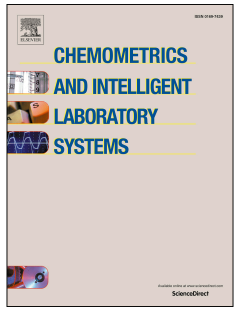
Accepted Manuscript



Feature selection using particle swarm optimization-based logistic regression model

Omar Saber Qasim, Zakariya Yahya Algamal

PII: S0169-7439(18)30380-0

DOI: 10.1016/j.chemolab.2018.08.016

Reference: CHEMOM 3676

To appear in: *Chemometrics and Intelligent Laboratory Systems*

Received Date: 6 July 2018

Revised Date: 24 August 2018

Accepted Date: 31 August 2018

Feature selection using particle swarm optimization-based logistic regression model

Omar Saber Qasim

Department of Mathematics, University of Mosul, Mosul, Iraq

E-mail: omar.saber@uomosul.edu.iq

Zakariya Yahya Algamal *

Department of Statistics and Informatics, University of Mosul, Mosul, Iraq

E-mail: zakariya.algamal@uomosul.edu.iq

ORCID: 0000-0002-0229-7958

*Corresponding Author

Zakariya Yahya Algamal

E-mail: zakariya.algamal@uomosul.edu.iq

Telephone number: +964 7701640834

Abstract

In any classification problem, the dataset typically has a large number of features. However, not all features are necessary to obtain a good classification performance because some of them are irrelevant and redundant. Therefore, classifiers with less number of features but with better classification accuracy are favored for ease of interpretation. In this work, particle swarm optimization algorithm along with logistic regression model is proposed. Additionally, the Bayesian information criterion (BIC) as a fitness function is proposed. The performance of different fitness functions is investigated and compared with BIC. The performance of the proposed method is evaluated based on a large number of different types of datasets. Experimental results using different types of datasets demonstrate the usefulness of our proposed method in significantly obtaining an improved classification performance with few features. Further, the results show that the proposed methods have a competitive performance comparing with other existing fitness functions.

Keywords: Feature selection; particle swarm optimization; logistic regression; Bayesian information criterion; classification.

1. Introduction

Classification is an important task in data mining and machine learning, which aims to classify each case in the data set into different categories based on the information described by its features [1]. In many problems of classification, the dataset contains a large number of features which causes the problem known as 'miss classification', which occurs as a result of having a large number of irrelevant features and causes a major hitch in classification [2,3].

Feature selection is a fundamental and widely used technique to transact with the large data size. Feature selection can be selecting only the relevant features for classification by reducing irrelevant features and improve the classification performance [4]. Also, feature selection in classification task can be substantive as follows: given the main set F consisting of m available features, find a feature subset S consisting of n relevant features, where $n < m$ and $S \subset F$ without replacement. The number of features selection reduces through eliminating irrelevant features, and thus fulfills in enhanced efficiency and increased accuracy [5].

Different algorithms in recent years, have been proposed to get semi-optimal subsets of solutions. These algorithms include particle swarm optimization (PSO), genetic algorithms (GA), ant colony optimization (ACO) and others, which designed by emulate the natural evolution. Particle swarm optimization (PSO) is a one of the methods of comparatively recent evolutionary algorithms (EA) which is based on the swarm intelligence. PSO is less expensive and can converge more fast compared with other EA [5]. PSO has been used in several fields, including feature selection. Each particle in PSO has a position and moves based on an updated velocity. The parameters of particles (pbest and gbest) are updated at each iteration. Fitness function is calculated for each particle to find the best solution in the search space. This solution value is stored and represents the pbest (position). The value gbest is a global optimal value for the whole population [6,7].

To extend the algorithm of PSO to a binary or discrete space, Kennedy and Eberhart submitted a binary PSO (BPSO) algorithm [8]. In BPSO, a particle moves to near corners of a hypercube by flipping different numbers of bits; therefore, the particle velocity may be described by the number of bits changed for each iteration. The particle adjusts its position during movement by changing its velocity based on its experiment, as well as the experiment of its neighboring particles, until reached to optimal position of itself and its neighbor [9,10].

In this work, particle swarm optimization algorithm along with logistic regression model is proposed. Additionally, the Bayesian information criterion (BIC) as a fitness function is proposed. The performance of different fitness functions are investigated and compared with BIC. The performance of the proposed method is evaluated based on large number of different types of datasets.

