BA		CS	
Data set	Avg \pm Std	NSF	Avg \pm Std	NSF	Avg \pm Std	NSF	Avg \pm Std	NSF
Breast cancer	95.89 \pm 2.86	5.40 \pm 1.62	96.33 \pm 2.57	4.70 \pm 1.27	96.20 \pm 1.74	5.60 \pm 1.50	96.48 \pm 1.64	5.90 \pm 1.45
Wine	97.22 \pm 2.78	7.10 \pm 1.45	96.11 \pm 4.34	7.10 \pm 2.02	94.41 \pm 6.57	7.00 \pm 1.18	96.57 \pm 3.85	8.10 \pm 1.04
Heart	80.74 \pm 7.55	6.90 \pm 1.81	81.11 \pm 3.87	6.00 \pm 2.53	80.37 \pm 7.95	7.00 \pm 2.45	81.11 \pm 3.87	8.40 \pm 1.02
Parkinsons	95.92 \pm 5.85	9.10 \pm 2.47	92.37 \pm 6.83	11.30 \pm 1.10	92.29 \pm 4.67	12.70 \pm 1.68	95.37 \pm 4.30	10.90 \pm 2.21
Sonar	88.00 \pm 5.70	19.40 \pm 1.96	86.55 \pm 8.24	28.60 \pm 4.29	86.09 \pm 5.34	28.10 \pm 4.41	85.64 \pm 7.91	29.7 \pm 3.58
Vowel	98.86 \pm 1.25	8.50 \pm 1.02	99.05 \pm 0.95	8.30 \pm 0.64	99.05 \pm 1.27	8.30 \pm 1.00	98.67 \pm 1.49	8.40 \pm 0.92
Spectft	76.38 \pm 5.68	16.10 \pm 3.27	77.46 \pm 10.48	19.80 \pm 2.32	77.18 \pm 8.94	18.90 \pm 2.70	77.12 \pm 9.46	18.80 \pm 2.89
Ionosphere	92.59 \pm 3.67	13.70 \pm 2.10	90.88 \pm 5.24	14.80 \pm 3.28	90.88 \pm 4.75	16.70 \pm 2.90	92.89 \pm 5.87	16.70 \pm 4.15
German	73.50 \pm 3.91	12.30 \pm 1.55	70.20 \pm 5.36	10.40 \pm 2.20	71.70 \pm 4.50	11.90 \pm 1.51	74.70 \pm 3.77	15.40 \pm 2.69
Vehicle	81.32 \pm 5.68	11.40 \pm 1.11	81.68 \pm 3.36	11.80 \pm 1.17	80.14 \pm 6.55	11.40 \pm 2.00	81.67 \pm 4.31	13.20 \pm 1.66
Australian	85.36 \pm 4.07	5.10 \pm 0.70	85.07 \pm 3.55	3.60 \pm 0.49	84.35 \pm 4.09	4.90 \pm 0.94	84.49 \pm 6.12	7.40 \pm 2.29
Diabetes	76.69 \pm 1.84	5.30 \pm 1.19	74.87 \pm 4.50	3.30 \pm 1.42	75.13 \pm 4.58	4.80 \pm 1.47	76.70 \pm 4.20	6.10 \pm 1.04
Liver	62.01 \pm 7.04	3.40 \pm 1.56	66.43 \pm 7.10	2.90 \pm 1.30	62.92 \pm 6.88	4.00 \pm 1.55	68.11 \pm 7.26	5.40 \pm 0.66
Glass	63.57 \pm 8.10	4.30 \pm 0.64	67.27 \pm 7.82	4.60 \pm 1.11	68.29 \pm 10.96	4.80 \pm 1.33	68.64 \pm 13.13	5.00 \pm 1.18
Arrhythmia	69.71 \pm 4.67	67.00 \pm 14.02	66.41 \pm 10.49	137.70 \pm 6.21	71.68 \pm 4.64	142.30 \pm 6.77	69.92 \pm 6.59	140.70 \pm 5.2
Cleveland	56.10 \pm 8.49	6.20 \pm 2.18	55.42 \pm 10.30	5.20 \pm 1.99	55.80 \pm 9.60	5.60 \pm 1.50	58.09 \pm 7.70	7.10 \pm 1.51
Libras	87.22 \pm 5.30	30.40 \pm 4.76	86.39 \pm 3.15	44.40 \pm 5.06	87.22 \pm 4.51	43.60 \pm 3.98	87.5 \pm 4.167	43.60 \pm 4.15
TAE	52.46 \pm 11.70	3.70 \pm 0.90	52.50 \pm 16.18	4.40 \pm 0.80	53.67 \pm 12.51	3.90 \pm 0.94	56.83 \pm 9.84	4.00 \pm 0.77

(NFS stands for number of selected features)

(The best values are marked in italic)

most cases. This proves that the GOA algorithm is better than others in terms of stability and robustness.

