Mathematical Biosciences and Engineering

http://www.aimspress.com/journal/MBE

MBE, 16(5): 4135–4150.
DOI: 10.3934/mbe.2019206
Received: 25 January 2019
Accepted: 08 May 2019
Published: 13 May 2019

Research article

Prediction of crime tendency of high-risk personnel using C5.0 decision tree empowered by particle swarm optimization

Chunxue Wu[1], **Fang Yang**[1], **Yan Wu**[2] **and Ren Han**[1,*]

[1] School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai, 200093, China
[2] The School of Public and Environmental Affairs, Indiana University, Bloomington, USA

* **Correspondence:** Email: campushr@163.com; Tel: +8618521309518.

Abstract: The research on the big data in the security and protection industry has been increasingly recognized as the hotspot in case of the rapid development of the big data. This paper mainly focuses on addressing the problem that predicts the criminal tendency of the high-risk personnel based on the recorded behavior data of the high-risk personnel. Therefore, we propose a novel predictive model that is the crime tendency of high-risk personnel using C5.0 based on particle swarm optimization. In this model, the C5.0 decision tree algorithm is first used as a classifier, in which repeated tenfold cross-validation is used and then continuously tuned according to the custom fitness function based on particle swarm optimization. In addition, the classification accuracy, the reduced number of feature subset, specificity and sensitivity under different algorithms are compared. Finally, the proposed model has higher accuracy through the optimal value of the particle position, the error rate of the cost under different iterations and the trend and the concavity and convexity of ROC curve. The experimental results show that the proposed model has a good effect on the predictive classification, which may provide guidance for predicting crime tendency of high-risk personnel.

Keywords: big data; high-risk personnel; particle swarm optimization; C5.0; K-fold cross-validation

1. Introduction

With the fast development of networking, data storage, and the data collection capacity, big data are now rapidly expanding in all science and engineering domains, including physical, biological and biomedical sciences. In any case, the era of big data has arrived. Every day, 2.5 quintillion bytes of data was created and 90 percent of the data in the world today was produced within the past two years. Meanwhile, big data could be structured, semi-structured, or unstructured, which encounters more challenges when performing data storage and processing tasks [1]. To extract useful information from the massive data, data mining is usually used in which the data mining could help to obtain hidden information and patterns in the data. The distributed mining for massive data usually depends on some technologies of cloud computing, such as the distributed processing, the distributed database and the cloud storage, as well as the virtualization technologies. However, how to hand these massive heterogeneous data in highly distributed environments becomes a huge challenge, especially in cloud platforms. Therefore, in [2] authors provided a cloud computing based functional framework that identifies the acquisition, management, processing and mining areas of IoT big data, and defined and described several associated technical modules in terms of their key characteristics and capabilities, as well as analyzed the current research in IoT application, identified the challenges and opportunities associated with IoT big data research. In recent years, with the rapid development of smart city, intelligent transportation and other industries, large-scale integration, large-scale networking, and cloud technologies have pushed the security and protection industry into the era of big data, i.e., the big data in the security and protection industry. A major challenge facing all law-enforcement and intelligence-gathering organizations is accurately and efficiently analyzing the growing volumes of crime data. Data mining is a powerful tool that enables criminal investigators who may lack extensive training as data analysts to explore large databases quickly and efficiently. Therefore, in [3] authors present a general framework for crime data mining that draws on experience gained with the Coplink project, which researchers at the University of Arizona have been conducting in collaboration with the Tucson and Phoenix police departments since 1997.

