A Modified Firefly Algorithm with Support Vector Machine for Medical Data Classification

Brahim Sahmadi, Dalila Boughaci, Rekia Rahmani, Noura Sissani

A Modified Firefly Algorithm with Support Vector Machine for Medical Data Classification

Brahim Sahmadi[1], Dalila Boughaci[2], Rekia Rahmani[1] and Noura Sissani[1]

[1] LMP2M Laboratory, University Yahia Fares of Medea, Medea, Algeria
`sh_brahim@yahoo.fr, sahmadi.brahim@univ-medea.dz`
`rekia_rah@yahoo.fr`
`infores287@gmail.com`

[2] LRIA/Computer Sciences Department, University of Sciences and Technology Houari Boumediene (USTHB), Algiers, Algeria
`dalila_info@yahoo.fr, dboughaci@usthb.dz`

Abstract. Clinical information systems store a large amount of data in medical databases. In the use of medical dataset for diagnosis, the patient's information is selectively collected and interpreted based on previous knowledge for detecting the existence of disorders. Feature selection is important and necessary data pre-processing step in medical data classification process. In this work, we propose a wrapper method for feature subset selection based on a binary version of the Firefly Algorithm combined with the SVM classifier, which tries to reduce the initial size of medical data and to select a set of relevant features for enhance the classification accuracy of SVM. The proposed method is evaluated on some medical dataset and compared with some well-known classifiers. The computational experiments show that the proposed method with optimized SVM parameters provides competitive results and finds high quality solutions.

Keywords: Medical data classification, machine learning, feature selection, binary Firefly algorithm, support vector machine (SVM), cross-validation.

1   Introduction

Clinical information systems store a large amount of information in medical databases. So, the manual classification of this information is becoming more and more difficult. Therefore, there is an increasing interest in developing automated evaluation methods to follow up the diseases. Classification is one of the techniques of data mining which involves extracting a general rule or classification procedure from a set of learning examples. Medical data classification refers to learning classification models from medical datasets and aims to improve the quality of health care [3].

As medical datasets are generally characterized as having high dimensionality, and many of the feature attributes in a typical medical dataset are collected for reasons other than data classification. Some of the features are redundant while others are irrelevant adding more noise to the dataset, although in medical diagnosis, it is desirable to select

the clinical tests that have the least cost and risk and that are significantly important in determining the class of the disease [1].

Feature selection is important and necessary data pre-processing steps to increase the quality of the feature space. It aims to select a small subset of important (relevant) features from the original full feature set. It can potentially improve the performance of a learning algorithm significantly in terms of the accuracy; increase the learning speed, and simplifying the interpretation of the learnt models [2, 12]. Feature selection is used in different tasks of learning or data mining, in the fields of image processing, pattern recognition, data analysis in bioinformatics, categorization of texts, etc.

The methods used to evaluate a feature subset in the selection algorithms can be classified into three main approaches: filter methods, wrapper methods and embedded methods. Filter methods perform the evaluation independently of any classification algorithm; they are based on data and attributes [2]. Wrapper methods use the learning algorithm as an evaluation function. It therefore defines the relevance of the attributes through a prediction of the performance of the final system. Embedded methods combine the exploration process with a learning algorithm. The difference with wrapper methods is that the classifier not only serves to evaluate a candidate sub-set, but also to guide the selection mechanism.

Wrapper approaches conducts a search in the space of candidate subsets of features, and the quality of a candidate subset is evaluated by the performance of the classification algorithm trained on this subset [15]. Several wrapper methods have been proposed for feature selection, among them, the stochastic local search methods and the population based optimization metaheuristic methods, like the Genetic Algorithms (GA), the Memetic Algorithm (MA), Particle Swarm Optimization (PSO), and the Harmony Search Algorithm (HAS) [16].

In this work, we apply a wrapper method based on a binary version of Firefly Algorithm to the feature selection problem in medical data classification, in order to extract an ideally minimal subset of features with strong discriminative power. The proposed approach uses the SVM classifier for evaluating a feature subset.