The rest of this paper is arranged as follows: Section 2 displays the concept of feature selection, logistic regression, and the PSO algorithm. While Section 3 contains the details of the proposed method. Section 4 covers the real data application results. Finally, the conclusion is covered by Section 5.

2. Preliminaries

2.1. Feature selection

Feature selection method is a procedure that reduces or minimizes the number of features and selects some subsets of original features. Feature selection method is often used in preprocessing to determine relevant attributes that are often unknown prior and eliminate irrelevant or excrescent features which do not have importance in classification.

Feature selection can be applied in several areas, most notably which aims for better classification accuracy and mostly fall into three classes: filter, wrapper, and embedded methods.

2.2. Logistic regression model

Logistic regression model (LRM) is considered as a statistical method to model a binary target variable, such as cancer classification problem in which the target variable only has two values: 1 for the tumor class and 0 for the normal class [11]. In LRM, the regression equation has a nonlinear relation with the linear combination of the features.

Features matrix can be described mathematically as a matrix $\mathbf{F} = (f_{ij})_{n \times d}$, where each column represents a feature and each row represents an instance (observation). The numerical value of f_{ij} denotes the measurement of a specific feature $j\ (j = 1,...,d)$ in a specific instance $i\ (i = 1,...,n)$. Given a training dataset $\{(\mathbf{f}_i, y_i)\}_{i=1}^{n}$, where $\mathbf{f}_i = (f_{\mathrm{i,1}}, f_{\mathrm{i,2}}, ..., f_{\mathrm{i,d}})$ represents a vector of the i^{th} feature values, and $y_i \in \{0,1\}$ for $i = 1,...,n$, where $y_i = 1$ specifies the i^{th} sample is in class 1 and $y_i = 0$ specifies the i^{th} sample is in class 2. In general, the aim is to classify the new instance and identify the relevant features with high classification accuracy. Assume $p(\mathbf{f}_i)$ represents the class-conditional probability for instance i when $y_i = 1$, $p(\mathbf{f}_i) = \Pr(y_i = 1 | \mathbf{f}_i)$, then the LRM is

$$\ln\left[\frac{p(\mathbf{f}_i)}{1 - p(\mathbf{f}_i)}\right] = \alpha_0 + \mathbf{f}_i^T \boldsymbol{\alpha},\ i = 1, 2, ..., n, \tag{1}$$

where $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_d)^T \in \mathbf{R}^d$ is a vector of unknown feature coefficients. The negative log-likelihood function of Eq. (1) is

$$\begin{aligned}\ell(\boldsymbol{\alpha}) &= -\sum_{i=1}^{n}\left[y_i \ln(p(\mathbf{f}_i)) + (1-y_i)\ln(1-p(\mathbf{f}_i))\right] \\ &= -\sum_{i=1}^{n}\left[y_i (\alpha_0 + \mathbf{f}_i^T \boldsymbol{\alpha}) - \ln(1+\exp(\alpha_0 + \mathbf{f}_i^T \boldsymbol{\alpha}))\right]\end{aligned} \tag{2}$$

Minimizing Eq. (2) produces the maximum likelihood estimator (MLE) of $\boldsymbol{\alpha}$. The advantage of LRM is that it offers estimating the probabilities $p(\mathbf{f}_i)$ and $1-p(\mathbf{f}_i)$ for each class and classifying instances simultaneously. The classification rule can be designated as the instance belongs to class 1 if $\Pr(y_i = 1|\mathbf{f}_i) \geq 0.5$ and it belongs to class 2 if $\Pr(y_i = 1|\mathbf{f}_i) < 0.5$ [12].

2.3. Particle swarm optimization

The practical swarm optimization (PSO) algorithm is a population based global search algorithm and a substitution solution. The basic idea of PSO was primarily inspired by emulation of the public behavior of animals such as fish schooling, bird flocking [13,14]. In general, PSO is a simple technique requiring minimal parameters but powerful which generally reaches best solutions of the problem efficiently.

The PSO was made as another option to standard evolutionary strategies, like genetic algorithm (GA), but does not use the main evolutionary components as selection, crossover and mutation. PSO applied successfully in variety of problems as search and optimization method [15].

The initial population of PSO is a swarm of particles start with randomly generated and each particle has two properties of velocity and position for a search problem, where $X_i = \{x_{i1}, x_{i2}, ...x_{im}\}$, $i = 1, 2, ...n$ represent the position vector, $V_i = \{v_{i1}, v_{i2}, ...v_{im}\}$ $i = 1, 2, ...n$ represent the velocity vector and

$Pbes_i^t = \{p_{i1}, p_{i2}, ...p_{im}\} \quad i = 1, 2, ...n$ represent the personal best position attained for the i_{th} particle in a swarm with m particles.