In fact, since the high dimension and complex structure of the big data, both of them will have an effect on the accuracy of the big data prediction, thus, how to reduce the dimension becomes a huge challenge. Feature selection is widely used as the first phase of a classification task to reduce the dimensions of the problem, eliminate noise, increase speed, and mitigate memory constraints by eliminating uncorrelated or redundant features. To cope with it, many researchers proposed the method of reducing the data dimension, such as the rough sets in the filter method as often used. Authors in [4] propose a new feature selection strategy based on rough sets and particle swarm optimization. Rough sets have been used as a feature selection method with much success. However, the evaluation of a feature set still needs to learn the algorithm, so the optimal subset obtained by the filter method is weaker than the wrapper method in terms of classification accuracy. In the wrapper method, the learning algorithm directly participates in the evaluation of feature subsets. In [5], particle swarm optimization is employed to select feature subset for classification task and train RBF neural network simultaneously. This approach could select as small-sized feature subset as possible to satisfy high accuracy requirement with rational training time. Experimental results show that this method is attractive. Therefore, the accuracy of the classification can be improved by using the particle swarm optimization to reduce the data dimension, on the other hand, to select an appropriate classification theory is another important issue to guarantee the accuracy. C5.0 is widely used for the actual classification problem, and it also attracts more attentions because of its significantly lower error rate and less memory. In [6], four different classification techniques are used to build classifier models. Experimental results show that C5.0 classifier is the best with respect to accuracy, precision, and specificity. Based on the above analysis, we propose a predictive model, i.e., the criminal tendency of high-risk personnel using C5.0 based on particle swarm optimization. In this model, the

C5.0 decision tree algorithm is first used as a classifier, in which repeated tenfold cross-validation is used and then continuously tuned according to the custom fitness function based on particle swarm optimization. In addition, the classification accuracy, the reduced number of feature subset, specificity and sensitivity under different algorithms are compared. Finally, the proposed model has higher accuracy through the optimal value of the particle position, the error rate of the cost under different iterations and the trend and the concavity and convexity of ROC curve. The experimental results show that the proposed model has a good effect on the predictive classification, which may provide guidance for predicting crime tendency of high-risk personnel.

The main contributions of this study can be considered as follows:

(1) We propose a predictive model about crime tendency of high-risk personnel using C5.0 based on particle swarm optimization, and verify its performance under different algorithms. The experimental results show that the proposed model in this article has the best accuracy.

(2) The proposed model can be applied to the field of the big data in the security and protection industry to predict the classification for the criminal tendency of high-risk personnel.

The rest of the paper is organized as follows. Section 2 reviews the related work on crime analysis of high-risk personnel and classification prediction. Section 3 presents the methods that we used for our study. Then the proposed PSO-C5.0 method is described in the detailed description in Section 4. After that, the verification of the experiment is given and we analyze the results of the experiment in Section 5. Section 6 draws the conclusion and discusses the future work.

2. Related works

In recent years, predictive classification is one of the most essential and important tasks in data mining and machine learning. In recent years, with the rapid development of smart city, intelligent transportation and other industries, large-scale integration, large-scale networking, and cloud technologies have pushed the security and protection industry into the era of big data, i.e., the big data in the security and protection industry. Researchers have fully applied machine learning methods to practical crime problems to help case handlers quickly and efficiently solve crimes. In addition, many researchers have highlighted the potential of predictive classification to provide decision support for the public security agencies and law enforcement officers. Over the last few years, researchers have conducted extensive research on predictive classification problems. Many of them showed good classification accuracy.

Machine learning, one of major branches of artificial intelligence and the most rapidly developing subfield of AI research, has been used for years in the criminal domain for the intelligent data analysis. Not only can the results of some future cases be predicted, but potential relationships in crime data can also be found. It holds a great potential for predictive classification in criminal datasets, as evident from recent literature survey [7–9]. Authors in [10] compare different approaches to the problem of forecasting the number of crimes in different areas of the city. The predictive factors used in these models have been selected using the feature selection techniques. This approach increases the accuracy of predictions and avoids the model’s overfitting. In [11], authors present a comparative study on correlation and information gain algorithms to evaluate and produce the subset of crime features. The results of the experiment demonstrated that, the correlation method outperformed information gain and human expert with a mean accuracy of 96.94% for entire classifier and FSs with 13 optimal features selection. In [12] authors presents a dynamic network model for improving service resilience to data loss. The network model identifies statistically significant shared temporal trends across multivariate spatiotemporal data streams and utilizes these trends to improve data prediction performance in the case of data loss. The model also correctly identifies all the optimal network connections, according to prediction error minimization.