This paper is organized as follows: first, we briefly outline the main idea of support vector machine methods (SVM) and the Binary Firefly Algorithm in Section 2. In Section 3, we describe the proposed approach for feature selection and classification of medical data. The experimental results are presented and discussed in Section 4. Finally, we conclude this study and discuss possible future work in Section 5.

2 Background

2.1 Support Vector Machines

Support Vector Machines (SVM) are a class of supervised learning algorithms introduced by Vladimir Vapnik [4]. The main principle of SVM is the construction of a function f called decision function that for an input vector x matches a value y, $y = f(x)$, where x is the example to classify and y is the class which corresponds to the example input. SVM are originally defined for binary classification problems, and their extension to nonlinear problems is offered introducing the kernel functions. SVM are

widely used in statistical learning and has proved effective in many application areas such as image processing, speech processing, bioinformatics, natural language processing, and even data sets of very large dimensions [13].

SVM classifiers are based on two key ideas: the notion of maximum margin and the concept of kernel function. The first key idea is the concept of maximum margin. We seek the hyperplane that separates the positive examples of negative examples, ensuring that the distance between the separation boundary and the nearest samples (margin) is maximal, they are called support vector. And as it seeks to maximize the margin, we will talk about wide margin separators [14].

The second key idea in SVM is the concept of kernel function. This is transforming the data space entries in a space of larger dimension called feature space in which it is likely that there is a dividing line, in order to deal with cases where the data are not linearly separable. Some examples of kernel functions are:

– Linear kernel: $K(x_i, x_j) = x_i . x_j$

– Polynomial kernel: $K(x_i, x_j) = (\gamma x_i . x_j + r)^d, \gamma > 0.$

– RBF kernel: $K(x_i, x_j) = e^{-\frac{|x_i - x_j|^2}{2\gamma^2}}, \gamma > 0.$

– Sigmoid kernel: $K(x_i, x_j) = tanh(\gamma x_i . x_j + r).$

Where γ, r and d are kernel parameters. In this study, we utilized the LIBSVM toolset and chose Radial Basis Function (RBF) as the kernel function, and its C and γ parameters are optimized using an iterative search method. Previous studies show that these two parameters play an important role on the success of SVMs [8].

2.2   Binary Firefly Algorithm

The Firefly algorithm is a recent bio-inspired metaheuristic developed by Xin She Yang in 2008 and it has become an important tool for solving the hardest optimization problems in almost all areas of optimization [6]. The algorithm is based on the principle of attraction between fireflies and simulates the behavior of a swarm of fireflies in nature, which gives it many similarities with other meta-heuristics based on the collective intelligence, such as the PSO (Particle Swarm Optimization) algorithm or the bee colony optimization algorithm. He uses the following three idealized rules:

- All fireflies are unisex, meaning that one firefly is attracted by another, regardless of sex.
- The attractiveness and brightness are proportional, so that for two flashing fireflies, the less bright will move towards the brighter. Attractiveness and brightness decrease with increasing distance. If there is not one firefly brighter than the other, they will place themselves randomly.
- The brightness of a firefly is determined by the point of view of the objective function to optimize. For a maximization problem, the luminosity is simply proportional to the value of the objective function.

Since the attractiveness is proportional to the luminosity of the adjacent fireflies, then the variation in the attractiveness β with the distance r is defined by:

$$\beta = \beta_0\, e^{-\gamma r2} \tag{1}$$

Where β_0 is the attractiveness at $r = 0$ and γ is the absorption coefficient.
The distance r_{ij} between 2 fireflies is determined by the formula (2).

$$r_{ij} = \|X_i - X_j\| = \sqrt{\sum_{k=1}^{d}\left(X_i^k - X_j^k\right)^2} \tag{2}$$

Where X_i^k is the kth component of the spatial coordinate of the ith firefly and d is the number of dimensions.

The movement of a firefly X_i to another firefly X_j more attractive is calculated by:

$$X_i^{t+1} = X_i^t + \beta_0 e^{-\gamma r_{ij}^2}\left(X_j^t - X_i^t\right) + \alpha\left(rand - \frac{1}{2}\right) \tag{3}$$

Where X_i^t and X_j^t are the current position of the fireflies X_i and X_j, and X_i^{t+1} is the ith firefly position of the next generation. The second term is due to attraction. The third term introduces randomization, with α being the randomization parameter.