Table 8 shows the obtained P values from the Wilcoxon's rank test between the GOA algorithm and other algo-

rithms for all data sets after applying feature selection. The table shows that the accuracy results of the GOA are significantly improved for most of the data sets.

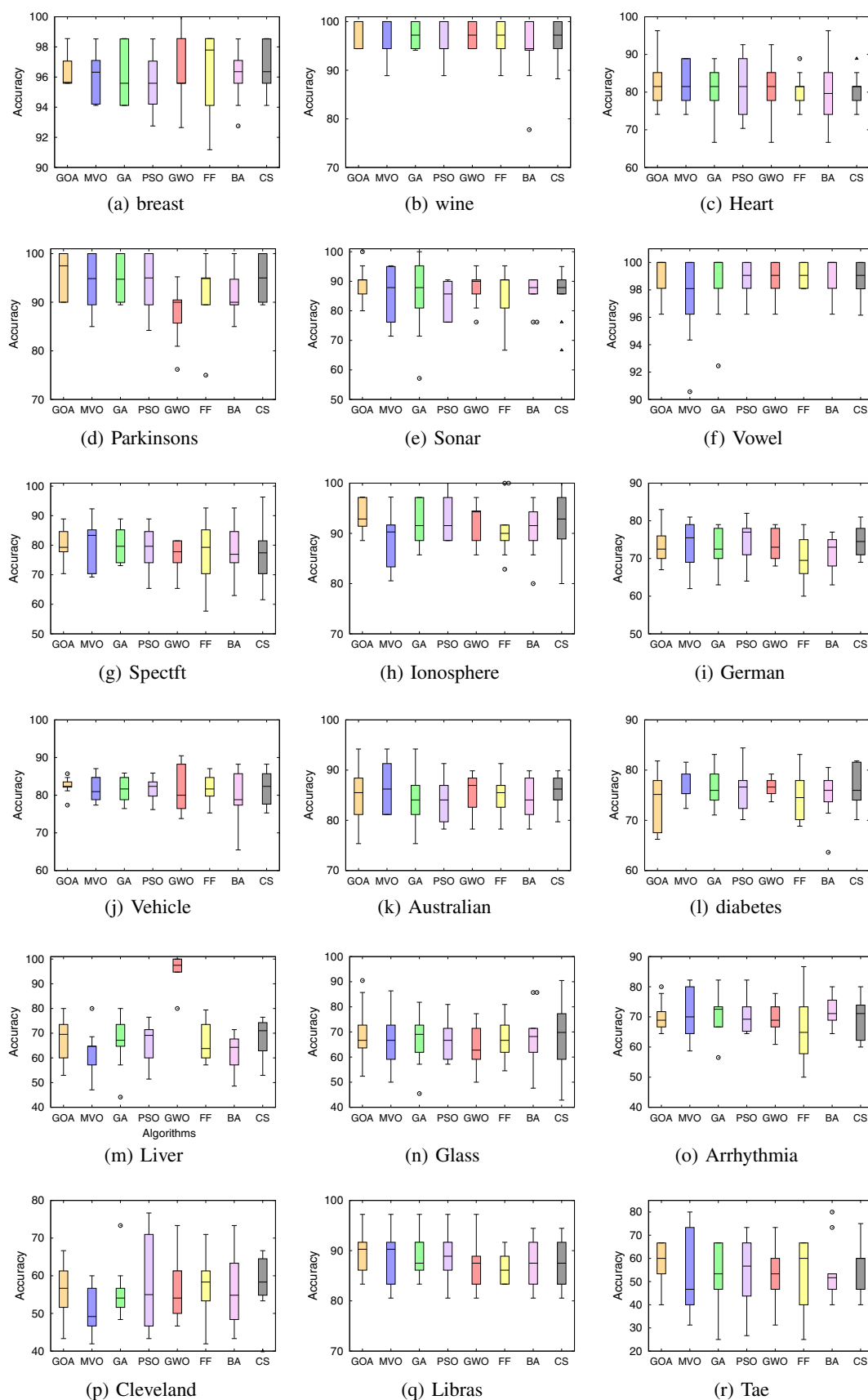


Fig. 7 Boxplot charts for GOA and other algorithms based on 18 data sets after applying features selection