Many classification methods have been used by researchers in a variety of fields, such as medical domain, financial field and so on [13–15]. Authors in [16] propose a new deep learning method, the greedy deep weighted dictionary learning for mobile multimedia for medical diseases analysis. The results show that the learning method has a good effect on the classification of mobile multimedia for medical diseases, and the accuracy, sensitivity, and specificity of the classification have good performance, which may provide guidance for the diagnosis of disease in wisdom medical. Authors in [17] proposed a novel model which is the improved K-means algorithm and the logistic regression algorithm based on data mining techniques for predicting type 2 diabetes mellitus (T2DM). The conclusion shows that the model attained a 3.04% higher accuracy of prediction than those of other researchers. In [18] authors propose the design of the BP neural network and select the smallest simulation error of the BP neural network as a forecasting model of petroleum projects. Experimental results show that the scheme about the BP neural network of petroleum project evaluation is effective and has over performance than traditional schemes in terms of convenience and accurate decision-making suggestions. Xuehui Meng developed three predictive models (logistic regression, artificial neural networks and decision tree) then compared the performance by using 12 risk factors. The study suggested that the decision tree algorithm (C5.0) had the best classification accuracy of 76.13% [19]. A new classification method that combines fused lasso and elastic net as regularization for linear support vector machine (SVM), which uses huberized hinge loss as the loss function is proposed in [20], named oriented feature selection SVM (OFSSVM). In [21] authors propose an improved gene selection method based on binary particle swarm optimization (BPSO) and prior information. Constrained by the gene-to-class sensitivity information, the new method can select functional gene subsets which are significantly sensitive to the samples' classes. Experimental results verify the efficiency and effectiveness of the proposed gene selection method. Authors in [22] propose an algorithm for feature selection based on two cooperative ant colonies, which minimizes two objectives: the number of features and the classification error. Two pheromone matrices and two different heuristics are used for these objectives. The performance of the method is compared with other features selection methods, achieving equal or better performance. In [23] a score-based criteria fusion feature selection method (SCF) is proposed for cancer prediction, and this method aims at improving the prediction performance of the classification model. Experiments verify that SCF is able to find more discriminative features than the competing methods and can be used as a preprocessing algorithm to combine with other methods effectively.

3. Methods

3.1. binary particle swarm optimization algorithm

Binary PSO (BPSO) is an evolutionary computation technique proposed by Kennedy and Eberhart in 1997 and used for feature selection [24]. In BPSO, a swarm consists of N particles moving around in a D-dimensional search space. The whole swarm move in the search space to search for the best solution by updating the position of each particle based on the experience of its own and its neighboring particles. During movement, the current position of particle *i* at *t* iteration is represented by a vector $X_i^{(t)} = (x_{i1}, x_{i2}, ..., x_{iD})$, which is used to evaluate the quality of particle. The velocity of particle *i* in the *t*th iteration is represented as $V_i^{(t)} = (v_{i1}, v_{i2}, ..., v_{iD})$, which is limited by a predefined maximum and minimum velocity. The best previous position of a particle (*pbest*) is recorded as $P_i = (p_{i1}, p_{i2}, ..., p_{iD})$ and the best position (*gbest*) obtained by the population thus far is called $P_g = (p_{g1}, p_{g2}, ..., p_{gD})$. Based on *pbest* and *gbest*, BPSO searches for the optimal solution by updating the velocity and the position of each particle according to the following equations:

$$V_i^{(t)} = wV_i^{(t-1)} + c_1 rand()(Pi - X_i^{(t-1)}) + c_2 rand()(Pg - X_i^{(t-1)}) \quad (1)$$

$$X_i^{(t)} = \begin{cases} 1, rand < sigmoid(V_i^{(t)}) \\ 0, otherwise \end{cases} \quad (2)$$

$$sigmoid(V_i^{(t)}) = \frac{1}{1 + e^{-V_i^{(t)}}} \quad (3)$$

Where *w* is inertia weight, c_1 is cognitive learning factor, c_2 is social learning factor and $rand()$ is random numbers between 0 and 1. In the binary particle swarm optimization algorithm, the positional component $X_i^{(t)}$ of particle *i* takes a value of 0 or 1, and 0 and 1 just indicate whether an attribute is selected. The positional component of a particle can represent a subset of attributes, and then a suitable fitness function is designed to evaluate the subset of attributes represented by the particle to implement a feature selection method based on binary particle swarm optimization. The graph of the sigmoid function is as follow, and it is a monotonically increasing function.

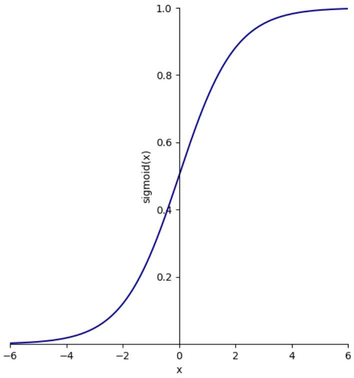


Figure 1. Sigmoid function.