The basic steps of the firefly algorithm can be formulated as the pseudo code shown in the Algorithm 1.

The original firefly algorithm is designed for optimization problems with continuous variables. Recently, several binary firefly algorithms were developed to solve discrete problems, such as scheduling, timetabling and combination. Compared with the original firefly algorithm, binary firefly algorithm obeyed similar fundamental principles while redefined distance, attractiveness, or movement of the firefly. In this study, we use a binary firefly algorithm for feature selection with new definitions of distance and movement of a firefly, similar to the approach used in [10].

3   The Proposed Method for Feature Selection and Classification

The feature selection task is a typical combination problem in essence, with the objective of selecting an optimal combination of features from a given feature space. Theoretically, for an n-dimensional feature space, there will be 2^n possible solutions (NP-hard problem). The use of metaheuristics methods as random selection algorithms, capable of effectively exploring large search spaces, which is usually required in case of feature selection. In this work, a binary firefly algorithm (BFA-SVM) is proposed; where the feature space is explored by a population of fireflies and the SVM classifier is used for evaluating a feature subset. The normalized Hamming distance was used to calculate attractiveness between a pair of fireflies and in order to increase the diversity of fireflies a dynamic mutation operator was introduced. The flowchart of BFA-SVM method is shown in Fig. 1.

3.1   Fireflies Representation and Initialization

To represent the subset of selected features, we chose a binary representation of a solution in the multidimensional search space. Every firefly x_i in the binary firefly algorithm represents a subset of the feature space (i.e. a possible solution for feature selection problem) as an N-dimensional binary array: bit values 1 and 0 represent a selected and unselected feature, respectively. The Initial population of fireflies is generated randomly; the bit positions for each firefly are randomly assigned as 1 or 0.

Algorithm 1. Pseudo code of the Firefly algorithm

Inputs : ***n*** *(number of fireflies),* ***max_iter*** *(maximum number of iterations),*
        α *(randomness parameter),* ***γ*** *(absorption coefficient).*

1. Define objective function *f(x), x = (x₁, ..., x_d)^T*;   // *d is positions dimension*
2. Initialize parameters *n, max_iter, α* and *γ*;
3. Initialize a population of fireflies *x_i (i = 1, 2, ..., n);*
4. Calculate the light intensity I_i (fitness) for each x_i by *f(x_i);*
5. Set t ← 0;
6. **while** (*t <max_iter*) **do**
    6.1. **for** i = 1 **to** n // all n fireflies
        for j = 1 **to** n // all n fireflies
          If ($I_j > I_i$)
          - Move firefly *i* towards *j* in all *d* dimensions;
          - Attractiveness varies with distance r_{ij} via $e^{-\gamma r_{ij}^2}$;
          - Evaluate new solutions and update light intensity;
          end if;
        end for j;
      end for i;
    6.2. Rank the fireflies and find the current best;
    6.3. Set t ← t+1;
    end while;
7. Return the global best firefly (i.e. the best solution);

3.2   Objective Function

The objective function of the BFA-SVM algorithm when searching for the optimal features subset is to maximize the accuracy rate in classifying the testing dataset. This is equivalent to an optimization problem seeking for a maximum solution. The classification rate *ACC* is calculated using cross-validation with 10-Folds [11]. This measure is calculated by the formula (4):

$$ACC = \left(\frac{Total\ correct}{L}\right) * 100 \tag{4}$$

Where *Total correct* is the number of examples correctly classified by the SVM classifier, and *L* is the total number of examples. The classification rate indicates whether the candidate subset permits good class discrimination.

3.3 The attractiveness of fireflies

For two fireflies X_i and X_j, the distance r_{ij} is defined based on the similarity ratio of the two fireflies using the normalized Hamming distance of the two position vectors as follow:

$$r_{ij} = 1 - \left(\sum_{k=1}^{d}(X_i^k \oplus X_j^k)\right)/d \qquad (5)$$

Where $\oplus$ denotes the XOR operation and d is the positions dimension. The attractiveness β between a pair of fireflies is calculated using the formula (1).