Each particle influenced by its own velocity, the best position achieves by Pbest (Personal best), and the best position achieve by the whole swarm Gbest (global best), where $Gbest = \{g_1, g_2, ...g_m\}$. In the search space, the movement of particles is governed by the position and velocity updating, where the velocity updating equation as given below [16]:

$$v_{ij}^{t+1} = wv_{ij}^t + k_1 r_1 (p_{ij}^t - x_{ij}^t) + k_2 r_2 (g_j^t - x_{ij}^t) \tag{3}$$

Also, the position will be calculated as follows:

$$x_{ij}^{t+1} = x_{ij}^t + v_{ij}^{t+1} \tag{4}$$

where t represents the number of iterations , w is the inertia weight ranging between 0 and 1, k_1 and k_2 represent the cognitive memory and social learning factors, respectively; and r_1 and r_2 are random numbers uniformly selected between 0 and 1. The particles in the swarm are update of the positions and velocities during each iteration, and the Pbest (new personal best) and Gbest (global best) values and corresponding particles are identified.

The PSO algorithm original was introduced for continuous optimization. Kennedy and Eberhart proposed a binary PSO (BPSO), which can be used as search space in the discrete problems. In BPSO, the position of each particle is encoded by a binary string are restricted to 0 or 1 and the velocity represents the probability of an element which has a value 1. Where it is possible to introduce the sigmoid function to transform velocity into the range of 0 and 1. The position of the particle in BPSO updates according to the following formulae [17]:

$$x_{ij}^{t+1} = \begin{cases} 1 & if \quad rand() < f\,(v_{ij}^{t+1}) \\ 0 & if \quad o.w \end{cases} \tag{5}$$

$$f(v_{ij}^{t+1}) = \frac{1}{1+e^{-v_{ij}^{t+1}}} \tag{6}$$

where $rand()$ represents a random number chosen from a uniform distribution in [0,1] and x_{ij}^{t} is transformed to [0,1] by a sigmoid function. Formulas (5) and (6) normalizes velocities into range [0, 1] and obtain the solution from the binary vector of particle positions by choosing the set of features with position set to 1. A pseudo code for the BPSO can be written as follows:

```
Begin
    Create an initial position X(0) and velocities V(0) for each particle
    Set iteration t=0
  Repeat
        Compute fitness function for each individual of swarm
        Begin (perform BPSO operation)
            Compute V(t+1) from Eq. (3)
            Compute X(t+1) from Eq. (5)
        END
        Set t = t + 1
  Until termination criteria satisfied
END
Return the best feature subset found by the swarm
```

3. The proposed algorithm

The proposed algorithm, PSO-LRBIC, is introduced where there are two ideas behind our proposed algorithm. The first idea is to use logistic regression model as a classifier and to use PSO to search for a best set of features. The second idea is to employ the BIC as a measure of fitness quality to find the most relevant features in PSO search.

Several studies used the following two fitness functions [18-21]

$$\text{Fit1} = \delta \times \text{CA} + (1-\delta) \times \left(\frac{p-q}{p} \right), \quad (7)$$

and

$$\text{Fit2} = \delta \times \text{CA} + (1-\delta) \times \left(\frac{1}{q} \right), \quad (8)$$

where CA is the classification accuracy, which is obtained by applying the classifier, p represents the number of features in the dataset, q represents the number of the selected features, and $\delta \in [0,1]$ is the relative quality between the CA and q. Usually, δ is setting to be 0.9. Related to our proposed algorithm, PSO-LRBIC, the classification accuracy is defined as the correctly classified classes using the logistic regression model as a classifiers. While, the number of the selected features, q, is obtained using PSO. For each particle, the fitness functions in Eq. (7) and Eq. (8) is computed and their maximum value is compared with the global best fitness. If the current fitness value is better, then the global best fitness is replaced with it.

Although Eq. (7) and Eq. (8) are frequently used, their advantages could be decreased. This is because both fitness functions depend on the value of δ. Changing δ will change their performances. On the other hand, using classification accuracy when the datasets are imbalanced is not preferred [22]. Consequently, the Bayesian information criterion as a fitness function is proposed. It is defined as

$$\text{Fit-BIC} = -2 \times \log \text{L}(\theta) + q \times \log(n), \quad (9)$$

where $\log \text{L}(\theta)$ is the log-likelihood function which is given by Eq. (2). It is noted that Fit-BIC integrates the loss function with the number of selected features. The global best fitness can be obtained by minimizing Eq. (9).

