Journal of Cyber Security
DOI:10.32604/jcs.2020.014045
Article

Improvement of the Firework Algorithm for Classification Problems

Yu Xue, Sow Alpha Amadou* and Yan Zhao

School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, 210044, China
*Corresponding Author: Sow Alpha Amadou. Email: Diaguissa1@outlook.fr
Received: 30 August 2020; Accepted: 19 October 2020

Abstract: Attracted numerous analysts' consideration, classification is one of the primary issues in Machine learning. Numerous evolutionary algorithms (EAs) were utilized to improve their global search ability. In the previous years, many scientists have attempted to tackle this issue, yet regardless of the endeavors, there are still a few inadequacies. Based on solving the classification problem, this paper introduces a new optimization classification model, which can be applied to the majority of evolutionary computing (EC) techniques. Firework algorithm (FWA) is one of the EC methods, Although the Firework algorithm (FWA) is a proficient algorithm for solving complex optimization issue. The proficient of the FWA isn't fulfilled when being utilized for solving the classification issues. In this paper we previously proposed optimization classification model according to the classification issue. At that point we legitimately utilize the model with FWA to solve the classification issue. Finally, to investigate the performance of our model, we select 4 datasets in the experiments, and the results indicate that an improved FWA can upgrade the classification accuracy by using this model.

Keywords: Swarm intelligence algorithm; firework algorithm; evolutionary algorithm; classification

1 Introduction

Classification problem may be a typical problem in data processing and machine learning that allots categories to a collection of data so as to assist in additional accurate predictions and analysis, which attracted many researcher's attention [1,2]. Within the past, many varieties of classification methods had been proposed such as: support vector machine (SVM) [3], decision tree (DT) [4,5], k-nearest neighbor (KNN) [6], artificial neural network (ANN) [7] and naive Bayesian classification (NBC) [8]. However, many of them always present insufficiencies because they easily trapped into local optima and mistakenly considered it as the global optima. The evolutionary computing (EC) has allowed to search the optimal parameters for classifiers [9]. Among the studies completed, for example, Prof. Holland [10] and Dr. Russell C. Eberhart proposed the genetic algorithm (GA) and the particle swarm optimization (PSO) achieved ideal classification performance, respectively. Xue et al. farther propose a novel classification model, and it can be solved classification problems straight forwardly by evolutionary algorithms (EAs). In their work, this model constructed by firework algorithm (FWA) and performs well [11]. It can be concluded that the novel classification model can solve the related problems feasibly.

Otherwise, for developing EC techniques many researchers focused on using self-adaptive mechanism and plenty effective self-adaptive EC techniques are new. Self-adaptive method has been used into EC which has performs better. For instance, to optimize the operating conditions, Fan et al. [12] introduced adaptive mutation strategy and control parameters based on DE (SSCPDE). In addition, a self-adaptive binary DE algorithm (SabDE) has put forward by Banitalebi et al. [13], the method obtained some good results on benchmark problems.

This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The classification model proposed by Xue et al. may be accustomed to overcome classification issues by EC techniques directly and simply. In this paper, firework algorithm (FWA) has been used to resolve the classification problems. Besides, a self-adaptive firework algorithm (SaFWA) [14], which employed a self-adaptive mechanism, four candidate's solution generation strategy to extend the range of the solution. Specifically, it does not employ more information in the swarm. However, Within the existing research work, since the beginning to overcome this problem, the researchers usually target modifying the parameters or operators of the classifiers. As an example, Yu et al. introduced the FWA with DE mutation operator (FWA-DM) [15]. A lot of effort has been made, but when applied to classification problems, there is a lack of accuracy. In order to figure out the problem, we proposed a method which is improvement of the firework algorithm (IFWA) in this paper. The aim of our research is to investigate the performance of IFWA once we face a classification problem and propose an efficient method to enhance the performance of the classifier. At first, we introduce the optimization classification model. Second, since the FWA is one of the EAs methods and has shortcomings. Then we introduce our IFWA through to the present supervised classification problem by using this model. Finally, we compare the FWA to IFWA. To further investigate the performance of the model, four datasets were employed in experiments.