3.4 The movement of fireflies

The original firefly algorithm is designed for optimization problems with continuous variables. For the binarization of continuous metaheuristics, there are two main groups of binarization techniques. The first group of techniques allows working with the continuous metaheuristics without operator modifications and includes steps of binarization of the continuous solution after the original continuous iteration. The second group of techniques is called continuous-binary operator transformation; it redefines the algebra of the search space, thereby reformulating the operators [18]. In this work, we use a modification in the movement of a firefly by the reformulation of the formula (2). When a firefly X_i moves to another firefly X_j more attractive, every bit in its representation vector will make a decision to change its value or not. Changing a bit X_i^k in firefly X_i is done in two steps: the β-step (attraction) as indicated in the formula (6), which is regulated by the attractiveness β, and the α-step (mutation) were using the formula (7), which is controlled by a parameter α.

$$X_i^k = \begin{cases} X_j^k & if \ X_i^k \neq X_j^k \ and \ rand(0.1) < \beta \\ X_i^k & otherwise \end{cases} \qquad (6)$$

$$X_i^k = \begin{cases} 1 - X_j^k & if \ rand(0.1) < \alpha \\ X_i^k & otherwise \end{cases} \qquad (7)$$

β is the probability of a hetero-bit in the moving firefly changes to the corresponding bit in the brighter firefly (0→1 or 1→0). The parameter α regulates the random moving behavior (mutation) of a bit X_i^k, and it is calculated in each iteration of the BFA-SVM algorithm by the following formula :

$$\alpha = 1 - \frac{0.5 * iter}{max\_iter} \tag{8}$$

The mutation probability α is high in initial iterations, which makes BFA-SVM focus on exploration. As the number of iteration increases, the mutation probability will decrease, and BFA-SVM will accelerate its converging pace gradually.

4 Experiments

The proposed BFA-SVM algorithm was implemented on a PC with an Intel Core 2 Duo CPU 2.93 GHz, 4 GB of memory and the Windows 7 operating system. The programs are coded in Java language and we have used the LIBSVM package [5] as a library for the SVMs.

4.1 Dataset

To evaluate the performance of the proposed method, we have used 11 medical datasets obtained from the UCI Machine Learning Repository [17]. Table 1 describes the main characteristics of these datasets. The prediction process with the SVMs requires that the dataset must be normalized. The main advantages of such operation are to avoid attributes in greater numeric ranges dominating those in smaller numeric ranges, and to avoid numerical difficulties during the computation step. The range of each feature value is linearly scaled to the range [−1, +1] using the WEKA tools [7].

Table 1. The dataset description

Dataset	Number of features	Number of instances	Number of classes
Arrhythmia	279	452	16
Breast cancer	10	683	2
Colon cancer	2000	62	2
Dermatology	34	366	6
Diabetes	8	768	2
Heart-c	13	303	5
Heart-stat	13	270	2
Hepatitis	19	155	2
Liver-disorders	6	345	2
Lung cancer	56	32	3
Lymphography	18	148	4