The pseudo-code of the PSO-LRBIC algorithm is described in Algorithm 1.

Algorithm 1 Pseudo-Codes of PSO-LRBIC algorithm

1: Set the initial parameters of the algorithm: m , maxiter , k_1, k_2, w .

2: Initialize the positions and velocities from uniform distribution within range [0,1] and [0,4], respectively.

3: Evaluate each particle according to $\text{Fit-BIC} = -2 \times \log \text{L}(\theta) + q \times \log(n)$ and assign the best global particle (Gbest) with $\min(\text{Fit-BIC})$.

4: Set the iteration from 1 to maxiter .

5: Update the particle velocity and position according to Eqs. (3) and (5).

6: Stope when $\text{i} \le \text{maxiter}$ satisfied return Obtain the best global particle.

7: Return the best selected features.

4. Experimental setting

In this section, the proposed algorithm, PSO-LRBIC, is evaluated and its usefulness is compared with PSO-LRFIT1 (PSO depending on Eq. (7) as a fitness function) and PSO-LRFIT2 (PSO depending on Eq. (8) as a fitness function) methods.

4.1. Datasets

Four binary class datasets: HDAC8 inhibitory activity (IC50) [23], antimicrobial agents (pMIC) [24], Human intestinal absorption (HIA) [25], and P-glycoproteins(P-gp) of the anti-cancer drugs [25], which were obtained from the several studies, were used. Table 1 summarizes a brief description of the used datasets. These datasets are varied in terms of number of observations and number of features. Dragon software (v. 6.0) was used to generate the descriptors for all the used datasets, where each molecular structure of the used dataset compound was sketched using Chem3D software and was optimized using the molecular mechanics (MM2).

Each used dataset was divided into training and testing datasets. A training dataset of 70% of total number of observations was taken to evaluate the internal classification, while the test dataset consisting of 30% of observations was used to evaluate the external classification of the used methods.

4.2. Evaluation criteria

To evaluate the classification performance of the proposed method and the other two used methods, three classification performance criteria are employed: (1) the classification accuracy (CA), (2) the geometric mean of sensitivity and specificity (G□mean), and (3) the area under the curve (AUC). The CA stands for the proportion of correctly classified positive and negative class, which measures the classification power of the classifier. The CA can be described as

$$\mathrm{CA} = \frac{\mathrm{TP+TN}}{\mathrm{TP+FP+FN+TN}} \times 100\%, \tag{10}$$

where TP is the number of true positive, FP is the number of false positive, TN is the number of true negative, and FN is the number of false negative.

A typical classifier should maximize the accuracy on the both of positive and negative classes. Accordingly, the G□mean has been used as a criterion to highlight the joint performance of sensitivity and specificity. It is defined as

$$\text{G-mean} = \sqrt{\text{Sensitivity} \times \text{Specificity}}, \quad (11)$$

where sensitivity is the fraction of positive classes that were successfully classified, and specificity is the fraction of negative classes that were properly classified. The AUC was used to quantitatively assess the overall classification performance of a classifier. Its value can vary from 0 to 1, the closer the value to 1, the better overall classification performance is.

4.3. PSO parameters setup

In all of the used datasets, the parameter configurations for the PSO algorithm are presented as follows. The number of particles, np , is set to 30 and the number of iterations, maxiter , is set to 100. The acceleration coefficients k_1 and k_2 are set with a minimum value equals to 1.5 and with a maximum value equals to 4. The acceleration coefficients are updating during the iteration according to the following equations:

$$k_1 = k_{1,\min} + \frac{t}{\text{maxiter}}(k_{1,\max} - k_{1,\min}), \quad (12)$$

$$k_2 = k_{2,\min} + \frac{t}{\text{maxiter}}(k_{2,\max} - k_{2,\min}). \quad (13)$$

In addition, the minimum and the maximum values for the inertial weight, $w_{\min}$ and $w_{\max}$, are set to 0.1 and 0.9, respectively. The inertial weight is updating according to the following equation:

$$w = w_{\max} - \frac{t}{\text{maxiter}}(w_{\max} - w_{\min}). \quad (14)$$

5. Results

5.1. Classification results

Each method is independently run 20 times on each dataset, and the average of each evaluation criteria is determined based on all these runs. Table 2 reports the experimental results of the used methods in terms of the CA and G-mean for both the training and testing datasets, and the number of selected features (# selected features). In this Table, the best average results obtained with respect to each criterion are shown in bold font. As it can be seen from Table 2, one can summarize the following points:

