




A multi-objective feature selection method based on bacterial foraging optimization

Ben Niu¹ · Wenjie Yi¹ · Lijing Tan¹ · Shuang Geng¹  · Hong Wang¹

Published online: 27 July 2019
© Springer Nature B.V. 2019

Abstract

Feature selection plays an important role in data preprocessing. The aim of feature selection is to recognize and remove redundant or irrelevant features. The key issue is to use as few features as possible to achieve the lowest classification error rate. This paper formulates feature selection as a multi-objective problem. In order to address feature selection problem, this paper uses the multi-objective bacterial foraging optimization algorithm to select the feature subsets and k-nearest neighbor algorithm as the evaluation algorithm. The wheel roulette mechanism is further introduced to remove duplicated features. Four information exchange mechanisms are integrated into the bacteria-inspired algorithm to avoid the individuals getting trapped into the local optima so as to achieve better results in solving high-dimensional feature selection problem. On six small datasets and ten high-dimensional datasets, comparative experiments with different conventional wrapper methods and several evolutionary algorithms demonstrate the superiority of the proposed bacteria-inspired based feature selection method.

Keywords Feature selection · Multi-objective optimization · Bacterial foraging optimization · Information exchange mechanism

1 Introduction

With the development of information technology, the main challenge of data mining now is how to extract useful feature information from existing enormous data rather than how to collect a large amount of data. Feature selection can eliminate irrelevant or redundant features, so that it assists in reducing the number of features, improving model accuracy, and shortening running time. On the other hand, selecting truly relevant features can simplify the model, making it easier for researchers to understand how data is produced. Many researchers proposed various methods to select the most suitable features. Some previous

researches have viewed feature selection as a single-objective problem to minimize the classification error rate. Actually, feature selection problem can also be regarded as selecting a feature subset from an original set with the minimum feature subset size (Hamdani et al. 2007). This problem can be defined as a multi-objective problem.

Traditional feature selection algorithms can be classified into mainly two types: filter method and wrapper method (Jović and Bogunović 2015). The main principle of filter method is to use evaluation criteria to enhance correlation between individual features and classes, and to reduce correlation among features simultaneously. Filter methods save the training steps of classifier, as a result, the calculation time is generally less and the calculation complex is lower, which can quickly eliminate features with irrelevant correlation. However, this method has a tendency to choose redundant features for the reason that the correlation among features is not considered. Different from filter methods, wrapper methods use classifier to evaluate the classification performance of the selected feature subset. In general, wrapper methods obtain more efficient feature subsets than that of filter methods with lower classification error rate and fewer features.

Shuang Geng and Hong Wang contributed equally to this article.

✉ Shuang Geng
gracegeng0303@163.com

✉ Hong Wang
ms_hongwang@gmail.com

¹ College of Management, Shenzhen University, Shenzhen 518060, China

There exist many learning algorithms for feature evaluation in wrapper methods, such as decision tree, Bayesian classifier, and neighbor algorithm as well as support vector machine. Traditional wrapper methods include sequential forward selection (SFS) and sequential backward selection (SBS) (Xue et al. 2016). However, in these sequential search process, once the feature is added or removed, it will keep unchanged during the whole process. Consequently, it is easy to become trapped in local optima and can only obtain approximate optimal solution (Xue et al. 2014). To effectively address the existing deficiencies mentioned above, research communities began to use evolutionary algorithms with random search strategies to tackle feature selection problem. Those evolutionary algorithms includes simulated annealing (SA) (Lin et al. 2008), genetic algorithm (GA) (Zhu et al. 2007), particle swarm optimization (PSO) (Wang and Yan 2015), bacterial foraging optimization (BFO) (Wang et al. 2017; Chen et al. 2017; Wang and Niu 2017), to name but just a few. However, most of the existing feature selection methods mainly aim at improving the classification accuracy or reducing the classification error rate, which consider feature selection as a single-objective optimization problem. Actually, feature selection can also be considered as a multi-objective problem that aims at reducing the classification error rate with the minimum feature number, which is more general and more applicable.

In this paper, a novel feature selection approach based on multi-objective bacterial foraging optimization algorithm (MOBFO) is leveraged to address the drawbacks of the existing feature selection techniques. Furthermore, in order to tackle high-dimensional feature selection problems, four different information exchange mechanisms are introduced into the bacteria-inspired based feature selection method with the purpose of escaping from the local optima. The key idea of the proposed method is to select feature subsets based on MOBFO algorithm, and then evaluate the subsets of features by using the classification algorithm called KNN.

The main contributions of our approach are listed as follows:

- (1) Unlike previous feature selection methods, this paper considers the situation that the number of features is not fixed.
- (2) Under the condition of unfixed optional number of feature subset, this paper extends a single-objective feature selection problem to a multi-objective integer optimization problem with two main conflicting objectives: minimize both the classification error rate and the size of feature subset.
- (3) In order to ensure that there is no dominant relationship in the final solution set, this paper adds

a non-dominant sorting and crowding distance calculation technique to the algorithm.

This paper is divided into five sections. Section 2 reviews the literature on feature selection methods and bacteria-inspired algorithm. A new bacteria-inspired based feature selection approach is described in the third section. Section 4 presents the experimental design and results. Finally, conclusions are drawn in the final section.

2 Related work

Feature selection problems have been extensively studied by research communities. Compared with filter methods, the wrapper methods are more concerned because of its higher classification accuracy. From an optimization perspective, the development of intelligent algorithms provide new ideas as to how to solve the feature selection problem more efficiently. These kinds of approaches can be categorized into wrapper method. This section presents some related works on different feature selection methods and discusses the advantages and disadvantages of different approaches.

2.1 Traditional feature selection algorithms

As a classical problem in machine learning, traditional feature selection has been largely investigated in previous researches. Prior studies in traditional feature selection mainly follow two streams (Jović and Bogunović 2015). One stream takes feature selection as an independent process without classifiers, which we name it as filter methods. Filter approaches evaluate the data set directly by using evaluation criterions to enhance correlation between individual features and classes, and to reduce correlation among features simultaneously. Researchers adopt mainly four evaluation criterion, including the distance measurement, information measurement, correlation measurement, and consistency measurement.

Distance measurement uses distance as a measure of similarity between samples. The smaller the distance, the similar the samples are. Filter methods based on distance measurement consist Relief algorithm (Jia et al. 2013), branch and bound method (Dai and Yao 2017), mahalobis distance algorithm (Jin et al. 2012), as well as Bhattacharyys distance algorithm (Choi and Lee 2003) and so on. Distance-based filter methods are simple to calculate, however, they tend to select redundant features. In order to reduce the correlation between feature subsets, information measurement based filter methods use information gain or mutual information to effectively select the key features and eliminate irrelevant features. The information distance based methods include information gain

(IG) (Wang et al. 2011), minimum Redundancy Maximum Relevance (mRMR) (Peng et al. 2005), interact feature selection (Zhao and Liu 2009), redundancy-complementariness dispersion (Chen et al. 2015), mutual information (MI) (Bennasar et al. 2015) and so on. In recent years, how to develop an effective information measurement based filter method has become a hot research direction. However, with the increase of dimension of data, such algorithms still face the challenge of computational complexity (Xue et al. 2016).