4.2 Parameter Settings

The parameter values of the proposed algorithm are fixed by an experimental study. After a series of experiments, the different parameters are fixed empirically. The values of each parameter for the proposed method are given in Table 2.
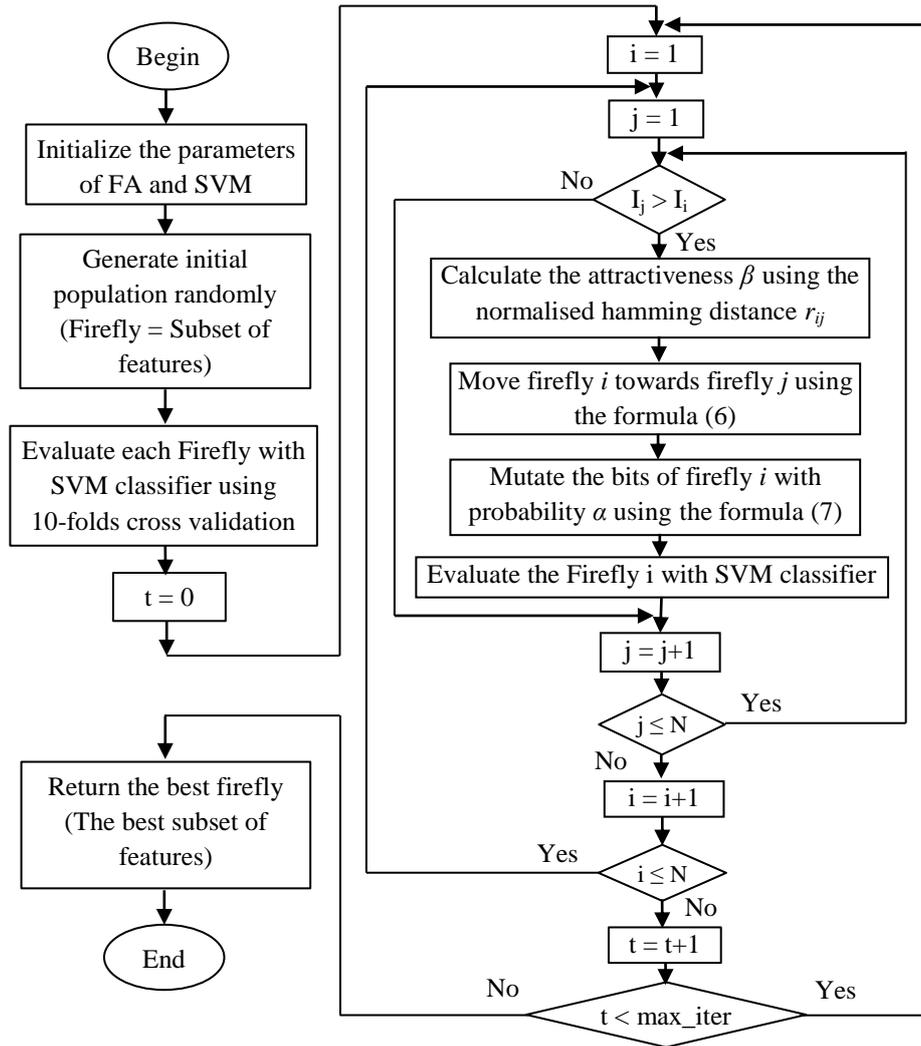

Fig.1. The flowchart of the proposed binary firefly algorithm for feature selection

Table 2. Parameters of BFA-SVM algorithm

Parameters	Values
Population size	30
Maximum number of generations	200
β_0	1.0
γ	1.0
Number of folds in cross-validation	10
Number of runs	20

4.3 Numerical Results

Due to the non-deterministic nature of the proposed method, several executions (20) were considered for each dataset. The minimum value, the maximum value, and the average of the accuracy rate of the classification for each dataset are reported. The best results are in bold font.

Table 3 gives a comparison between the results (mean of accuracy rate) obtained by the use of SVM classifier with default parameters for RBF kernel function, the use of SVM with optimized parameters given with a grid-search method, and the results obtained with by BFA-SVM with optimized parameters. The best results are obtained with BFA-SVM algorithm for all datasets, confirming that the feature selection and optimization of SVM parameters improves significantly the classification accuracy.

Table 3. Comparison between SVM $_{default}$, SVM $_{Grid\text{-}search}$ and BFA-SVM $_{Optimized}$

Dataset	SVM $_{default}$ Mean (%)	SVM $_{Optimized}$ Mean (%)	BFA-SVM $_{Optimized}$ Mean (%)
Arrhythmia	55.93	71.99	**75.84**
Breast cancer	97.01	97.12	**97.66**
Colon cancer	80.89	87.34	**91.94**
Dermatology	97.95	97.99	**99.36**
Diabetes	77.14	77.39	**79.23**
Heart-c	82.67	83.12	**85.07**
Heart-stat	82.44	83.94	**86.56**
Hepatitis	85.06	86.06	**92.46**
Liver-disorders	58.41	73.59	**75.70**
Lung cancer	73.91	77.50	**99.53**
Lymphography	83.04	84.56	**89.66**