1- From the results of Table 2, we observed that PSO-LRBIC gives highly competitive results compared with PSO-LRFIT1 and PSO-LRFIT2 in all datasets. The most remarkable result for PSO-LRBIC concerns the HIA dataset. PSO-LRBIC obtains 84.07%, 80.91%, and 0.821, classification accuracy of training and testing datasets, and G-mean, respectively, with average 4.75 selected features. While the PSO-LRFIT1 and PSO-LRFIT2 reach a classification rate no greater than 75.40% of the training dataset and 72.30% of the testing dataset with at least 6.7 features.
2- Depending on the training dataset, the results of the PSO-LRBIC indicate that the proposed method is able to get better classification accuracy results with a considerable reduction in the number of original features. For example, for the P-gp dataset, while the PSO-LRFIT1 and PSO-LRFIT2, respectively, yield 90.07 ± 0.014 and 89.86 ± 0.017 classification accuracy with 24.2 ± 1.771 and 24.8 ± 1.771 features, the PSO-LRBIC yields 95.69 ± 0.006 classification accuracy with 12 ± 1.214 features.
3- Although, PSO-LRFIT1 achieved higher average classification accuracy (75.37%) for the training dataset and (72.13%) for the testing dataset on HIA dataset with 6.67 averaged selected features comparing with PSO-LRFIT2, the proposed PSO-LRFIT1 achieved close average classification accuracy compared with PSO-LRFIT2 on the rest of the used datasets.
4- In terms of G-mean criteria, PSO-LRBIC signals that it has a significant balance of classification performance between positive and negative cases comparing with PSO-LRFIT1 and PSO-LRFIT2. The PSO-LRBIC achieves the highest performance as its

G□mean is much higher than that of PSO-LRFIT1 and PSO-LRFIT2. Related to Antimicrobial dataset, for instance, the G-mean obtained by PSO-LRBIC is 0.960 which is better than those obtained by PSO-LRFIT1 at 0.909 and PSO-LRFIT2 at 0.897.

5- From all the ten datasets, the PSO-LRBIC outperformed the PSO-LRFIT1 and PSO-LRFIT2 in terms of classification accuracy of the testing dataset, where PSO-LRBIC obtains the highest classification accuracy. Further, it can be seen that the PSO-LRFIT1 is the second best method.

Overall, the results yielded by PSO-LRBIC are superior compared with the other two used competitor methods on all datasets in terms of classification performance and selected features. These results reveal that the PSO-LRBIC not only achieves the high classification performance but also can reduce the number of features.

5.2. Statistical test

For further convincing conclusions regarding our proposed method, PSO-LRBIC, in selecting the most related features with high classification accuracy, a Friedman test, which is a non-parametric test, was utilized. The Friedman test was performed depending on the AUC values of the training dataset. When the null hypothesis is rejected, post hoc of Bonferroni test was computed under different critical values (0.01, 0.05, and 0.1). Table 3 listed the statistical test results.

Based on the achieved results, the null hypothesis is rejected at $\alpha = 0.05$. As a result, there are statistical significance between PSO-LRBIC, PSO-LRFIT1, and PSO-LRFIT2 over the ten used datasets depending on the AUC criterion. In addition to that, the PSO-LRBIC has the lowest average rank with 2.452 comparing with PSO-LRFIT1 and PSO-LRFIT2. Depending on Bonferroni test results, it appears that the average ranks of PSO-LRFIT1 and

PSO-LRFIT2 are higher than $\alpha_{0.05}$, $\alpha_{0.01}$, and $\alpha_{0.10}$. These results suggesting that both PSO-LRFIT1 and PSO-LRFIT1 are significantly worse than PSO-LRBIC over the four used datasets. Moreover, it is clearly seen that the difference between the averaged AUC of PSO-LRFIT1 and PSO-LRFIT2 is not significant at $\alpha = 0.05$.

5.3. Stability test

To further highlight the performance of the PSO-LRBIC, a stability test using the Jaccard index was obtained. A stability test is an indicator of feature selection consistency. The Jaccard index can be defined as the size of the intersection between any two groups divided by the size of their union. Let G_1 and G_2 be subsets of the selected features such that $G_1, G_2 \subseteq G$, then the Jaccard index is defined as follows:

$$I_J(G_1, G_2) = \frac{|G_1 \cap G_2|}{|G_1 \cup G_2|}. \tag{15}$$

For a number of solutions $G = \{G_1, ..., G_r\}$, the stability test value among these solutions is defined as

$$Stability\ \ test = \frac{2}{r(r-1)} \sum_{i=1}^{r-1} \sum_{j=i+1}^{r} I_J(G_i, G_j). \tag{16}$$

The higher the stability test value is, the more stable the feature selection is. Accordingly, the selected features are more reliable and the classification accuracy is stronger. Figure 1 depicts the stability test values on four datasets for the used methods. As seen in Figure 1, the PSO-LRBIC displays the high rate of stability compared with PSO-LRFIT1 and PSO-LRFIT2.