Table 8 *P* values of the Wilcoxon's test of GOA classification results vs other algorithms after applying features selection (*P* ≥ 0.05 are underlined)

Data set/algorithm	MVO	GA	PSO	GWO	FF	BA	CS
Breast cancer	2.56E-02	4.21E-03	5.23E-07	<u>2.06E-01</u>	<u>5.37E-01</u>	<u>6.18E-01</u>	<u>9.02E-01</u>
Wine	<u>2.62E-01</u>	<u>5.14E-02</u>	2.73E-06	<u>1.57E-01</u>	1.34E-02	1.83E-05	1.34E-02
Heart	<u>7.10E-01</u>	<u>4.54E-01</u>	<u>4.59E-01</u>	<u>3.85E-01</u>	<u>3.13E-01</u>	2.55E-02	<u>3.13E-01</u>
Parkinsons	5.97E-04	9.02E-04	9.38E-02	1.27E-14	3.19E-05	2.66E-10	1.22E-01
Sonar	2.57E-02	6.40E-02	8.57E-09	<u>2.13E-01</u>	<u>3.75E-01</u>	1.55E-04	5.58E-03
Vowel	1.23E-07	<u>6.78E-01</u>	<u>1.76E-01</u>	<u>2.80E-01</u>	<u>6.84E-01</u>	<u>1.00E+00</u>	<u>5.96E-02</u>
Spectft	<u>8.06E-01</u>	<u>6.22E-01</u>	<u>1.39E-01</u>	1.24E-04	<u>1.76E-01</u>	1.40E-03	3.20E-03
Ionosphere	1.45E-06	<u>1.31E-01</u>	<u>1.69E-01</u>	<u>2.12E-01</u>	1.30E-05	1.76E-03	<u>9.02E-01</u>
German	<u>5.00E-02</u>	<u>9.03E-01</u>	4.78E-03	<u>3.26E-01</u>	1.76E-05	<u>2.19E-01</u>	2.73E-02
Vehicle	2.69E-02	<u>2.20E-01</u>	<u>2.18E-01</u>	1.42E-02	3.21E-03	3.71E-02	<u>7.14E-01</u>
Australian	<u>6.50E-02</u>	<u>1.10E-01</u>	<u>1.40E-01</u>	<u>2.66E-01</u>	<u>7.12E-01</u>	<u>4.60E-01</u>	<u>7.11E-01</u>
Diabetes	4.50E-03	9.93E-03	1.35E-02	9.74E-03	<u>3.90E-01</u>	<u>3.91E-01</u>	9.99E-03
Liver	9.08E-04	<u>8.07E-01</u>	<u>2.20E-01</u>	2.06E-34	<u>3.25E-01</u>	3.04E-05	<u>3.26E-01</u>
Glass	<u>3.28E-01</u>	<u>6.24E-01</u>	<u>4.63E-01</u>	3.13E-05	<u>8.07E-01</u>	<u>1.00E+00</u>	<u>1.00E+00</u>
Arrhythmia	<u>1.00E+00</u>	3.63E-02	<u>5.40E-01</u>	<u>2.21E-01</u>	2.35E-04	1.36E-02	<u>6.24E-01</u>
Cleveland	3.04E-06	<u>2.67E-01</u>	<u>8.07E-01</u>	<u>7.13E-01</u>	<u>5.39E-01</u>	<u>9.03E-01</u>	9.76E-03
Libras	<u>7.77E-02</u>	<u>1.68E-01</u>	<u>8.04E-01</u>	8.36E-04	9.03E-07	1.91E-03	4.22E-03
TAE	3.46E-04	1.12E-03	<u>1.71E-01</u>	4.27E-06	<u>3.11E-01</u>	7.29E-07	2.37E-02

As an overall summary for all data sets, Table 9 shows the number wins, ties, and losses on accuracy, minimum number of features, and *P* values measures for each algorithm. It appears that the GOA algorithm is superior on 12 data sets out of 18 in terms of accuracy. Moreover, in term of statistical significance, the results show that the GOA are better on 9, 4, 6, 7, 7, 10, and 7 data sets compared with MVO, GA, PSO, GWO, FF, BA, and CS, respectively. In addition, the Friedman test shows that the GOA obtains the best rank.