As for the correlation measurement and consistency measurement, the former one uses the correlation coefficient to judge the correlation between features and classes to obtain the feature subset (Ozturk et al. 2013), while the latter one is dedicated to finding the minimum-size feature subset which achieve the same effect as the whole feature set (Dash et al. 2000). In summary, the major advantage of filter methods mentioned above is that these methods save the training steps of classifier so that calculation time is generally less and the calculation complex is lower, which quickly eliminate features with irrelevant correlation. Unfortunately, this kind of method tends to choose redundant features for the reason that the correlation among features is not considered.

Another stream of research takes feature selection as the process with classifiers to evaluate the performance of the selected feature subset, which is called wrapper methods. Conventional wrapper methods can be categorized into sequential forward selection (SFS) and sequential backward selection (SBS) (Xue et al. 2016). In wrapper methods, once the feature is added or removed, it will remain constant during the whole process. Therefore, it is easy to get stuck in a local optimum and only obtain approximate optimal solution (Xue et al. 2014). Generally speaking, wrapper methods obtain more efficient feature subsets than filter methods with lower classification error rate and fewer features. However, the efficiency of feature subset generation is low due to the fact that the classification algorithm need to be performed frequently. It is necessary to design an algorithm to improve the efficiency of the wrapper approaches. To achieve the purpose of dimensionality reduction, how to develop high-precision and high-efficiency wrapper approaches for feature selection problems has become a hot research topic.

2.2 Evolutionary based feature selection algorithms

In recent years, there has been an increasing amount of literature on using evolutionary algorithms to address feature selection problems. This type of algorithms construct the feature selection into the optimization problem with classification accuracy or classification error rate as the objective function, and the corresponding optimization

solution is the selected feature subset. Hsu (2004) adopted decision-making tree method to select features and further used genetic algorithm to seek the feature subsets which can make the classification error rate reach the lowest. Similarly, Chiang and Pell (2004) also introduced genetic algorithm into feature selection process. Researchers also use ant colony algorithm to select the most suitable feature subsets. As an illustration, Kashef and Nezamabadi-Pour (2015) modified the original ant colony algorithm and employed the improved algorithms into the feature selection problem solving. In that method, features are regarded as graph nodes and ant colony algorithm is employed to choose nodes. Several simulation experiments on UCI datasets have demonstrated this method's effectiveness.

However, for high-dimensional feature selection problems, such evolutionary algorithms require plenty of computational time to evaluate the combined effect of all possible feature subsets. Therefore, such feature selection methods which depend on evaluating the combined effect of all feature subsets are not feasible in the case of high-dimensional feature selection problem. In order to resolve this problem, Wang and Yan (2015) put forward a PSO-based method, which views features as optimization variables, sets the weight for the feature subsets on the basis of the classification performance, and selects the feature subset with the best classification performance. Furthermore, Xue et al. (2014) discussed and compared the influence of different initialization strategies on feature selection, and drew a conclusion that adopting SFS and SBS simultaneously in PSO can reduce the computational complexity and get better results.

Actually, those methods which are based on random search strategies, including random generation sequence selection algorithm (RGSS) (Park and Kim 2015), simulated annealing algorithm (SA) (Lin et al. 2008), genetic algorithm (GA) (Zhu et al. 2007) and many others, are the most commonly used techniques. Most evolutionary algorithm based feature selection methods belong to the wrapper methods. The principle idea is to use optimization algorithm to select feature subsets firstly and then use classification algorithms like KNN as an evaluation function. The key advantage of evolutionary algorithm based methods is that these approaches are always with less control parameters and strong robustness.

2.3 Bacteria-inspired algorithms

2.3.1 Original bacteria-inspired algorithm

Most swarm intelligent computation or optimization methods are based on higher organism with more complex behaviours. Particle swarm optimization (PSO) gets inspiration from the social behaviours of bird and fish (Eberhart and Kennedy 1995). Ant colony optimization (ACO) is to

mimic the foraging behaviours of nature ants (Dorigo et al. 1996). Artificial bee colony (ABC) is proposed to simulate the intelligent foraging behaviour of a honey bee swarm (Karaboga 2005). However, the biological behaviours mentioned above are relatively complicated and many of them are difficult to be described qualitatively. Because of this, some assumptions factors need to be added to the proposed optimization model. In this way, although the established optimization model reflects some characteristics of biological systems, it still cannot completely describe the actual situations of the biological systems. Therefore, the results of the optimization problems would be affected to some extent.

Under this circumstance, several studies have focused on the behaviours of microorganisms which are easier to be described qualitatively. Bacteria, as the simplest unicellular organisms, have simple patterns of behaviours which can be easily described. Besides, as one of the oldest biological creatures on earth, bacteria's strong vitality and abilities to flexibly adapt to the complicated environment fully demonstrate their optimization instinct in the process of survival activities. For the two advantages above, research communities have developed the bacteria-inspired (BI) based methods from a new perspective. These BI-based techniques are inspired by the social behaviours of low-grade microorganism bacteria, and further consider the foraging process of bacteria as the optimization problem solving process. To be more specific, the bacteria-inspired algorithm primarily stimulates three typical bacterial foraging behaviours, including chemotaxis, reproduction, and elimination/dispersal.

As the original BI-based algorithm (Passino 2002), bacterial foraging optimization (BFO) is summarized as:

- (i) *Initialization* BFO generates initial population randomly and then calculates the fitness value of every individual by using the fitness function. In the first iteration, the individual with the best fitness function will be regarded as the best individual.
- (ii) *Chemotaxis operator* All the bacterial individuals tend to avoid the harmful environment and choose to move to favorable one. This kind of behavior is called chemotaxis. During the chemotaxis process, bacteria's movement behaviors, including tumbling and swimming, are stimulated. Tumbling refers the bacterial behavior of moving a unit step in any direction. When the bacteria complete the tumbling movement, if the fitness value is better than that last time, then the bacteria will keep on moving in the same direction.
- (iii) *Reproduction operator* The biological evolution process follows the law of survival of the fittest in

nature. After the chemotaxis process, those individuals with poor foraging ability will be eliminated, while the other bacteria with strong foraging ability will survive and further reproduce their next generation by splitting to produce two identical cells. In this way, the stability of the whole population is guaranteed, and the quality of the population is improved.

- (iv) *Elimination/dispersal operator* During the chemotaxis and reproduction process, it is likely that the survival environment of the population would be damaged so that some of the individuals would die or migrate to another regions. This elimination/dispersal operator is able to control the exploration processes and escape from the local optima with a certain probability.