6. Conclusion

In this paper, a particle swarm optimization algorithm along with logistic regression model is proposed and implemented for feature selection and classification. This proposed method, PSO-LRBIC, makes use of advantages of both PSO and the logistic regression with BIC as a fitness function and the experimental results show the promising behavior of the PSO-LRBIC. The PSO-LRBIC has resulted in reduced size of the feature number and increased classification accuracies. PSO-LRBIC has resulted in classification performance comparable to other two competitor method.

References

1. Hameed SS, Hassan R, Muhammad FF. Selection and classification of gene expression in autism disorder: Use of a combination of statistical filters and a GBPSO-SVM algorithm. *PloS one*. 2017; **12**: e0187371.
2. Jirapech-Umpai T, Aitken S. Feature selection and classification for microarray data analysis: evolutionary methods for identifying predictive genes. *BMC Bioinformatics*. 2005; **6**: 148.
3. Sugumaran V, Muralidharan V, Ramachandran KI. Feature selection using Decision Tree and classification through Proximal Support Vector Machine for fault diagnostics of roller bearing. *Mechanical Systems and Signal Processing*. 2007; **21**: 930-942.
4. Hua J, Tembe WD, Dougherty ER. Performance of feature-selection methods in the classification of high-dimension data. *Pattern Recognition*. 2009; **42**: 409-424.
5. Unler A, Murat A. A discrete particle swarm optimization method for feature selection in binary classification problems. *European Journal of Operational Research*. 2010; **206**: 528-539.
6. Kiran MS. Particle swarm optimization with a new update mechanism. *Applied Soft Computing*. 2017; **60**: 670-678.
7. Zouache D, Nouioua F, Moussaoui A. Quantum-inspired firefly algorithm with particle swarm optimization for discrete optimization problems. *Soft Computing*. 2015; **20**: 2781-2799.
8. Eberhart R, Kennedy J. A new optimizer using particle swarm theory. 1995. IEEE. p 39-43.
9. Chuang LY, Chang HW, Tu CJ, Yang CH. Improved binary PSO for feature selection using gene expression data. *Computational biology and chemistry*. 2008; **32**: 29-37.
10. Chuang L-Y, Yang C-H, Li J-C. Chaotic maps based on binary particle swarm optimization for feature selection. *Applied Soft Computing*. 2011; **11**: 239-248.
11. Algamal ZY, Lee MH. Regularized logistic regression with adjusted adaptive elastic net for gene selection in high dimensional cancer classification. *Comput Biol Med*. 2015; **67**: 136-145.
12. Algamal ZY, Lee MH. Penalized logistic regression with the adaptive LASSO for gene selection in high-dimensional cancer classification. *Expert Systems with Applications*. 2015; **42**: 9326-9332.