3.2. Decision tree

In the last few years, a great number of algorithms have been developed for classification based data mining. Decision tree is an important classification algorithm in data mining. The main advantage of decision tree algorithms is that it is easy to construct and the resulting trees are readily interpretable. It is commonly used in various areas. Researchers have developed a variety of decision tree algorithms over a period of time with enhancement in performance and ability to handle different types of data. Popular decision tree algorithms including CART, C5.0, J48 etc. [25].

3.2.1. Classification and regression trees

Classification and regression trees (CART) is a kind of classification data mining algorithm. CART essentially divides the feature space into two parts and can split the scalar and continuous attributes. The two basic ideas of the classification tree: one is to recursively divide the training samples into trees, and the other is to use the verification data for pruning. The goal of the classification tree is to divide the data into smaller and more homogeneous groups. Homology here means that the split nodes are purer (That is, there is a large sample size of each class at each node). In classification, a simple way to define purity is to maximize accuracy or to minimize classification errors. However, accuracy as a measure of purity is a bit misleading because the focus of the method is to split the data by minimizing misclassification rather than splitting the data by classifying the samples primarily into a class. In view of the shortcomings of the above metrics, there are two alternative metrics, the Gini coefficient and the cross entropy, also known as information or divergence, which shift the focus from precision to purity [26]. Let $x_1, x_2, \cdots, x_n$ be the n attributes of a single sample and y be the category. Select an independent variable x_i, then select a value w_i of x_i, and w_i divides the n-dimensional space into two parts, $x_i \le w_i$ and $x_i > w_i$. For non-continuous variables, the value is equal to or not equal to the value. During the division, the $Gini$ indicator is used as the standard. Assuming that a sample has a C class, the impurity of the node is defined as:

$$Gini(A) = 1 - \sum_{i=1}^{C} p_i^2 \tag{4}$$

Where p_i is the probability of belonging to class i. When $Gini(A) = 0$, all samples belong to the same class.

3.2.2. C5.0

C5.0 is an improved algorithm of C4.5 algorithm [27], presented by Ross Quinlan. The attributes associated with any record in the training data set has some information gain associated with it. The C5.0 classifier works by extracting the attribute with the maximum information gain and uses this attribute field as the splitting factor. This function is performed recursively to generate multiple subsets. Eventually, a tree-like structure is generated, which follows structural hierarchy to implement the classification of the training set.

Gain Ratio is the base for a C5.0 decision tree construction. It incorporates entropy, which is the measure of un-orderliness in the training set. Entropy is given by:

$$\mathrm{H(X)} = -\sum_{i=1}^{n} p_i \log p_i \tag{5}$$

Information gain, which is a measure of the orderliness in the data set. The information gain of feature A on training data set D is defined as the difference between the empirical entropy of set D and the empirical conditional entropy of D under the given condition A, is given by:

$$g(\mathrm{D}, A) = H(D) - H(D \mid A) \tag{6}$$

Finally, the Gain Ratio is calculated by:

$$\mathrm{Gain}Rain = \frac{g(D, A)}{H(D)} \qquad （7）$$

C5.0 has additional features such as boost and different error types for different losses. Boosting refers to a general and effective method of producing a quite accurate classifier by combining rough and moderately inaccurate rules of thumb. It is based on the observation that finding many rough rules of thumb can be much easier than finding a single and highly accurate classifier. The boosting algorithm repeatedly calls weak learner, each time feeding it a different distribution over the training data. Each call generates a weak classifier, which must be combined into a single classifier expected to be more accurate than any one of the rules [28].