Table 4. Comparison between BFA-SVM Optimized, HSA+SVM, MA+SVM and GA+SVM

Dataset	BFA-SVM Optimized				HSA+SVM				MA+SVM				GA+SVM			
	Mean (%)	Max (%)	Min (%)	Sd (%)	Mean (%)	Max (%)	Min (%)	Sd (%)	Mean (%)	Max (%)	Min (%)	Sd (%)	Mean (%)	Max (%)	Min (%)	Sd (%)
Arrhythmia	**75.84**	**76.77**	**74.78**	0.57	69.15	71.24	66.59	1.28	--	--	--	--	--	--	--	--
Breast cancer	**97.66**	**97.80**	**97.51**	0.05	97.44	97.65	97.22	0.10	97.44	**97.80**	97.49	0.20	97.20	97.36	96.63	0.14
Colon cancer	**91.94**	**91.94**	**91.94**	0.00	89.94	90.32	88.71	0.69	--	--	--	--	--	--	--	--
Dermatology	99.36	**100.00**	98.91	0.41	**100.00**	**100.00**	**100.00**	0.00	99.86	**100.00**	98.90	0.02	98.69	98.90	98.63	0.08
Diabetes	**79.23**	**79.56**	**78.78**	0.17	77.79	78.13	77.47	0.17	77.71	78.26	77.34	0.21	77.53	77.99	76.82	0.27
Heart-c	**85.07**	**85.81**	**84.16**	0.48	59.80	61.05	58.41	0.49	57.88	63.38	56.44	1.11	59.30	60.73	58.08	0.63
Heart-stat	**86.56**	**87.04**	**86.30**	0.27	84.64	85.56	83.70	0.57	84.56	86.30	83.70	0.57	84.11	85.18	83.70	0.35
Hepatitis	92.46	93.55	90.97	0.71	**99.94**	**100.00**	99.35	0.19	99.66	**100.00**	98.06	0.45	88.83	90.00	86.45	0.89
Liver-disorders	**75.70**	**76.23**	**75.36**	0.22	72.60	73.33	71.30	0.42	71.75	73.04	70.43	0.58	63.49	64.35	62.60	0.55
Lung cancer	99.53	**100.00**	96.88	1.14	98.75	**100.00**	93.75	2.26	**99.96**	**100.00**	**99.86**	0.18	88.22	90.62	87.50	0.62
Lymphography	**89.66**	**91.22**	**88.51**	0.93	49.00	55.41	43.92	3.09	48.73	54.70	46.62	1.82	50.83	52.02	49.32	0.66
Average	**88.46**	**89.08**	**87.65**	**0.45**	81.73	82.97	80.04	0.84	81.95	83.72	80.98	0.57	78.69	79.68	77.75	0.47

- The SVM with optimized parameters is used as classifier with the algorithm BFA-SVM
- -- Method did not use the dataset for test.
- The best results are in bold.

In order to evaluate the effectiveness of the proposed algorithm BFA-SVM, a comparison of the experimental results obtained by the method with the results of the works cited in [8, 9] is presented in the Table 4, where gives the average (Mean), the best (Max), the worst (Min) values of the classification accuracy, and the standard deviation (Sd) obtained by different methods. In [9], a hybrid search method based on both harmony search algorithm and stochastic local search, combined with a support vector machine (HAS+SVM) is given for feature selection in data classification. And the authors of [8] propose a genetic algorithm (GA) and memetic algorithm (MA) with SVM classifier for feature selection and classification.

As shown in Table 4, BFA-SVM algorithm succeeds in finding the best results for almost the checked datasets compared to HAS+SVM, MA+SVM and GA+SVM methods in term of classification accuracy point of view (in 8 datasets among 11, the BFA-SVM algorithm gives the best classification rate average and for the max value of the classification accuracy is reached in 10 datasets among 11). The small standard deviations of the classification accuracies presented show the consistency of the proposed algorithm. This proves the ability of the proposed algorithm as a good classifier in medical data diagnosis.