2.3.2 Multi-objective bacteria-inspired algorithm

The original bacterial foraging optimization algorithm is developed for the single-objective problems. When it comes to the problems with more than one objective, the original bacterial foraging optimization algorithms cannot be applied directly for the reason that the solution is no longer an absolute global optimal value but a non-dominant solution set. In order to propose a suitable method for multi-objective feature selection problem, the most important point is to select a good leader for bacterial population from a set of potential non-dominated solutions. Non-dominated sorting genetic algorithm (NSGAI) is one of the most widely known evolutionary optimization techniques. Niu et al. (2013) modified the original BFO algorithm into multi-objective algorithm by adding non-dominant sorting mechanism and crowding distance calculation from NSGAI (Deb et al. 2002) to construct the dominant population front in the non-dominant hierarchy. This multi-objective bacteria-inspired algorithm was then extended to improve the accuracy and diversity of the non-dominant frontier, introducing two neighborhood search strategies which are based on the ring topology and star topology respectively.

3 Multi-objective approach

In this paper, according to the characteristic of feature selection problem, we further improve the performance of the multi-objective bacteria-inspired algorithm and bring forward a new variant of MOBFO for feature selection problem (MOBIFS for short).

3.1 Mapping scheme

When using MOBIFS to seek the most appropriate feature subset, every bacterium generates a potential solution satisfying problem constraints. More specifically, each bacterium is endowed with three attributes, i.e. the features being selected, value of corresponding classification error rate and the size of feature subset. As shown in the following equations, n and m stand for the number of features and the bacterial population size respectively. Equation (1) demonstrates the coding of every bacterium’s first attribute about the selected features. When the element in the matrix fi is zero, it means that the feature is not selected, whereas if the value is non-zero, it means that the feature is selected. Equation (2) shows the coding of the whole population’s first attribute about the selected features. After calculating the fitness value, the classification error rate of every bacterium can be obtained. Matrix fit_1^i is set up to store the values of classification error rate. And then the number of features being selected are counted and stored in another matrix called fit_2^i .

$$fi = [xi1, xi2, xi3 \dots xim], \quad i = 1, \dots m \tag{1}$$

$$P = [f'_1, f'_2, \dots, f'_m] \tag{2}$$

$$Pop = \begin{bmatrix} P \\ fit_1^i \\ fit_2^i \end{bmatrix}, \quad i = 1, \dots m \tag{3}$$

3.2 Two important mechanisms of MOBIFS

Before describing the computational steps to deal with feature selection problem, we briefly introduce two important mechanisms of the proposed algorithm.

3.2.1 Wheel roulette mechanism

In the process of constructing a feature subset, it is inevitable that an individual would choose repetitive feature. For instance, if the result of an individual is [55.13 20.54 85.54 54.86], the processing becomes [55 21 86 55]. It means 55 is selected twice, which is not allowed. So we need to find another feature to replace the number of 55. The method should replace the repeated features within a reasonable range, and ensure the feature subset converges to the optimal subset more quickly.

According to the reference Khushaba et al. (2011), a distribution factor is proposed to do the task of replacing repeated features. As depicted in Table 1, wheel roulette mechanism uses a weighted scheme to calculate the probability of each feature being selected. The greater the probability, the higher the probability of being chosen. The concrete calculation of positive factor PD_j , negative factor

Table 1 The principle of wheel roulette mechanism

Step 1	Calculate the fitness function value of each individual
Step 2	Individuals whose values are smaller than the average fitness function value of the group are classified as A regions and individuals whose values are larger than the average fitness function of the group are classified as B regions
Step 3	Reflect the bacterial population to binary matrix
Step 4	Count the distribution of each feature in A and B regions, recorded as PD_j and ND_j
Step 5	Calculate the distribution factor FD_j

ND_j , and distribution factor FD_j are referred to the literature (Deb et al. 2002).

3.2.2 External archive management mechanism

3.2.2.1 Non-dominance sorting In MOBIFS, non-dominant solutions obtained during the optimization process would be saved to external archive. However, increasing the solution set slows down the convergence speed, so we must limit the size of the external archive. Therefore, how to maintain and manage the external archive is an important component of this algorithm. The flowchart of non-dominance sorting is given in Fig. 1.

3.2.2.2 External archive updating mechanism As Fig. 2 presents, when all the bacteria in current population completed the location updating operation, the non-dominated mechanism would be carried out within the population first. Then all individuals in the population will be given a rank result and those bacteria with the highest rank will be selected. Those bacteria with duplicate values are removed at the same time. Next, individuals which are selected through above steps will be compared with the elite individuals in the previous external archive.

The purpose of this operation is to avoid the dominance relationship of the solutions between individuals within a population. Therefore, the non-dominated mechanism should conduct the pairwise comparison of fitness value between bacterium in the current population and that in the external archive. After the comparison, the external archive will be updated by eliminating the bacterium with worse fitness value.

If all individual compared with bacteria in the external archive directly, we can only ensure that each individual in the population has no dominate relationship with the external archive’s bacteria. But we cannot guarantee that there is no dominate relationship among individuals within the population itself. Because of this, the final solution set we obtain may still exist dominate relationships.

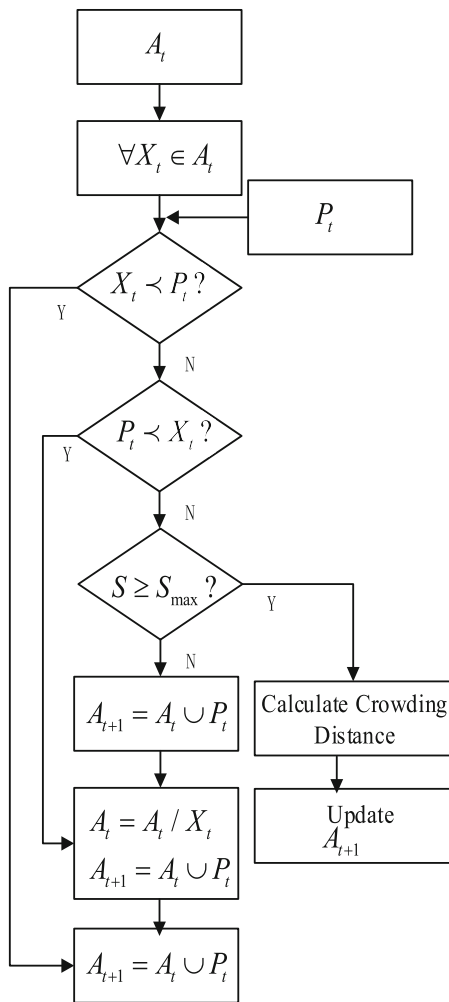


Fig. 1 Non-dominance sorting

3.3 Information exchange mechanisms of MOBIFS

To alleviate the stagnation in local optima, four different information exchange strategies are incorporated to the MOBIFS with the consideration of the lack of information communication. Researchers have investigated different topology structures in their previous work (McNabb et al. 2009). Neighborhood topology is considered as an effective mechanism because this kind of methods facilitate the bacteria converge to the global optima. With such kind of topology structures, every individual in the bacterial group learns from each other aim at obtain useful information to guide their forging behaviors during the whole process.