13. Chen K-H, Wang K-J, Tsai M-L, Wang K-M, Adrian AM, Cheng W-C, Yang T-S, Teng N-C, Tan K-P, Chang K-S. Gene selection for cancer identification: a decision tree model empowered by particle swarm optimization algorithm. *BMC Bioinformatics*. 2014; **15**: 49.
14. Lu Y, Wang S, Li S, Zhou C. Particle swarm optimizer for variable weighting in clustering high-dimensional data. *Machine Learning*. 2009; **82**: 43-70.
15. Chen K-H, Wang K-J, Wang K-M, Angelia M-A. Applying particle swarm optimization-based decision tree classifier for cancer classification on gene expression data. *Applied Soft Computing*. 2014; **24**: 773-780.
16. Cervantes J, Garcia-Lamont F, Rodriguez L, López A, Castilla JR, Trueba A. PSO-based method for SVM classification on skewed data sets. *Neurocomputing*. 2017; **228**: 187-197.
17. Vieira SM, Mendonça LF, Farinha GJ, Sousa JMC. Modified binary PSO for feature selection using SVM applied to mortality prediction of septic patients. *Applied Soft Computing*. 2013; **13**: 3494-3504.
18. Aziz MAE, Hassanien AE. Modified cuckoo search algorithm with rough sets for feature selection. *Neural Computing and Applications*. 2018; **29**: 925-934.
19. Aladeemy M, Tutun S, Khasawneh MT. A new hybrid approach for feature selection and support vector machine model selection based on self-adaptive cohort intelligence. *Expert Systems with Applications*. 2017; **88**: 118-131.
20. Lai C-M, Yeh W-C, Chang C-Y. Gene selection using information gain and improved simplified swarm optimization. *Neurocomputing*. 2016; **218**: 331-338.
21. Sayed GI, Hassanien AE, Azar AT. Feature selection via a novel chaotic crow search algorithm. *Neural Computing and Applications*. 2017.
22. Moayedikia A, Ong K-L, Boo YL, Yeoh WGS, Jensen R. Feature selection for high dimensional imbalanced class data using harmony search. *Engineering Applications of Artificial Intelligence*. 2017; **57**: 38-49.
23. Cao GP, Arooj M, Thangapandian S, Park C, Arulalapperumal V, Kim Y, Kwon YJ, Kim HH, Suh JK, Lee KW. A lazy learning-based QSAR classification study for screening potential histone deacetylase 8 (HDAC8) inhibitors. *SAR QSAR Environ Res*. 2015; **26**: 397-420.
24. Xing J-J, Liu Y-F, Li Y-Q, Gong H, Zhou Y-P. QSAR classification model for diverse series of antimicrobial agents using classification tree configured by modified particle swarm optimization. *Chemometrics and Intelligent Laboratory Systems*. 2014; **137**: 82-90.
25. Liu F, Zhou Z. A new data classification method based on chaotic particle swarm optimization and least square-support vector machine. *Chemometrics and Intelligent Laboratory Systems*. 2015; **147**: 147-156.

Table 1: Description of the used datasets

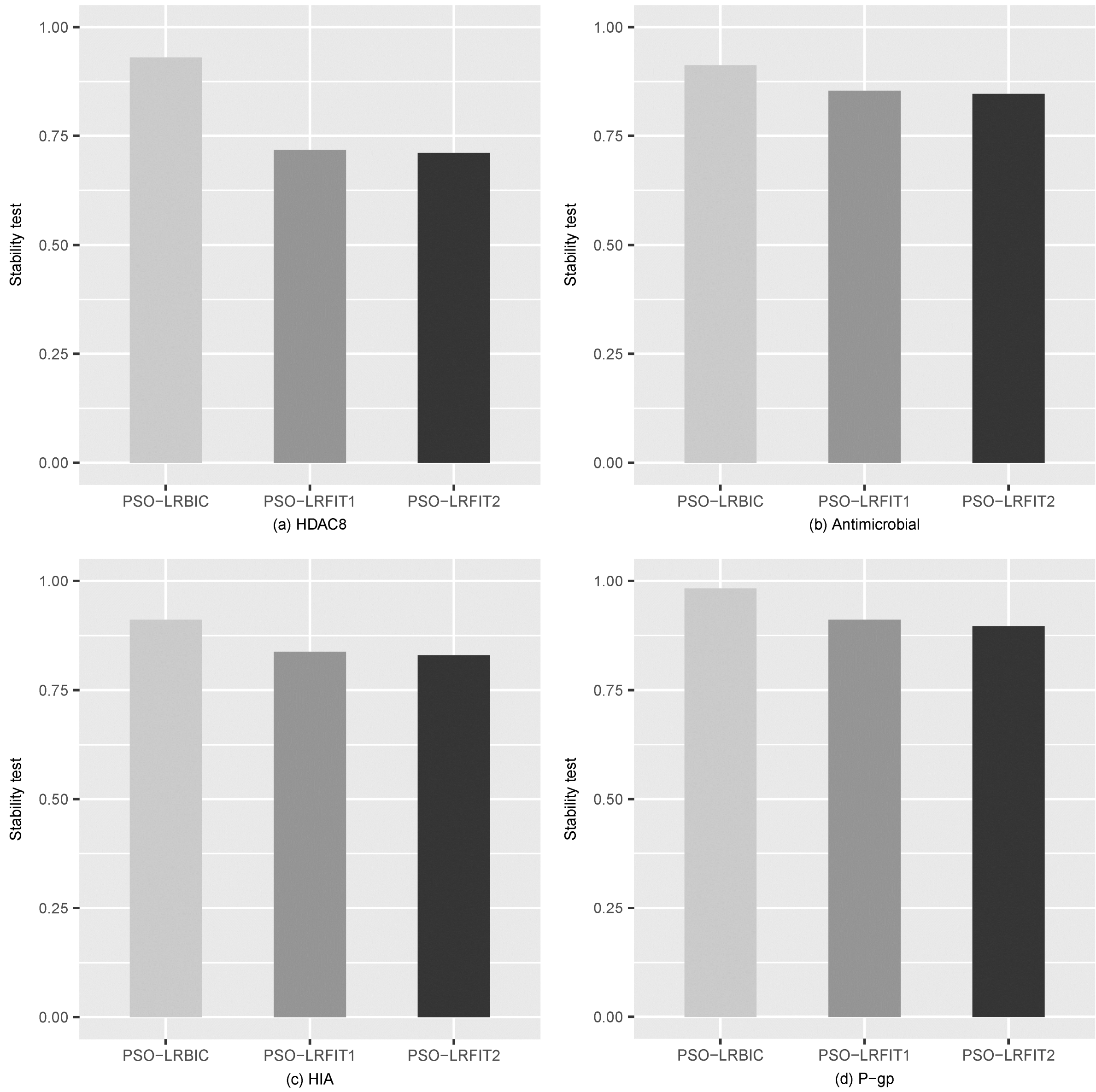
Dataset	# observations	# features	class
HDAC8	75	2014	38 inhibitors/ 37 non-inhibitor
Antimicrobial	212	3657	108 active compounds/ 104 inactive compounds
HIA	196	736	131 absorbable HIA+ Compounds/ 65 non-absorbable HIA- compounds.
P-gp	201	1136	116 mediums/ 85 non-mediums