Experiment III: Comparisons with Metaheuristics in the Literature

In this section, we compare the results of the optimized SVM using GOA with grid search, and four popular

metaheuristics algorithms (MVO, GA, PSO, and GSA) that are obtained directly from the literature (12 data sets used). The comparison is conducted based on the system architecture used in these works [19, 23, 25]. Table 10 shows more details about the approaches that are used for the comparisons.

Table 11 shows the results of the GOA and other well-regarded approaches. It can be seen that GOA outperforms the grid search method in all data sets with notable differences in the accuracy results. In addition, GOA outperforms the other metaheuristics algorithms and performs reasonably in 8 data sets out of 12. Moreover, the results of the GOA for the other four data sets are very close to the other algorithms.

Taken together, the results of this section showed that the GOA along with SVM can make a very effective memetic, wrapper-based algorithm to solve a wider range

Table 9 Summary statistical results on a variety of measures (with feature selection)

Algorithm	GOA			MVO			GA			PSO			GWO			FF			BA			CS		
	W	T	L	W	T	L	W	T	L	W	T	L	W	T	L	W	T	L	W	T	L	W	T	L
Accuracy	12	1	7	1	0	17	0	0	18	1	0	17	0	0	18	0	1	17	1	1	16	1	0	17
Features	2	1	15	0	0	18	3	0	15	0	0	18	5	1	12	7	0	11	0	1	17	0	0	18
<i>P</i> value	–	–	–	1	8	9	2	12	4	1	11	6	1	10	7	0	11	7	1	7	10	2	9	7
Rank	2.31			4.83			4.86			4.36			5.08			5.31			5.61			3.64		

The table values represent the number of data sets each algorithm won/losses/ties on a variety of measures along with their rank using *F* test

Table 10 Key to comparative methods

Key	Method name	Population \times iteration	Data set number	Reference
GA-SVM	Genetic algorithm based feature selection and parameters optimization for support vector machines.	500 \times 600	11	[19]
MVO-SVM	A multi-verse optimizer approach for feature selection and optimizing SVM parameters based on a robust system architecture	50 \times 30	10	[15]
PSO-SVM	Particle swarm optimization for parameter determination and feature selection of support vector machines.	8 \times 250	17	[25]
GSA-SVM	Gravitational search algorithm and support vector machines hybrid system for classification of binary problems.	5 \times 20	8	[23]
Grid-SVM	Grid search	—	17	[25]
GOA-SVM	Grasshopper optimisation algorithm for feature selection and parameters optimization support vector machines	100 \times 200	18	Our

of classification problems. For one, the GOA algorithm addressed the high-dimensional nature of feature selection problems. This algorithm has been equipped with effective operators to avoid the local solutions of such problems and find a reasonable estimation of the global optimum in a reasonable time. For another, the SVM used the selected feature by the GOA algorithm to classify the data sets with an accurate classification rate. The outcome of using GOA and SVM to handle difficulties in two steps of classification proven to be very effective as per the results of this section.

As mentioned earlier, the No Free Lunch theorem states and has been logically proved that there is no general optimizer to solve all optimization problems. However, some algorithms might be beneficial for a set of problems with or without modification or hybridization. The operator integrated in GOA make it suitable for selecting an optimal

set of feature from a give large set of features. In fact, this algorithm benefits from a high exploratory behavior and this was the main reason of its success it this work. As per the results and No Free Lunch theorem, we state that this algorithm is worthy of use in the field of feature selection to assist classifier in finding a suitable set of features. It has the potential to leverage the whole classification process.

Conclusions

This work presents a novel hybrid approach for optimizing SVM based on the grasshopper optimisation algorithm (GOA). The proposed approach is able to tune the parameters of the SVM kernel and at the same time can find the minimal and optimal subset of the model features.