In this paper, four information communicational systems are chosen and integrated in MOBIFS, including Elite Learning, Ring topology, Star topology, and Von Neumann topology. The corresponding proposed algorithms are named as MOBIFS-EL, MOBIFS-RI, MOBIFS-ST and MOBIFS-VN, respectively. Table 2 gives detailed

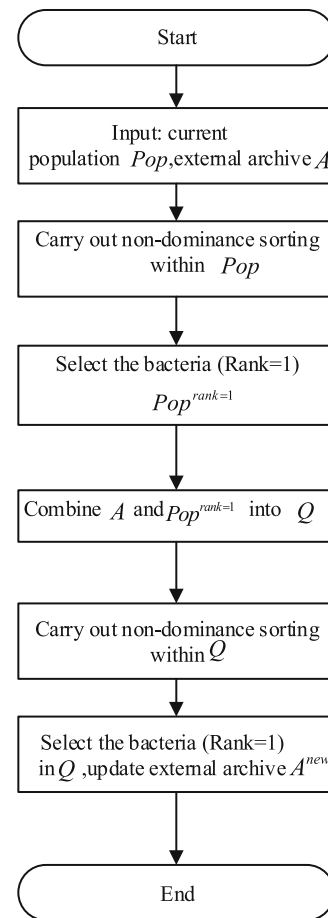


Fig. 2 External archive updating

information about the definitions of the variables firstly, while Table 3 illustrates different bacterial position updating formulas under different information exchange mechanisms.

3.4 Computational steps of MOBIFS algorithm

Based on the above important mechanisms, we propose a multi-objective bacteria-inspired algorithm for feature selection problem. Besides, four different information exchange mechanisms are integrated into the bacteria-inspired feature selection algorithm to enhance the performance of the method. Table 4 provides the pseudo-code of MOBIFS algorithm.

4 Experimental design

4.1 Datasets and comparison techniques

To test the performance of the proposed MOBIFS method on low-dimensional datasets, six small datasets with less

Table 2 Parameters and definitions

Parameters	Definitions
$\theta^i(j, k, l)$	The position of the i th bacterium at j th chemotaxis, k th reproduction and l th elimination-dispersal step
S	The population size of the bacteria
N_c, N_{re}, N_{ed}	Maximum number of iterations of chemotaxis, reproduction, elimination/dispersal
$C(i)$	The chemotactic step of the i th bacterium
rep_size	The size of the external archive
$elite_rank_{d-1}^b(j, k, l)$	The position of two individuals at $(d - 1)^{th}$ rank in the external archive; d : the current rank of the bacterium, $b = 1, 2$
$elite(j, k, l)$	The position of one random individual in the external archive
c_i	Learning factor; $i = 1, 2, 3, 4$
r_i	Random number; $i = 1, 2, 3, 4$
i_{hub}	The hub of the bacteria population in star topology
$ileft/iright/iabove/ibelow$	The left/right/above/below neighbor of i th bacterium

Table 3 Equations for the bacteria position updating

Algorithm	Equation
MOBIFS-EL	$\theta^i(j + 1, k, l) = w * \theta^i(j, k, l) + c_1 r_1 (pbest(j, k, l) - \theta^i(j, k, l)) + c_2 r_2 (elite(j, k, l) - \theta^i(j, k, l))$ (4)
MOBIFS-RI	$\theta^i(j + 1, k, l) = w * \theta^i(j, k, l) + c_1 r_1 (\theta^{ileft}(j, k, l) - \theta^i(j, k, l)) + c_2 r_2 (\theta^{iright}(j, k, l) - \theta^i(j, k, l))$ (5)
MOBIFS-ST	$\theta^i(j + 1, k, l) = w * \theta^i(j, k, l) + c_1 r_1 (\theta^{ihub}(j, k, l) - \theta^i(j, k, l))$ (6)
MOBIFS-VN	$\theta^i(j + 1, k, l) = w * \theta^i(j, k, l) + c_1 r_1 (\theta^{ileft}(j, k, l) - \theta^i(j, k, l)) + c_2 r_2 (\theta^{iright}(j, k, l) - \theta^i(j, k, l)) + c_3 r_3 (\theta^{iabove}(j, k, l) - \theta^i(j, k, l)) + c_4 r_4 (\theta^{ibelow}(j, k, l) - \theta^i(j, k, l))$ (7)

than one hundred features, named Wine, Australian, Zoo, German, Ionosphere, Lung cancer, are used as the benchmark datasets. Moreover, in order to examine the proposed method’s performance on datasets with higher dimensions, another ten datasets are chosen. They have different numbers of features (from 2309 to 15,009), classes (from 2 to 26), and instances (from 50 to 308). All data sets are collected from UCI machine learning repository (Frank and Asuncion 2010). Tables 5 and 6 summarize the characteristics of the low-dimensional datasets and high-dimensional datasets respectively. As for the justification of dataset selection, the main principle is to choose the same datasets as previous works (Khushaba et al. 2011; Xue et al. 2013; Yang et al. 2010; Chuang et al. 2008) so that

the experimental results can be directly compared with the figures in literatures.

For the experiments on small data sets, two single-objective algorithms and another two conventional wrapper methods are used as comparison methods. Two single objective algorithms are existing PSO-based feature selection methods, including commonly used PSO algorithm (ERFS) (Kennedy and Eberhard 1997; Lin et al. 2008; Chuang et al. 2011) and PSO with a two-stage fitness function (2SFS) (Xue et al. 2012). The main difference of these two algorithms is the fitness function. To specify, ERFS employs the fitness function that only takes the classification error rate into account. 2SFS divides the whole evolutionary process into two stages, where fitness function considers the classification performance in the first stage, and includes the number of features in the second stage.

Other traditional wrapper methods are linear forward selection (LFS) (Gutlein et al. 2009) and greedy stepwise backward selection (GSBS) (Caruana and Freitag 1994). These two conventional methods are derived from SFS and SBS, respectively. For the LFS approach, it limits the number of features that are considered in every step of the forward selection, so that the number of evaluations is reduced. As a result, the LFS method can lower computation costs and obtain better results than SFS. Unlike LFS method with a forward search, GSBS chooses a backward search. This method starts with all features and stops as long as removing any remaining features leads to a decline in evaluation. The experimental results of these four comparison methods are directly derived from Xue et al. (2013).