Table 2: The evaluation criteria values obtained by the used methods

Dataset	Method	# selected features	Training dataset CA	G-mean	Testing dataset CA
HDAC8	PSO-LRBIC	**9.6 ± 1.516**	**88.61 ± 0.015**	**0.851 ± 0.011**	**84.95 ± 0.011**
	PSO-LRFIT1	10.8 ± 2.588	82.25 ± 0.029	0.817 ± 0.026	79.08 ± 0.031
	PSO-LRFIT2	11.2 ± 3.033	82.11 ± 0.011	0.812 ± 0.011	78.89 ± 0.014
Antimicrobial	PSO-LRBIC	**12.4 ± 2.302**	**97.15 ± 0.007**	**0.960 ± 0.006**	**94.57 ± 0.008**
	PSO-LRFIT1	15.6 ± 2.641	91.33 ± 0.011	0.909 ± 0.012	88.35 ± 0.013
	PSO-LRFIT2	16.3 ± 2.884	90.87 ± 0.015	0.897 ± 0.016	87.06 ± 0.016
HIA	PSO-LRBIC	**4.75 ± 0.577**	**84.07 ± 0.013**	**0.821 ± 0.011**	**80.91 ± 0.014**
	PSO-LRFIT1	6.67 ± 1.421	75.37 ± 0.065	0.748 ± 0.065	72.13 ± 0.067
	PSO-LRFIT2	6.91 ± 1.674	70.27 ± 0.071	0.701 ± 0.072	67.89 ± 0.081
P-gp	PSO-LRBIC	**12.0 ± 1.214**	**95.96 ± 0.006**	**0.943 ± 0.007**	**91.64 ± 0.008**
	PSO-LRFIT1	24.2 ± 1.562	90.07 ± 0.014	0.898 ± 0.014	87.51 ± 0.015
	PSO-LRFIT2	24.8 ± 1.771	89.86 ± 0.017	0.895 ± 0.017	87.04 ± 0.018

Table 3: Friedman and Bonferroni test results for the used methods over the ten datasets

	Friedman average rank	Friedman test results	Bonferroni test results
PSO-LRBIC	2.452	$\chi^2_{\text{Friedman}} = 16.217$, p-value (0.05)=0.0001	$\alpha_{0.05} = 6.185$, $\alpha_{0.01} = 6.839$, $\alpha_{0.10} = 5.907$
PSO-LRFIT1	8.107		
PSO-LRFIT2	8.146		



1.00
0.75
0.50
0.25
0.00
Stability test
PSO−LRBIC
PSO−LRFIT1
PSO−LRFIT2
(a) HDAC8
(b) Antimicrobial
(c) HIA
(d) P−gp

Highlights

- We examined the performance of the proposed method, PSO-LRBIC, for descriptor selection in QSAR classification.
- The PSO-LRBIC method has better performance than existing fitness functions.
- The classification ability for the PSO-LRBIC method is quite high.