Table 11 Classification accuracy comparison between GOA and different methods in the literature

Data set	GOA	Grid search	MVO	GA	PSO	GSA
Wine	<i>100.00</i>	96.60	<i>100.00</i>	—	<i>100.00</i>	—
Ionosphere	<i>99.43</i>	93.08	99.15	98.56	99.01	96.01
Iris	<i>100.00</i>	96.00	—	<i>100.00</i>	99.20	—
Pima	81.52	76.69	—	81.50	82.68	<i>94.09</i>
Vehicle	<i>90.18</i>	84.28	89.00	84.06	89.83	—
Vowel	<i>100.00</i>	98.91	<i>100.00</i>	99.30	100.00	—
Australian	91.45	84.84	—	88.10	91.03	<i>95.71</i>
Breast cancer	<i>99.27</i>	96.64	<i>99.27</i>	96.19	99.18	<i>95.57</i>
German	82.20	75.30	79.20	85.60	81.62	<i>94.51</i>
Sonar	98.07	87.90	<i>99.02</i>	98.00	96.26	—
Parkinsons	<i>100.00</i>	—	99.50	—	—	—
Spectf	<i>96.25</i>	—	95.91	—	—	—

—The results of this data set are not available
(The best values are marked in italic)

This study shows that optimising SVM classifier by finding suitable features and the optimal kernel parameters simultaneously lead to better overall classification accuracy. The experimental results on the benchmark data sets revealed the effectiveness of the GOA on enhancing the SVM classifier accuracy. The GOA outperforms other optimisers such as GA, MVO, GWO, PSO, FF, BA, and CS in most of the data sets in terms of classification accuracy.

For future work, the application of the proposed GOA-SVM model to other real word problems can be investigated and applied. For example, the efficacy of the proposed model could be employed to tackle data mining problems in different domains such as real business problems in organizational environments. Furthermore, the performance of the model can be investigated on larger scale problems.

Acknowledgements The authors would like to thank Dr. Simon Andrews from Babraham Institute, Cambridge, UK for thoroughly proofreading this paper.

Compliance with Ethical Standards All procedures followed were in accordance with the ethical standards of the responsible committee on human experimentation (institutional and national) and with the Helsinki Declaration of 1975, as revised in 2008 (5).

Conflict of interests The authors declare that they have no conflict of interest.

Informed Consent All procedures followed were in accordance with the ethical standards of the responsible committee on human experimentation (institutional and national) and with the Helsinki Declaration of 1975, as revised in 2008 (5).

Human and Animal Rights This article does not contain any studies with human or animal subjects performed by any of the authors.