A. Contribution of this paper

After a careful analysis on the explosion of the firework we noticed that when the sparks cannot approach the local space, they cannot contribute to the optimization process while taking a lot of resource. Thus, the current proposed methods use a new strategy of generating the individuals by updating the position of the sparks after the firework explode and keep the sparks with the good fitness value into the next iteration.

B. Organization of this paper

The organization of the paper is as follows: In section 2, we give the outline of the FWA. In section 3 we introduce the optimization classification model. Section 4 and section 5 describes the experimental design and results analysis, respectively. Finally, the last section concludes the research of the paper.

2 Related Work

FWA is the typical evolutionary algorithm to solve the complex optimization problems, which put forward by Tan et al. [16]. The processes of the algorithm are as follows:

Step 1: N fireworks are randomly initialize, and the fitness of each individual was evaluated to determinate the amplitudes of the explosion and the numbers of sparks.

Step 2: Implement the explosion and mutation operator according to their fitness values.

Step 3: Fireworks with higher fitness produced more sparks in a smaller range, otherwise, fireworks with lower fitness generate fewer sparks over a larger range.

The explosion amplitudes and sparks number are two important factors for the explosion operator. The number of sparks is described in Eq. (1):

$$s_i = m * \frac{Y_{max} - f(x_1) + \varepsilon}{\sum_{I-1}^{N}(Y_{max} - f(x_I)) + \varepsilon} \tag{1}$$

where s_i is the number of sparks for each individual, m is the number of sparks, and $Y_{\max}$ means the worst individual. x_i represents the individual and $f\left(x_i\right)$ represents the fitness for x_i, while ε denotes the smallest constant in the computer, which to prevent the denominator from being 0.

The number of sparks is limited as Eq. (2):

$$\hat{s}_i = \begin{cases} round(a.m), if\ S_i < am \\ round(b.m), if\ S_i > bm\ , a < b < 1 \\ round(a.m), otherwise \end{cases} \tag{2}$$

where a and b are two constant number, $\hat{s}_i$ is the limit of the number of sparks. For the explosion amplitude:

$$s_i = \hat{A} * \frac{f(x_i) - y_{max} + \varepsilon}{\sum_{I-1}^{N}(f(x_i) - y_{min}) + \varepsilon} \tag{3}$$

where Eq. (3). $\hat{A}$ is the sum of all the amplitudes, and $y_{\min}$ is the fitness value of the optimal individual in N individuals.

In the generating sparks part, a new method [17] is selected in this paper, which is described as follows:

$$c = \frac{(D-1)N(0.1)}{D} + \frac{C(0.1)}{D} \tag{4}$$

$$v_{id}^{t+1} = \left(mean_{id}^{t} - x_{id}^{t}\right) + \frac{c}{\sqrt{3}}\sqrt{\left((P_{id} - mean_{id}^{t}\right)^2 + (x_{id}^{t} - mean_{id}^{t})^2 + (x_{ad}^{t} - mean_{id}^{t})^2} \tag{5}$$

where N (0, 1) and C (0, 1) are randomly generated by the Gaussian distribution and Cauchy distribution, respectively. $mean_{id}^{t}$ is the same as that in [18]. t represents the t^{th} iteration in the evolutionary process. $i \in ps$ represents the current particle, and ps is the population size. $d \in D$ denote the d^{th}dimension of the search space, and D is the dimensionality of the search space. x_{id}^{t} denote the t^{th} dimension of the current position of the particle. $v_{id}^{t} \in [-v_{max}, v_{max}]$ is the velocity of the t^{th} particle in the current iteration. p_{id} is the d^{th} of personal best solution for the individual, x_{id}^{t} and x_{ad}^{t} are position vectors for two random particles, respectively.

Once explosion operator, mutation operator, and mapping rules have been employed, some sparks generated from the process need to be picked out for the next iteration. The distance-based strategy is also adapted in the FWA. In order to select the sparks for next iteration, first, the next iteration always selects the best spark. Then, the rest (N − 1) individuals are choose according to the distance maintaining the diversity of the population.