For the experiments on high-dimensional data sets, three evolutionary algorithms, namely Differential Evolution

Table 4 The pseudo-code of MOBIFS

Start

Initialize parameters and bacteria's location

$$p = 10, S = 200, N_c = 200, N_s = 4, N_{re} = 4, N_{ed} = 2, P_{ed} = 0.25; C(i); \theta^i(j, k, l); rep_size = 200$$

For ($l = 1: Ned$):

For ($k = 1: Nre$):

For ($j = 1: Nc$):

 Do chemotaxis steps;

 Update the external archives using non-dominance mechanism in Fig. 2;

 Update position according to different equations in Table 3 and fitness;

 Do boundary control;

End

 Do reproduction steps based on health evaluation mechanism ;

End

 Do elimination/dispersal steps;

End

Output: the non-dominant solutions in external archive

Table 5 Description of the small datasets employed

Datasets	#Features	#Classes	#Instances
Wine	13	3	178
Australian	14	2	690
Zoo	17	7	101
German	24	2	1000
Ionosphere	34	2	351
Lung cancer	56	3	32

Table 6 Description of the high-dimensional datasets employed

Dataset	#Features	#Classes	#Instances
9_Tumors	5726	9	60
11_Tumors	12,533	11	174
14_Tumors	15,009	26	308
Brain_Tumor1	5920	5	90
Brain_Tumor2	10,367	4	50
DLBCL	5470	2	77
Leukemia1	5328	3	72
Leukemia2	11,225	3	72
SRBCT	2309	4	83
Prostate tumor	10,509	2	202

based feature selection method (DEFS) (Khushaba et al. 2011), Information Gain-Genetic Algorithm (IG-GA) (Yang et al. 2010), and Improved Binary Particle Swarm

Optimization (IPPSO) (Chuang et al. 2008), are adopted as comparison methods. Besides, to further improve the performance on solving problems with high dimension, four different swarm search strategies are applied in the foraging process of bacteria, so that MOBFO algorithm is able to improve the parallel processing ability and considerably enhance the search efficiency. Specially, four information exchange mechanisms, including elite learning strategy, Ring topology, Star topology, Von Neumann topology, are integrated into the MOBIFS algorithm. Therefore, various MOBIFS methods are developed and employed to solve the problems at the same time.

Experiments are conducted in MATLAB environment. Instances of each dataset are randomly divided into two sets (Wang et al. 2017), including 30% testing set and 70% training set. Each bacterium is regarded as a feature subset during the training process, and evaluated by the classification algorithm called KNN during the training process as well as the testing process. What needs to be emphasized is that one of the optimal objectives is the classification error rate during the testing process rather than the training process.

When bacteria-inspired algorithm is used to solve practical problems, the population size is usually set between 100 and 200. The size of the external archive is always set the same number of the population size. Therefore, in the MOBIFS algorithm, the size of population is 200, and the external archive size is 200 as well. The upper limit of the number of feature depends on maximum

value of the dataset itself, and the lower limit is set to 1 uniformly. In addition, the maximum number of iterations is set to 100 in the experiment.

4.2 Results and analysis

Figure 3 shows the experimental results attained by MOBIFS on six small benchmark datasets and the evaluation algorithm is KNN. As shown in the experimental curve, MOBIFS technique gets four or more solutions, most of which can achieve better performance than using all features in all benchmark datasets. For instance, for

wine dataset, MOBIFS use only 5 features to achieve the classification error rate of 3.77% while using all 13 features can only get the classification error rate of 23.46%. This means that MOBIFS achieve the purpose of feature reduction without increasing the classification error rate.

Actually, feature selection is originally a discrete problem, so it is understandable that there are fewer non-dominant solutions on account of the fact that there are few individuals which satisfy the discrete condition. What's more, if the design of the algorithm is not good, there are no solutions directly.

Fig. 3 Experimental results on six small datasets

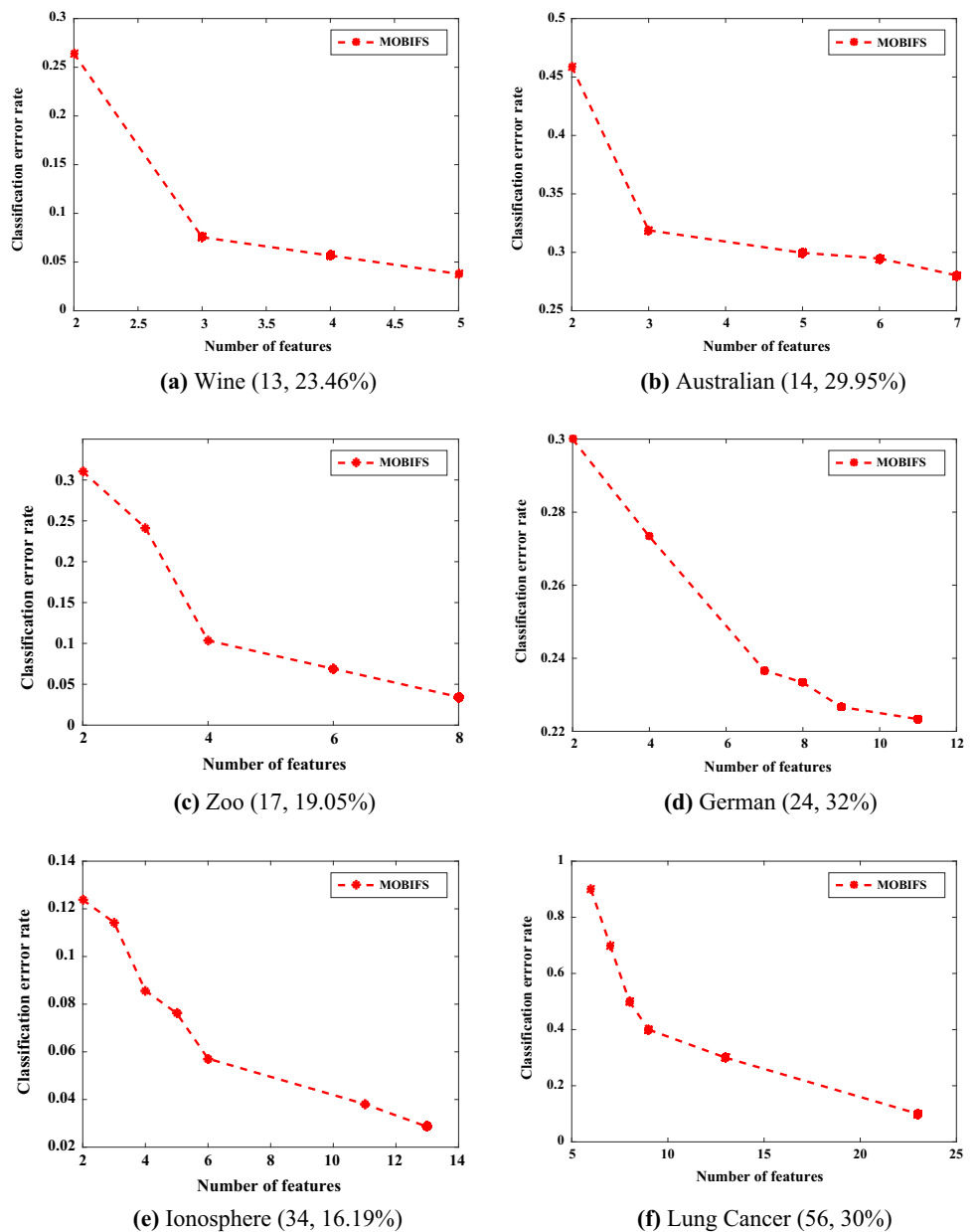


Table 7 Experimental results on small datasets

Datasets	Average	Method					
		All	LFS	GSBS	ERFS	2FES	MOBIFS
Wine	A_Size	13	7	8	8	8	5
	A_Error	23.46	25.93	14.81	4.04	4.04	3.77
Australian	A_Size	14	4	12	3.88	3.42	7
	A_Error	29.95	29.95	30.43	14.52	15.76	28.02
Zoo	A_Size	17	8	7	9.18	9.18	8
	A_Error	19.05	20.95	20	4.5	4.5	3.45
Ionosphere	A_Size	34	4	30	12.58	12.05	13
	A_Error	16.19	13.33	21.9	11.6	11.86	2.86
German	A_Size	24	3	18	13.48	11.92	11
	A_Error	32	31.33	35.67	11.6	11.86	22.33
Lung cancer	A_Size	56	6	33	27.35	27.38	23
	A_Error	30	10	10	28	27.75	10