There are many kinds of boosting algorithms. This paper mainly uses AdaBoost [29]. The adaptive boost method generates a series of weak classifiers. The algorithm finds the best classifier based on the current sample weights at each iteration. Samples that are misclassified in the k-th iteration will be given a higher weight in the k + 1 iteration, and the correctly classified samples will have a lower weight in the next iteration. This means that the weight of the sample that is difficult to classify will continue to increase until the algorithm identifies the model that correctly classifies the samples. Therefore, the algorithm requires different aspects of the iterative learning data for each iteration, with a view to including sample regions that are difficult to classify. Each phase of the iteration calculates a phase weight based on the error rate. Calculate the classification error of basic classifier $G_m(x)$ on the training data set according to Eq 8. The nature of the phase weights described in Algorithm 1 indicates that models with higher accuracy have higher positive sample values, while models with lower accuracy have lower negative sample values. A series of weighted classifiers are combined into a single set that has a strong potential for classification, better than any individual classifier.

$$e_m = \sum_{i=1}^{N} P(G_m(x_i) \neq y_i) = \sum_{i=1}^{N} w_{mi} I(G_m(x_i) \neq y_i) \qquad (8)$$

Table 1. Algorithm process.

Algorithm 1. Adaptive boosting algorithm for two-class problem

1: One kind of sample value is marked as +1 and the other is sampled as -1

2: Each sample has the same starting weight (*1/N*)

3: Execute for *m = 1* to *M*

4: Fit a weak classifier with the weighted samples and calculate the false positive rate of the *m*th model (e_m)

5: Calculate the value of the *k*th iteration $ln((1-e_m)/e_m)$

6: Update sample weights, increase the weight of misjudged samples, and reduce the weight of correctly determined samples

7: End

8: Calculate the prediction of each sample by the booster classifier as follows: multiply the value obtained in step *m* by the model prediction result of step *m* and sum the product for all *m*. If the sum is a positive number, then the sample is judged to be a +1 class, and vice versa

4. C5.0 based on particle swarm optimization

4.1. Framework of model

In this paper, the data set is preprocessed first. The actual data is often filled with useless and redundant features, so performance degradation will occur. In this paper, the C5.0 decision tree algorithm is used as the classifier, and the particle swarm optimization algorithm is used for feature selection. In this process, repeated ten-fold cross-validation is used, and then the optimization is continuously adjusted according to the custom fitness function. The accuracy of the model is evaluated by the optimal position of the particle at different iterations, the error rate of the cost and the number of feature subsets, and finally the classification performance under different algorithms is compared. The specific implementation process is shown in Figure 2.