Per to the Euclidean distance which is employed in this part where $d(x_i, x_j)$ represents the Euclidean distance between the two individuals x_i and x_j.

$$R(x_i) = \sum_{j=1}^{k} d(x_i, x_j) = \sum_{j-1}^{k} \|x_i - x_j\| j \in k \tag{6}$$

where $R(x_i)$ is the sum of distances between x_i and the rest of the all opposite individuals. $j \in k$ represent the location j from k, where k represent the association both the sparks generated by the two operations as mentioned before.

By using the roulette strategy to settle on the individuals for the following iteration, the chance $p(x)$for choosing the individual $x_i$should be:

$$p(x) = \frac{L(x_i)}{\sum_{j\in k} L(x_j)} \tag{7}$$

where $L(x_i)$representes the final distance between x_i and other locations.

3 Optimization Classification Model

According to a dataset D, where 70% are used for training set T. The example of T can be described as follow:

$$T = \begin{bmatrix} x_{11}, x_{12}, \dots, x_{1d}, y_1 \\ x_{21}, x_{22}, \dots, x_{2d}, y_2 \\ \dots, \dots, \dots, \dots \\ x_{m1}, x_{m2}, \dots, x_{md}, y_m \end{bmatrix} \tag{8}$$

where (x_i, y_i) is the i^{th} example, $x_i = x_{11}, x_{12}, \dots, x_{1d}$ is the i^{th} sample, $y_i \in Y = \{1,2, \dots, I\}(i = 1,2, \dots m)$ denote the label of the i^{th} sample.

From the training set T, introduced a weight parameter $W = (w_1, w_2, \dots, w_d)$, and set up Eq. (7).

$$\begin{cases} w_1x_1 + w_2x_{12} + \cdots + w_dx_{1d} = y_1 \\ w_1x_{21} + w_2x_{22} + \cdots + w_dx_{2d} = y_2 \\ \ldots + \cdots + \cdots + \cdots \qquad\qquad = \cdots \\ w_1x_{m1} + w_2x_{m2} + \cdots + w_dx_{md} = y_m \end{cases} \tag{9}$$

From Eq. (7). It can be noticed that: The label of $x_i$could be predicted, if Eq. (8). has solution. Obviously, through some EAs techniques, these problems can be solved effectively.

$$A = \begin{bmatrix} x_{11}, x_{12}, \ldots, x_{1d} \\ x_{21}, x_{22}, \ldots, x_{2d} \\ \ldots, \ldots, \ldots, \ldots \\ x_{m1}, x_{m2}, \ldots, x_{md} \end{bmatrix} \text{and} Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} \tag{10}$$

Thus, the Eq. (9) can be brought back to the form: $A.W = Y$.

It is important to note that the number of examples is way greater than the number of dimensions. Thus, the Eq. (8) with a high probability is going to be an inconsistent equation; However, for classification problem, a rough solution is enough for the subsequent equation:

$$\begin{cases} w_1x_1 + w_2x_{12} + \cdots + w_dx_{1d} \approx y_1 \\ w_1x_{21} + w_2x_{22} + \cdots + w_dx_{2d} \approx y_2 \\ \ldots + \cdots + \cdots + \cdots \qquad\qquad \approx \cdots \\ w_1x_{m1} + w_2x_{m2} + \cdots + w_dx_{md} \approx y_m \end{cases} \tag{11}$$

According to the previous equation we notice that:

$$\begin{cases} y_1 - \delta \le w_1x_1 + w_2x_{12} + \cdots + w_dx_{1d} < y_1 + \delta \\ y_2 - \delta \le w_1x_{21} + w_2x_{22} + \cdots + w_dx_{2d} < y_2 + \delta \\ \ldots \quad \le \quad \ldots + \ldots + \cdots + \cdots \quad < \quad \ldots \\ y_m - \delta \le w_1w_1x_{m1} + w_2x_{m2} + \cdots + w_dx_{md} < y_m + \delta \end{cases} \tag{12}$$

From y_i, it is possible to predict the label of x_i if the condition: $y_1 - \delta \le w_1x_1 + w_2x_{12} + \cdots + w_dx_{1d} < y_1 + \delta \quad (y_i \epsilon \{1,2,\ldots I\})$ is satisfied. Where, δ is a small threshold, and EAs can solve this problem by using a type of objective function:

$$\min\, (f(w) = \sqrt{\textstyle\sum_{i=1}^{m} \sum_{j=1}^{d} (w_j x_{ij})^2}) \tag{13}$$

The model is feasible under the condition that Eq. (10) is satisfied.