From the data in Table 7, it is apparent that MOBIFS achieves a lower classification error rate with fewer features in most cases. It is worth mentioning that the final solution of MOBIFS is a non-dominated solution set, which is different from the result of other comparison methods. We use the solution with the minimum classification error rate and minimum size of feature subset to compare with other methods' average subset's size and average classification error rate.

For Australian dataset, ERFS and 2FES outperform MOBIFS in terms of the number of features and the classification error rate, LFS use fewer features than MOBIFS but the classification accuracy is slightly lower than MOBIFS. For German dataset, with a slightly larger number of features, ERFS and 2FES get a lower classification error rate than MOBIFS. For Lung cancer dataset, the performance of MOBIFS is only slightly worse than LFS but better than the other three comparison methods.

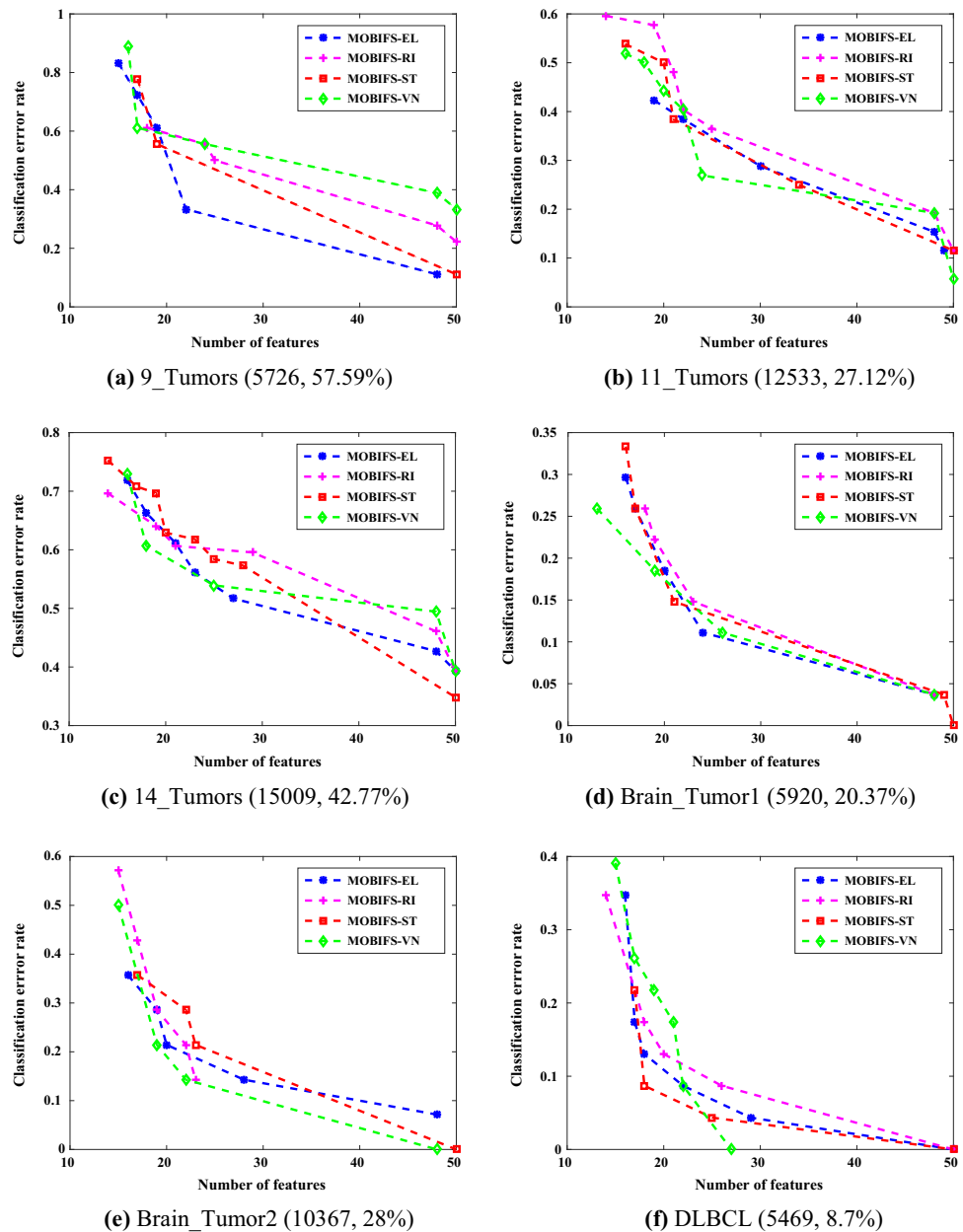
When it comes to the experimental results on data sets with high dimension, as shown in Fig. 4, four MOBIFS variants with different information exchange mechanisms obtain more than three solutions respectively. All of the solution obtained can reach the higher classification accuracy with much fewer features compared with using all the features in the dataset with no selection process. Take 9_Tumors dataset as an example, MOBIFS-EL algorithm is capable to select the most significant 48 features to reflect the key information in the dataset, with the classification error rate at only 11.11%. By contrast, if all the 5726 features are applied in the classification process, the classification error rate achieve 57.59%, which is almost five times as the figure in MOBIFS-EL. That is to say, MOBIFS methods have a good performance in dealing with high-dimensional feature selection problems. As displayed in Fig. 4, there is no one specific MOBIFS method can always

get the best solution for each dataset. In fact, it is associated with the characteristic of the benchmark datasets. Four search strategies are designed so as to get high-quality solution of different kinds of datasets, so that the decision maker can choose a better feature selection results for reference from all the final experimental results various MOBIFS obtained.

The results of comparisons between the MOBIFS techniques and other three evolutionary algorithms are displayed in Table 8. In general, MOBIFS method has better feature selection performance in all ten high-dimensional datasets, that is, MOBIFS technique can reach the highest classification accuracy with fewer feature numbers or at least it can achieve similar value of the classification error rate while using much fewer features.

As illustrated in Table 8, IG-GA and IBPSO are capable to reach the low classification error rate. However, for this two evolutionary algorithms, the size of the selected feature subset is still rather large. As for DEFS method, it gets a better performance in both of the classification error rate and the feature subset size. Even so, the multi-objective bacteria-inspired based method has a lower classification error rate in most cases than the figure in the DEFS method while using similar numbers of features. On top of that, as a multi-objective method, MOBIFS offers the decision maker different feature selection solutions with different feature subset size simultaneously while other three comparison methods can just only provide solutions with the fixed size of feature numbers.