5   Conclusion

Health care systems generates vast amount of information and it is accumulated in medical databases, and the manual classification of this data becoming more and more difficult. Therefore, there is an increasing interest in developing automated methods for medical data analysis. In this work we proposed a wrapper method for feature selection and classification of medical dataset based on a Binary firefly algorithm combined with the SVM classifier. The results obtained from tests carried out on several public medical dataset indicate that the proposed BFA-SVM method is competitive with other meta-heuristics (Genetic algorithm, Memetic algorithm and Harmony search algorithm) for the feature selection, and experiments have shown us that the method greatly improves the learning quality and ensures the stability of the generated prediction model. It also reduces the size of the representation space by eliminating noise and redundancy.

As a continuation of this work, it would be desirable to work on the reduction of the computation time by proposing a parallel implementation of the proposed method. It is also possible to use the binary firefly algorithm with other classification algorithms such as neural networks and Naïve Bayes.

12

References

1. Almuhaideb S., El-Bachir Menai M.: Hybrid metaheuristics for medical data classification. In: El-Ghazali T, ed. Hybrid Metaheuristics, Springer, 187–217, (2013).
2. Liu H., Yu L.: Toward integrating feature selection algorithms for classification and clustering, IEEE Trans. Knowl. Data Eng., vol. 17, no. 4, pp. 491–502, (2005).
3. Almuhaideb S., El-Bachir Menai M.: Impact of preprocessing on medical data classification. Front. Comput. Sci.10(6), 1082–1102, Doi: 10.1007/s11704-016-5203-5Aydin, (2016)
4. Vapnik, V.: The Natural of Statistical Learning theory. Springer, New York, (1995).
5. Chang C.C., Lin C.J.: LIBSVM: a library for support vector machines. ACM Transactions on Intelligent Systems and Technology, Vol. 2, No. 3, Article 27, (2011). http://www.csie.ntu.edu.tw/~cjlin/libsvm/, Accessed in October 11, 2017.
6. Yang X. S.: Nature-Inspired Metaheuristic Algorithms. Luniver Press, UK, (2008).
7. Eibe F., Mark A. H., and Witten I. H.: The WEKA Workbench. Online Appendix for Data Mining: Practical Machine Learning Tools and Techniques, Morgan Kaufmann, Fourth Edition, (2016).
8. Nekkaa M., Boughaci D.: A memetic algorithm with support vector machine for feature selection and classification. Memetic Comp. 7, 59-73, Doi: 10.1007/s12293-015-0153-2, (2015).
9. Nekkaa M., Boughaci D.: Hybrid Harmony Search Combined with Stochastic Local Search for Feature Selection. Neural Process Lett. Doi: 10.1007/s11063-015-9450-5, (2015).
10. Zhang J., Gao B., Chai H., Ma Z., Yang G.: Identification of DNA-binding proteins using multi-features fusion and binary firefly optimization algorithm. BMC Bioinformatics 17:323, Doi: 10.1186/s12859-016-1201-8, (2016)
11. Han J., Kamber M.: Data mining concepts and techniques, 2nd edn. Morgan Kaufmann, San Francisco, (2006).
12. Bonilla Huerta E. B., Duval B., Hao J. K.: A hybrid GA/SVM approach for gene selection and classification of microarray data. In: Rothlanf et al (eds) EvoWorkshops 2006, LNCS 3907, pp 34–44, (2006).
13. Kecman V.: Learning and soft computing: support vector machines. In Neural networks and fuzzy logic models. The MIT press, London, (2001).
14. Burgers C. J. C.: A tutorial on support vector machines for pattern recognition, Data Mining Knowledge Discov. 2, 121–167, (1998).
15. Duval B., Hao J. K.: Advances in metaheuristics for gene selection and classification of microarray data, Briefings in bioinformatics, Vol. 11. 1, pp. 127-141, (2009).
16. El Aboudi N., Benhlima l.: Review on Wrapper Feature Selection Approaches, International Conference on Engineering & MIS (ICEMIS), pp. 1-5, (2016).
17. Lichman, M.: UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science, (2013).
18. Crawford B., Soto R., Astorga G., García J., Castro C., Paredes F.: Putting Continuous Metaheuristics to Work in Binary Search Spaces. Complexity, vol. 2017:19, Doi:10.1155/2017/8404231, (2017).