To make sure the lower and the upper boundary of w_i, $i = 1,2, \ldots d$, there are a lot of ways, one of them is:

$$\pm\sigma \frac{\sum_{i=1}^{N} y_i}{\sum_{I=1}^{N} \sum_{j=1}^{d} x_{ij}} \tag{14}$$

4 Experiment Design and Details

4.1 Datasets Description of Data

The datasets utilized in the experiment are show in Tab. 1. Only four datasets are selected for the experiment to check the performance of the designed model and they are: Iono2, sonar, wine and iris.

Each dataset has a particular structure and has: The value of examples, labels and features. As mentioned in the second part of the paper 70% of the data is used for the training sets and 30% for the test sets. The table below shows the structure of the data used.

Table 1: Data used for the experiments

Datasets	Value of Examples	Value of labels	Value of features
Iono2	351	2	34
Sonar	208	2	60

Wine	178	3	13
Iris	150	3	4

4.2 Parameters Settings of IFWA

As we know, 4 data sets have been used and IFWA were employed to optimize our model on each dataset by finding W.

For the FWA, the maximum number of evaluations is set to 500000. The rest of the settings for FWA are:$n = 0.8$, $m = 64$, $a = 0.64$, $\hat{B} = 0.8$, $\hat{A} = 8$, and $\hat{m} = 8$.

Once W is found, for each feature $(x_i, y_i)$and the whole data set D, we calculated the classification accuracy by$-0.5 \leq (w.x_i - y_i) < +0.5$, finally we deemed the labels was predicted efficiently.

5 Experimental Results and Discussion

As we can see in table 3, each dataset has four values: max, min, mean and std. "max" and "min" denotes the maximum and the minimum values of classification accuracy, while "mean" and "std" represent the mean and standard deviation obtained in 30 trials. After the experiments, the performance of each datasets (Irono2, sonar, wine and iris) in some situations is good and promising with high classification accuracy value. The classification accuracy on iris is 0.9569 and the minimum value of mean on the whole datasets is 0.8504. Also, note that the values of std are lower in some cases and same for the amplitude which is small.

Table 2: Classification accuracy on four datasets with FWA

Classification accuracy of FWA				
Datasets	**Max**	**Min**	**Mean**	**Std**
Iris	0.966	0.94	0.9586	0.966
Iono2	0.7834	0.7037	0.7551	0.0095
Sonar	0.7163	0.6586	0.6912	0.0059
Wine	0.8988	0.7471	0.8491	0.0408

Table 3: Classification accuracy with the IFWA

Classification accuracy of IFWA				
Datasets	**Max**	**Min**	**Mean**	**Std**
Iris	0.9569	0.933	0.9989	0.966
Iono2	0.7997	0.7025	0.8504	0.0098
Sonar	0.7259	0.6442	0.9023	0.0106
Wine	0.8876	0.7865	0.9001	0.0206

6 Conclusion

Since the past, research has intensified and solutions are becoming more and more positive. In this paper, our IFWA has been used on four datasets to efficiently find the best good solution. The experiments show that the EA which employed our new classification optimization can be solve classification problems effectively. The results are being improved to get the best version of IFWA when it is used to classification problems. The next work will focus especially on the optimization of our model. Thus, the model will be reviewed to make it easier and efficient and planning to use more datasets in order to text the performance of IFWA.