Fig. 4 Experimental results on ten high-dimensional datasets



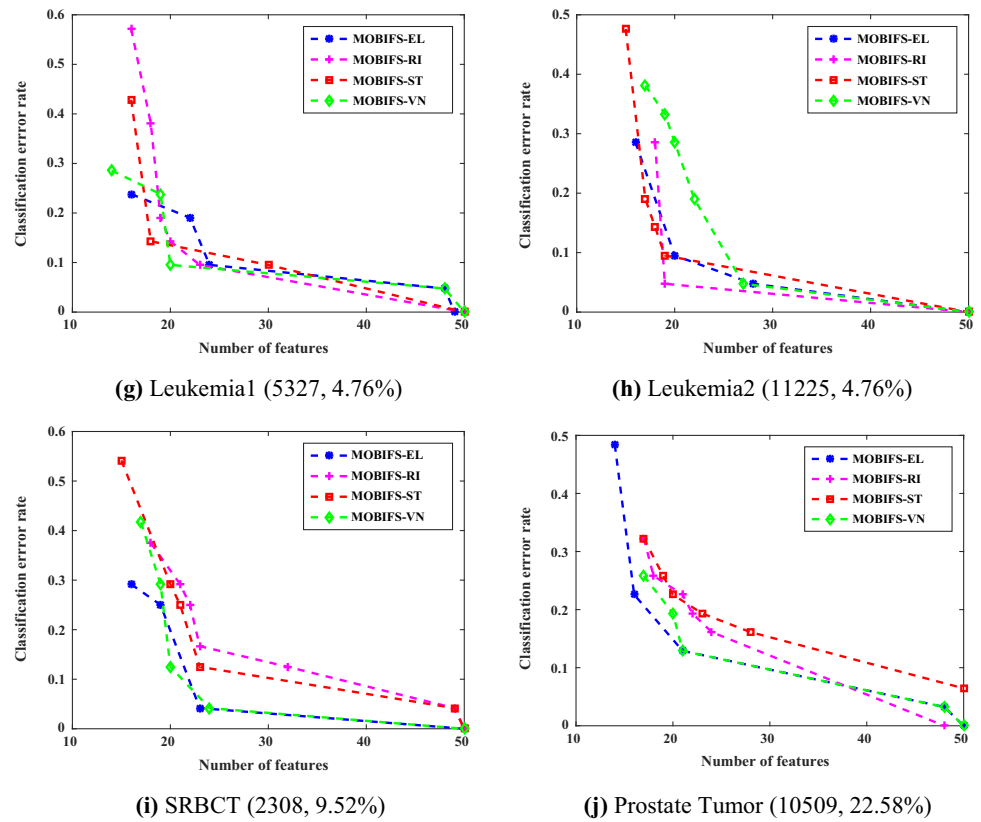
5 Conclusion

This paper formulates the feature selection problem as a multi-objective problem to minimize the classification error rate and the number of features being selected simultaneously. Then, a novel method to support the task of selecting the best features subset is investigated. Four information communication strategies are incorporated into the bacteria-inspired algorithm so that every individual is able to exchange information with each other and then use the useful information to guide their search. The principle of the proposed method is to use MOBFO

algorithm to select feature subset, and to use the classification algorithm called KNN to evaluate subsets of features.

Compared with two single-objective algorithms (ERFS, 2SFS) and other two conventional wrapper methods (LFS, GSBS), simulation results on six small datasets demonstrate that the proposed method has high effectiveness and efficiency in most cases. When it comes to the performance in high-dimensional feature selection problems, on ten benchmark datasets, the proposed MOBIFS algorithms with different information exchange mechanisms (Elite Learning, Ring topology, Star topology and Von Neumann topology) are capable of finding the most representative

Fig. 4 continued

**Table 8** Comparisons between evolutionary algorithms (KNN) on high-dimensional datasets

Datasets	Average	Algorithms				
		All	DEFS (Khushaba et al. 2011)	IG-GA (Yang et al. 2010)	IBPSO (Chuang et al. 2008)	MOBIFS
9_Tumors	A_Size	5726	40	52	1280	48
	A_Error	0.5759	0.204	0.15	0.2167	0.1111
11_Tumors	A_Size	12533	40	479	2948	50
	A_Error	0.2712	0.0678	0.0747	0.069	0.1154
14_Tumors	A_Size	15009	50	810	2777	50
	A_Error	0.4277	0.377	0.3474	0.3344	0.3483
Brain_Tumor1	A_Size	5920	30	244	754	50
	A_Error	0.2037	0.037	0.0667	0.0556	0
Brain_Tumor2	A_Size	10367	25	489	1197	48
	A_Error	0.28	0.025	0.12	0.06	0
DLBCL	A_Size	5469	15	343	1294	27
	A_Error	0.087	0.015	0.0392	0.0771	0
Leukemia1	A_Size	5327	13	82	1034	49
	A_Error	0.0476	0	0	0	0
Leukemia2	A_Size	11225	10	782	1292	27
	A_Error	0.0476	0	0.0139	0	0
SRBCT	A_Size	2308	11	56	431	50
	A_Error	0.0952	0	0	0	0
Prostate tumor	A_Size	10509	5	107	1042	48
	A_Error	0.2258	0	0	0	0

The bold value indicates the minimum value of average classification error rate in the specific datasets

feature subset with fewer features to reach the lower classification error rate. The related experimental results suggest that different strategies are suitable for different kinds of datasets. In addition, two evolutionary algorithms (DEFS, IG-GA, and IBPSO) are also selected as the comparison algorithms. Overall, the simulation results support that the proposed MOBIFS methods outperform other evolutionary algorithms in terms of the classification error rate and the size of the selected feature subsets.

In the proposed method, the number of features is not fixed as the prior studies assumed. Therefore, this approach does not depend on prior knowledge of the datasets. By applying this method to the feature selection problems, decision makers are capable to obtain a series of solutions, each of which gives a clear idea concerning which features are selected and the corresponding classification error. Under this circumstance, decision makers can choose the one that they consider to be most suitable from the solutions obtained. While this study contributes novel insights into the optimization based feature selection method, future research should be undertaken to investigate how the proposed MOBIFS method performs on a diverse range of real-world feature selection problems.

Acknowledgements This work is partially supported by The National Natural Science Foundation of China (Grants Nos. 71571120, 71271140, 71471158, 71001072, and 61472257). Natural Science Foundation of Guangdong Province (2016A030310074, 2018A030310575), Shenzhen Science and Technology Plan (CXZZ20140418182638764), Research Foundation of Shenzhen University (85303/00000155), and Research Cultivation Project from Shenzhen Institute of Information Technology (ZY201717).