References

- Aljarah I, Faris H, Mirjalili S. Optimizing connection weights in neural networks using the whale optimization algorithm. *Soft Comput*. 2016; 1–15.
- Aljarah I, Faris H, Mirjalili S, Al-Madi N. Training radial basis function networks using biogeography-based optimizer. *Neural Comput Appl*. 2016; 1–25.
- Faris H, Aljarah I, Mirjalili S. Training feedforward neural networks using multi-verse optimizer for binary classification problems. *Appl Intell*. 2016;45(2):322–332.
- Faris H, Aljarah I, Al-Betar MA, Mirjalili S. Grey wolf optimizer: a review of recent variants and applications. *Neural Comput Appl*. 2017; 1–23.
- Arana-Daniel N, Gallegos AA, López-Franco C, Alanís AY, Morales J, López-Franco A. Support vector machines trained with evolutionary algorithms employing kernel adatron for large scale classification of protein structures. *Evol Bioinform Online*. 2016;12:285.
- Babaoğlu I, Findik O, Bayrak M. Effects of principle component analysis on assessment of coronary artery diseases using support vector machine. *Expert Syst Appl*. 2010;37(3):2182–2185.
- Bao Y, Hu Z, Xiong T. A pso and pattern search based memetic algorithm for svms parameters optimization. *Neurocomputing*. 2013;117:98–106.
- Beyer K, Goldstein J, Ramakrishnan R, Shaft U. When is “nearest neighbor” meaningful? In: International conference on database theory. Springer; 1999. p. 217–235.
- Blondin J, Saad A. Metaheuristic techniques for support vector machine model selection. In: 2010 10th International conference on hybrid intelligent systems (HIS). IEEE; 2010. p. 197–200.
- Boser BE, Guyon IM, Vapnik VN. A training algorithm for optimal margin classifiers. In: Proceedings of the fifth annual workshop on Computational learning theory. ACM; 1992. p. 144–152.
- Bouraooui A, Jamoussi S, BenAyed Y. A multi-objective genetic algorithm for simultaneous model and feature selection for support vector machines. *Artif Intell Rev*. 2017; 1–21.
- Chang CC, Lin CJ. Libsvm: a library for support vector machines. *ACM Trans Intell Syst Technol (TIST)*. 2011;2(3):27.
- Cortes C, Vapnik V. Support-vector networks. *Mach Learn*. 1995;20(3):273–297.
- Eswaramoorthy S, Eswaramoorthy S, Sivakumaran N, Sivakumaran N, Sekaran S, Sekaran S. Grey wolf optimization based parameter selection for support vector machines. *COMPEL-The Int J Comput Math Electr Electron Eng*. 2016;35(5):1513–1523.
- Faris H, Hassonah MA, Ala’M AZ, Mirjalili S, Aljarah I. A multi-verse optimizer approach for feature selection and optimizing svm parameters based on a robust system architecture. *Neural Comput Appl*. 1–15.
- Gepperth A, Karaoguz C. A bio-inspired incremental learning architecture for applied perceptual problems. *Cogn Comput*. 2016;8(5):924–934.
- Goldberg DE. Genetic algorithms in search, optimization and machine learning, 1st ed. Boston: Addison-Wesley Longman Publishing Co., Inc.; 1989.
- Huang CL. Aco-based hybrid classification system with feature subset selection and model parameters optimization. *Neurocomputing*. 2009;73(1):438–448.
- Huang CL, Wang CJ. A ga-based feature selection and parameters optimization for support vector machines. *Expert Syst Appl*. 2006;31(2):231–240.
- James G, Witten D, Hastie T, Tibshirani R. An introduction to statistical learning, vol. 6. Springer; 2013.
- Kanevski M, Pozdnukhov A, Timonin V. Machine learning algorithms for geospatial data applications and software tools. 2008.
- Kohavi R, John GH. Wrappers for feature subset selection. *Artif Intell*. 1997;97(1):273–324.
- Li C, An X, Li R. A chaos embedded gsa-svm hybrid system for classification. *Neural Comput Appl*. 2015;26(3):713–721.
- Lichman M. UCI machine learning repository; 2013. <http://archive.ics.uci.edu/ml>.
- Lin SW, Ying KC, Chen SC, Lee ZJ. Particle swarm optimization for parameter determination and feature selection of support vector machines. *Expert Syst Appl*. 2008;35(4):1817–1824.
- Liu X, Zeng Y, Zhang T, Xu B. Parallel brain simulator: a multi-scale and parallel brain-inspired neural network modeling and simulation platform. *Cogn Comput*. 2016;8(5):967–981.
- Liu Y, Wang G, Chen H, Dong H, Zhu X, Wang S. An improved particle swarm optimization for feature selection. *J Bionic Eng*. 2011;8(2):191–200.
- Mirjalili S, Mirjalili SM, Hatamlou A. Multi-verse optimizer: a nature-inspired algorithm for global optimization. *Neural Comput Appl*. 2016;27(2):495–513.
- Mirjalili S, Mirjalili SM, Lewis A. Grey wolf optimizer. *Adv Eng Softw*. 2014;69:46–61.