Funding Statement: This work was partially supported by the Science and technology program of ministry of Housing and Urban-Rural Development (2019-K-142), the Entrepreneurial team of Sponge City (2017R02002).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] E. Viegas, A. O. Santin, A. Franca, R. Jasinski, V. A. Pedroni and L. S. Oliveira, "Towards an energy-efficient anomaly-based intrusion detection engine for embedded systems," *IEEE Transactions on Computers*, vol. 66, no. 1, pp. 163–177, 2016.

[2] X. Li, S. Han, L. Zhao, C. Gong and X. Liu, "New dandelion algorithm optimizes extreme learning machine for biomedical classification problems," *Computational Intelligence and Neuroscience*, vol. 2017, pp. 1–13, 2017.

[3] Y. Xu, Z. Yang and X. Pan, "A novel twin support-vector machine with pinball loss," *IEEE Transactions on Neural Networks Learning System*, vol. 28, no. 2, pp. 359–370, 2017.

[4] O. T. Yildiz, "VC-dimension of univariate decision trees," *IEEE Transactions on Neural Networks Learning System*, vol. 26, no. 2, pp. 378–387, 2015.

[5] R. C. Barros, D. D. Ruiz and M. P. Basgalupp, "Evolutionary model trees for handling continuous classes in machine learning," *Information Sciences*, vol. 181, no. 5, pp. 954–971, 2011.

[6] S. Mehta, X. Shen, J. Gou and D. Niu, "A new nearest centroid neighbor classifier based on k local means using harmonic mean distance," *Information*, vol. 9, no. 9, pp. 234, 2018.

[7] M. N. Bojnordi and E. Ipek, "Memristive boltzmann machine: A hardware accelerator for combinatorial optimization and deep learning," in *Proc. E3S*, Berkeley, CA, USA, vol. 2018, pp. 1–3, 2018.

[8] S. Liu, G. Mingas, and C. S. Bouganis, "An unbiased mcmc fpga-based accelerator in the land of custom precision arithmetic," *IEEE Transactions on Computers*, vol. 66, no. 5, pp. 745–758, 2017.

[9] L. Yi, "Study on an improved PSO algorithm and its application for solving function problem," *International Journal of Intelligent Systems Technologies and Applications Smart Home*, vol. 10, no. 3, pp. 51–62, 2016.

[10] J. H. Holland, "Genetic Algorithms," *Scientific American*, vol. 267, no. 1, pp. 66–73, 1992.

[11] Y. Xue, B. Zhao and T. Ma, "Classification based on fireworks algorithm," *Communication in Computer and Information Science*, vol. 682, pp. 35–40, 2016.

[12] Q. Fan and X. Yan, "Differential evolution algorithm with self-adaptive strategy and control parameters for P-xylene oxidation process optimization," *Soft Computing*, vol. 19, no. 5, pp. 1363–1391, 2015.

[13] A. Banitalebi, M. I. A. Aziz, and Z. A. Aziz, "A self-adaptive binary differential evolution algorithm for large scale binary optimization problems," *Information Sciences*, vol. 367–368, pp. 487–511, 2016.

[14] Y. Xue, B. Zhao, T. Ma and W. Pang, "A self-adaptive fireworks algorithm for classification problems," *IEEE Access*, vol. 6, pp. 44406–44416, 2018.

[15] C. Yu, L. Kelley, S. Zheng and Y. Tan, "Fireworks algorithm with differential mutation for solving the CEC 2014 competition problems," in *Proc. CEC*, pp. 3238–3245, 2014.

[16] Y. Tan and Y. Zhu, "Fireworks algorithm for optimization," in *Proc. ICSI*, Beijing, China, pp. 355–364, 2010.

[17] Y. Wang, B. Li, T. Weise, J. Wang, B. Yuan and Q. Tian, "Self-adaptive learning based particle swarm optimization," *Information Science*, vol. 181, no. 20, pp. 4515–4538, 2011.

[18] Y. Xue, S. Zhong, Y. Zhuang and B. Xu, "An ensemble algorithm with self-adaptive learning techniques for high-dimensional numerical optimization," *Applied Mathematics and Computation*, vol. 231, pp. 329–346, 2014.