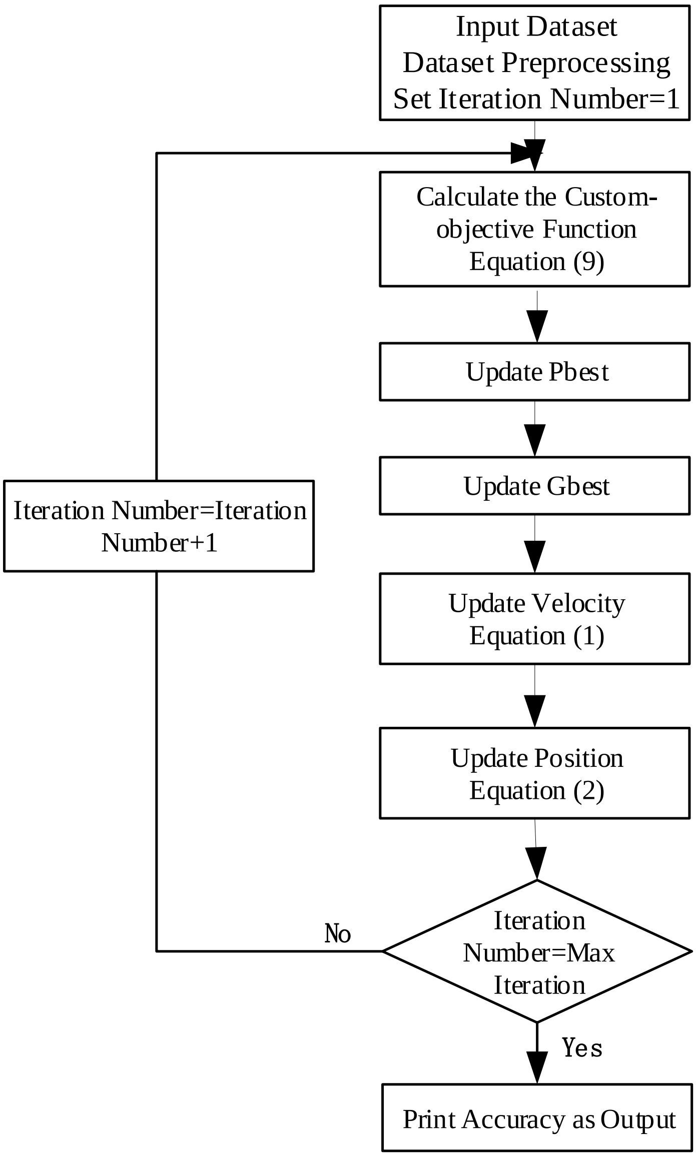

Input Dataset
Dataset Preprocessing
Set Iteration Number=1
Calculate the Custom-objective Function Equation (9)
Update Pbest
Update Gbest
Update Velocity Equation (1)
Update Position Equation (2)
Iteration Number=Max Iteration
No
Iteration Number=Iteration Number+1
Yes
Print Accuracy as Output

Figure 2. The flow chart of the proposed model.

4.2. Particle coding

The essence of feature selection is to select *N* features to form a subset among *M* features. Therefore, each feature can be defined as a one-dimensional discrete binary variable of the particle, and *M* features constitute the *M*-dimensional discrete binary space of the particle. For each particle, if the *i*th bit is 1 which means that the *i*th feature is selected, otherwise it means that the feature is not selected. Therefore, each particle represents a different subset of features, which is a candidate solution.

4.3. Fitness function

The purpose of feature selection is to find the combination of features with the strongest classification ability, and a quantitative criterion is needed to measure the classification ability of the feature combination. The algorithm aims at the joint optimization of feature subsets and C5.0 parameters, and improves the classification accuracy of C5.0 decision tree and feature subset as the independent variables of the fitness function. The fitness function formula is:

$$f = w_1 Accuracy + w_2 \frac{N_f}{N_t} \tag{9}$$

Where w_1 is the weight of classification accuracy, *Accuracy* is classification accuracy, w_2 is the weight of feature number, N_f is the number of feature subsets and N_t is total dimension of the data set. *Accuracy* is used to evaluate classifier performance. This measure is defined as the total number of correct classification over the total number of available examples. As usual most of the classification problems have two classes, including positive and negative cases. Thus, confusion matrix can be used to describe classification performance: true positives (*TP*), true negatives (*TN*), false positives (*FP*) and false negatives (*FN*). Based on the above, the accuracy is defined as:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{10}$$

This criterion is limited, especially if one of the classes is much larger than the other. With an unbalance classification problem, misclassification in the minority class will not have a large impact for the accuracy. Further, a good classification of a class might be more important than classifying other classes, and this cannot be assessed with accuracy. To account for these issues, additionally two other performance measures, commonly used in binary classification problems, were considered:

$$Sensitivity = \frac{TP}{TP + FN} \tag{11}$$

$$Specificity = \frac{TN}{FP + TN} \tag{12}$$

Sensitivity is the ratio that is accurately determined to be "occurring" in all samples in which "occurring" is observed. *Specificity* refers to the ratio at which no observed samples are accurately determined as "not occurring".

Table 2. Algorithm process.

Algorithm 2. The predictive model about crime tendency of high-risk personnel using C5.0 based on particle swarm optimization

```
Input: Dataset $D$, feature number $N$, the inertia weight $w_1$ and $w_2$ of fitness function, particle number $n$
Output: The accuracy of model
1： Preprocess dataset $D$ and divide it into train and test
2： while iteration condition is not satisfied do
3：     for $i$ = 1 to $n$ do
4：         update fitness value $f$ according to equation (9)
5：         update $pbest$
6：     end for
7：     update $gbest$
8：     for $i$ = 1 to $n$ do
9：         for $j$ = 1 to $N$ do
10：            update $V_i^{(t)}$ according to equation (1)
11：            update $X_i^{(t)}$ according to equation (2)
12：        end for
13：    end for
14： end while
```

5. Experiments and results

5.1. Experimental environment and data

The model is implemented in Python. The computer is configured with 4GB of memory, the CPU is AMD 1.90GHz, and the operating system is Win10. The rapid development of the IoT is bound to produce massive data which can be used to mine potential value. The data set used in this article is the data generated by the simulation based on the IoT. The data in the dataset has all been preprocessed into numeric variables. The number of instances of the entire data set is 25000, and the number of features is 34. The method proposed in this paper is used to deal with the two classification problem. The parameters of the particle swarm optimization algorithm are set as follows: $w = 0.3$, $c_1 = c_2 = 2$, $V_{max} = 4$, the number of population particles n is 20, $w_1 = 0.8$, $w_2 = 0.2$, and the maximum number of iterations is 100.