References

- Bennasar M, Hicks Y, Setchi R (2015) Feature selection using joint mutual information maximisation. *Expert Syst Appl* 42(22):8520–8532
- Caruana R, Freitag D (1994) Greedy attribute selection. In: *Machine learning proceedings*, pp 28–36
- Chen ZJ, Wu CZ, Zhang YS, Huang Z, Ran B, Zhong M et al (2015) Feature selection with redundancy-complementariness dispersion. *Knowl Based Syst* 89:203–217
- Chen YP, Li Y, Wang G et al (2017) A novel bacterial foraging optimization algorithm for feature selection. *Expert Syst Appl* 83:1–17
- Chiang LH, Pell RJ (2004) Genetic algorithms combined with discriminant analysis for key variable identification. *J Process Control* 14(2):143–155
- Choi E, Lee C (2003) Feature extraction based on the Bhattacharyya distance. *Pattern Recognit* 36(8):1703–1709
- Chuang LY, Chang HW, Tu CJ, Yang CH (2008) Improved binary PSO for feature selection using gene expression data. *Comput Biol Chem* 32(1):29–38
- Chuang LY, Tsai SW, Yang CH (2011) Improved binary particle swarm optimization using catfish effect for feature selection. *Expert Syst Appl* 38(10):12699–12707
- Dai Q, Yao C (2017) A hierarchical and parallel branch-and-bound ensemble selection algorithm. *Appl Intell* 46:1–17
- Dash M, Liu H, Motoda H (2000) Consistency based feature selection. *Pacific-Asia conference on knowledge discovery and data mining*. Springer, Berlin, pp 98–109
- Deb K, Pratap A, Agarwal S et al (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans Evol Comput* 6(2):182–197
- Dorigo M, Maniezzo V, Colomi A (1996) Ant system: optimization by a colony of cooperating agents. *IEEE Trans Syst Man Cybern Part B Cybern* 26(1):29–41
- Eberhart R, Kennedy J (1995) A new optimizer using particle swarm theory. In: *Proceedings of the sixth international symposium on micro machine and human science*. IEEE, pp 39–43
- Frank A, Asuncion A (2010) UCI machine learning repository. <http://archive.ics.uci.edu/ml>
- Gutlein M, Frank E, Hall M, Karwath A (2009) Large-scale attribute selection using wrappers. In: *Proceeding. IEEE symposium on computational intelligence and data mining*, pp 332–339
- Hamdani TM, Won JM, Alimi AM, Karray F (2007) Multi-objective feature selection with NSGA II. *Int Conf Adapt Natural Comput Algorithms* 4431:240–247
- Hsu WH (2004) Genetic wrappers for feature selection in decision tree induction and variable ordering in bayesian network structure learning. *Inf Sci* 163(17):103–122
- Jia JH, Yang N, Zhang C, Yue AZ, Yang JY, Zhu DH (2013) Object-oriented feature selection of high spatial resolution images using an improved relief algorithm. *Math Comput Model* 58(3–4):619–626
- Jin X, Ma EWM, Cheng LL, Pecht M (2012) Health monitoring of cooling fans based on mahalanobis distance with mrmr feature selection. *IEEE Trans Instrum Meas* 61(8):2222–2229
- Jović A, Bogunović N (2015) A review of feature selection methods with applications. In: *International convention on information communication technology, electronics and microelectronics*. IEEE
- Karaboga D (2005) An idea based on honey bee swarm for numerical optimization. *Erciyes University, Kayseri*
- Kashef S, Nezamabadi-Pour H (2015) An advanced ACO algorithm for feature selection. *Neurocomputing* 147:271–279
- Kennedy J, Eberhard R (1997) A discrete binary version of the particle swarm algorithm. *Proc IEEE Int Conf Syst Man Cybern Comput Cybern Simul* 5:4104–4108
- Khushaba RN, Al-Ani A, Al-Jumaily A (2011) Feature subset selection using differential evolution and a statistical repair mechanism. *Expert Syst Appl* 38(9):11515–11526
- Lin SW, Lee ZJ, Chen SC, Tseng TY (2008a) Parameter determination of support vector machine and feature selection using simulated annealing approach. *Appl Soft Comput* 8(4):1505–1512
- Lin SW, Ying KC, Chen SC, Lee ZJ (2008b) Particle swarm optimization for parameter determination and feature selection of support vector machines. *Expert Syst Appl* 35(4):1817–1824
- McNabb A, Gardner M, Seppi K (2009) An exploration of topologies and communicational in large particle swarms. In: *Proceedings of the IEEE congress on evolutionary computation* IEEE Press, pp 712–719
- Niu B, Wang H, Wang J, Tan LJ (2013) Multi-objective bacterial foraging optimization. *Neurocomputing* 116:336–345
- Ozturk O, Aksac A, Elsheikh A, Ozyer T, Alhadj R (2013) A consistency-based feature selection method allied with linear SVMs for HIV-1 protease cleavage site prediction. *PLoS ONE* 8(8):e63145
- Park CH, Kim SB (2015) Sequential random k-nearest neighbor feature selection for high-dimensional data. *Expert Syst Appl* 42(5):2336–2342

- Passino KM (2002) Biomimicry of bacterial foraging for distributed optimization and control. *IEEE Control Syst* 22(3):52–67
- Peng HC, Long FH, Ding C (2005) Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans Pattern Anal Mach Intell* 27(8):1226–1238
- Wang H, Niu B (2017) A novel bacterial algorithm with randomness control for feature selection in classification. Elsevier, Amsterdam
- Wang HS, Yan XF (2015) Optimizing the echo state network with a binary particle swarm optimization algorithm. *Knowl Based Syst* 86:182–193
- Wang G, Ma J, Yang SL (2011) IGF-bagging: information gain based feature selection for bagging. *Int J Innov Comput Inf Control* 7(11):6247–6259
- Wang H, Jing X, Niu B (2017) A discrete bacterial algorithm for feature selection in classification of microarray gene expression cancer data. Elsevier, Amsterdam
- Xue B, Zhang M, Browne WN (2012) New fitness functions in binary particle swarm optimisation for feature selection. In: *Evolutionary computation (CEC). 2012 IEEE Congress*
- Xue B, Zhang M, Browne WN (2013) Particle swarm optimization for feature selection in classification: a multi-objective approach. *IEEE Trans Cybern* 43(6):1656–1671
- Xue B, Zhang M, Browne WN (2014) Particle swarm optimisation for feature selection in classification: novel initialisation and updating mechanisms. *Appl Soft Comput* 18:261–276
- Xue B, Zhang M, Browne W, Yao X (2016) A survey on evolutionary computation approaches to feature selection. *IEEE Trans Evol Comput* 20(4):606–626
- Yang CH, Chuang LY, Yang CH (2010) IG-GA: a hybrid filter/wrapper method for feature selection of microarray data. *J Med Biol Eng* 30(1):23–28
- Zhao Z, Liu H (2009) Searching for interacting features in subset selection. *IOS Press* 13(2):207–228
- Zhu Z, Ong YS, Markov DM (2007) Blanket-embedded genetic algorithm for gene selection. *Pattern Recognit* 40(11):3236–3248

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.