30. Mirjalili SZ, Mirjalili S, Saremi S, Faris H, Aljarah I. Grasshopper optimization algorithm for multi-objective optimization problems. *Appl Intell*. 2017; 1–16.
31. Nguyen MH, De la Torre F. Optimal feature selection for support vector machines. *Pattern Recog*. 2010;43(3):584–591.
32. Perdisci R, Gu G, Lee W. Using an ensemble of one-class svm classifiers to harden payload-based anomaly detection systems. In: *Sixth International conference on data mining, 2006. ICDM'06*. IEEE; 2006. p. 488–498.
33. Puntura A, Theera-Umporn N, Auephanwiriyakul S. Optimizing support vector machine parameters using cuckoo search algorithm via cross validation. In: *2016 6th IEEE International conference on control system, computing and engineering (ICCSCE)*. IEEE; 2016. p. 102–107.
34. Rodan A, Faris H. Credit risk evaluation using cycle reservoir neural networks with support vector machines readout. In: *Asian Conference on intelligent information and database systems*. Springer; 2016. p. 595–604.
35. Rodan A, Faris H, Alsakran J, Al-Kadi O. A support vector machine approach for churn prediction in telecom industry. *Int Inf Inst (Tokyo) Inf*. 2014;17(8):3961.
36. Samadzadegan F, Soleymani A, Abbaspour RA. Evaluation of genetic algorithms for tuning svm parameters in multi-class problems. In: *2010 11th International symposium on computational intelligence and informatics (CINTI)*. IEEE; 2010. p. 323–328.
37. Saremi S, Mirjalili S, Lewis A. Grasshopper optimisation algorithm: theory and application. *Adv Eng Softw*. 2017;105:30–47.
38. Sheta A, Ahmed SE, Faris H. A comparison between regression, artificial neural networks and support vector machines for predicting stock market index. *Int J Adv Res Artif Intell (IJARAI)*. 2015;4(7):55–63.
39. Sotiris VA, Peter WT, Pecht MG. Anomaly detection through a bayesian support vector machine. *IEEE Trans Reliab*. 2010;59(2):277–286.
40. Staelin C. Parameter selection for support vector machines. Hewlett-Packard Company, Tech. Rep HPL-2002-354R1. 2003.
41. Takeuchi K, Collier N. Bio-medical entity extraction using support vector machines. *Artif Intell Med*. 2005;33(2):125–137.
42. Tanveer M. Robust and sparse linear programming twin support vector machines. *Cogn Comput*. 2015;7(1):137–149.
43. Tuba E, Mrkela L, Tuba M. Support vector machine parameter tuning using firefly algorithm. In: *2016 26th International conference Radioelektronika (RADIOELEKTRONIKA)*. IEEE; 2016. p. 413–418.
44. Tuba E, Tuba M, Simian D. Adjusted bat algorithm for tuning of support vector machine parameters. In: *2016 IEEE Congress on evolutionary computation (CEC)*. IEEE; 2016. p. 2225–2232.
45. Vapnik V. An overview of statistical learning theory. *IEEE Trans Neural Netw*. 1999;5:988–999.
46. Vapnik V. The nature of statistical learning theory. Springer Science & Business Media. 2013.
47. Weston J, Mukherjee S, Chapelle O, Pontil M, Poggio T, Vapnik V. Feature selection for svms. In: *Proceedings of the 13th international conference on neural information processing systems*. MIT Press; 2000. p. 647–653.
48. Wolpert DH, Macready WG. No free lunch theorems for optimization. *IEEE Trans Evol Comput*. 1997;1(1):67–82.
49. Wu T, Yao M, Yang J. Dolphin swarm extreme learning machine. *Cogn Comput*. 2017;9(2):275–284.
50. Xin J, Chen G, Hai Y. A particle swarm optimizer with multi-stage linearly-decreasing inertia weight. In: *International Joint conference on computational sciences and optimization, 2009. CSO 2009, vol 1*. IEEE; 2009. p. 505–508.
51. Xu Y, Guo R, Wang L. A twin multi-class classification support vector machine. *Cogn Comput*. 2013;5(4):580–588.
52. Yang J, Gong L, Tang Y, Yan J, He H, Zhang L, Li G. An improved svm-based cognitive diagnosis algorithm for operation states of distribution grid. *Cogn Comput*. 2015;7(5):582–593.
53. Yang XS, Deb S. Cuckoo search via lévy flights. In: *World Congress on nature & biologically inspired computing, 2009. NaBIC 2009*. IEEE; 2009. p. 210–214.
54. Yang XS, Deb S, Fong S. Accelerated particle swarm optimization and support vector machine for business optimization and applications. In: *International conference on networked digital technologies*. Springer; 2011. p. 53–66.
55. Yang XS, He X. Firefly algorithm: recent advances and applications. *Int J Swarm Intell*. 2013;1(1):36–50.
56. Yang XS, Hossein Gandomi A. Bat algorithm: a novel approach for global engineering optimization. *Eng Comput*. 2012;29(5):464–483.
57. Yin S, Zhu X, Jing C. Fault detection based on a robust one class support vector machine. *Neurocomputing*. 2014;145:263–268.
58. Zhang H, Berg AC, Maire M, Malik J. Svm-knn: discriminative nearest neighbor classification for visual category recognition. In: *IEEE Computer society conference on computer vision and pattern recognition, 2006, vol. 2*. IEEE; 2006. p. 2126–2136.
59. Zhang J, Tittel FK, Gong L, Lewicki R, Griffin RJ, Jiang W, Jiang B, Li M. Support vector machine modeling using particle swarm optimization approach for the retrieval of atmospheric ammonia concentrations. *Environ Model Assess*. 2016;21(4):531–546.
60. Zhang X, Chen X, He Z. An aco-based algorithm for parameter optimization of support vector machines. *Expert Syst Appl*. 2010;37(9):6618–6628.