5.2. Experimental results

This article uses common evaluation metrics to perform performance evaluations on all models used. As shown in Table 3, the accuracy, the reduced number of features, sensitivity and specificity under different models are shown. As can be seen from the table, the accuracy of the C5.0 model on the original 34-dimensional data set is not high due to the lack of dimensionality reduction. In terms of the efficiency, the C5.0 algorithm has the shortest running time because of the lack of feature selection and parameter optimization. The PSO-CART algorithm and the PSO-C5.0 algorithm have more running time because the learning algorithm participates in the evaluation of the feature subset. The number of feature subsets selected by the PSO-C5.0 algorithm is less than that selected by the

PSO-CART algorithm compared to the PSO-C5.0 algorithm. The superiority of the PSO-C5.0 algorithm can also be seen from the sensitivity and specificity, and that of the C5.0 algorithm cannot achieve satisfactory results. Overall, the performance of the proposed algorithm model is superior regardless of accuracy, reduced number of features, sensitivity and specificity.

Table 3. Comparison of different model.

Model	Accuracy	NF	Sensitivity	Specificity
C5.0	0.891	34	0.761	0.956
PSO-CART	0.942	17	0.904	0.958
PSO-C5.0	0.957	13	0.942	0.965

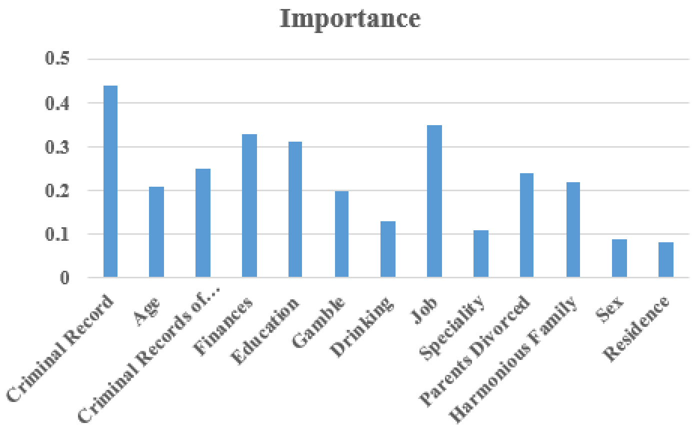
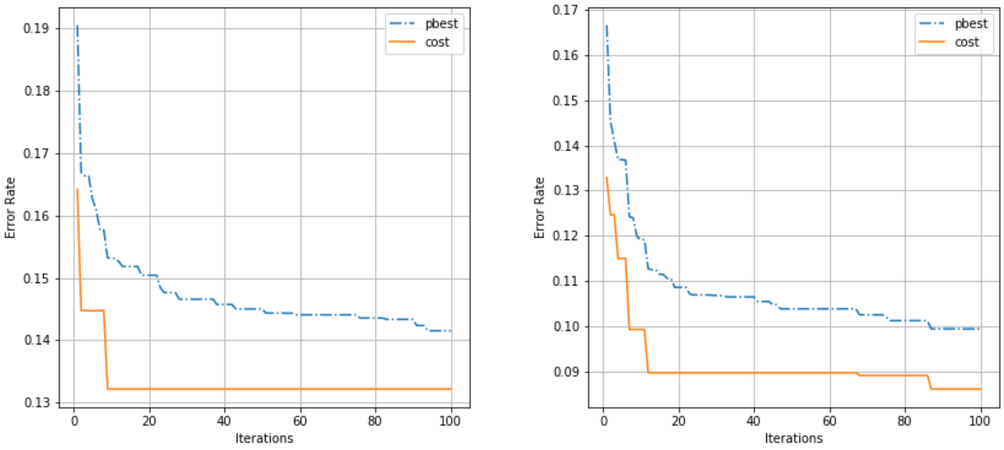


Figure 3. The importance of features.

The feature importance analysis refers to the importance relationship between the analysis feature and the target value. The relationship can be expressed by the importance coefficient to analyze the anomaly of the feature, so as to adjust and optimize the model feature. For dimensionally reduced data, the importance of its characteristics can be assessed. The way to calculate the overall importance is to record the reduction in each feature's optimization goal. If a feature is at the top of the tree, or appears multiple times in the tree, it will be more important than the feature variables that appear at the bottom of the tree or that do not appear at all. Figure 3 shows the importance of all features of the reduced dimensional data set. It can be seen from the figure that the more important features are whether there is a criminal record, whether there is work, economic status and education. It is worth noting that the main reason for whether a high-risk personnel has a criminal tendency depending on the criminal record. Therefore, a person who has the criminal record could cause higher criminal tendency than ordinary people. In addition, the job is also an important reason for the criminal tendency of high-risk people. Those who do not have jobs could have greater possibility to commit crime because of their livelihood. Among them, the economic situation and education status also have a great impact on the criminal tendency of high-risk personnel. Those with poor economic status and relatively low education level would have a greater chance of committing crimes. From the above analysis, it can be seen that high-risk personnel have a higher probability of committing crimes if they have criminal records, no jobs, poor economic conditions or low education levels. Therefore, high-risk personnel with the above characteristics should pay more attention to their behaviors in order to prevent their crimes earlier and more effectively.

Figure 4 shows the error rate that changes over different iterations. The lower the error rate, the accuracy of the model is higher. As the number of iterations increases, the error rate is correspondingly lower and lower. The solid line represents the cost. As the number of iterations increases, the error rate of the cost also decreases. The more iterations, the penalty cost assigned to the model is more accurate and the accuracy of model is higher. The dotted line represents the error rate of the current optimal value of the particle. It also decreases as the number of iterations increases, indicating that the current optimal value of the particle is also constantly approaching the global optimal value, and the accuracy of the model is higher. Figure 4(a) is a diagram of the error rate of the PSO-CART model, and Figure 4(b) is a diagram of the error rate of the PSO-C5.0 model. The comparison shows that the error rate of proposed model in this paper is lower, that is, the accuracy of model is higher regardless of the particle optimal value or the cost.



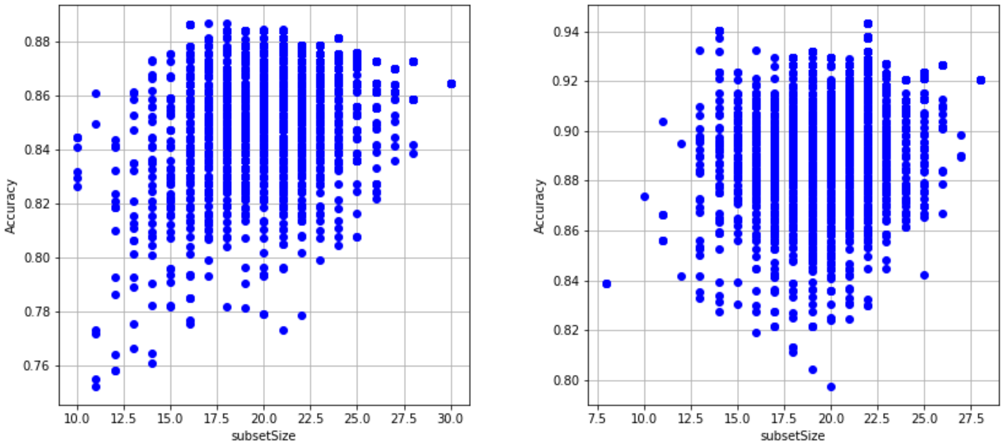
(a) PSO-CART

(b) PSO-C5.0

Figure 4. Error rate in the different iteration.

Figure 5 shows the accuracy of the model under different feature subsets. It can be seen from the figure that all features are not required to achieve high accuracy, and only half or less of them can be used to achieve satisfactory accuracy, which reduces the running time and reduces the required memory space. The size of the feature subset will affect the accuracy of the model to a certain extent. It can be seen from the figure that the accuracy is relatively high when the number of feature subsets is substantially half of the overall feature subset. That is to say, a complete feature set is not required to achieve satisfactory accuracy for a data set with a large dimension. It is only necessary to be able to take half or less of the feature set without reducing the accuracy of the model.

Figure 5(a) shows the feature subset accuracy of the PSO-CART model, and Figure 5(b) shows the feature subset accuracy of the PSO-C5.0 model. It can be seen from the figure that the feature subset of the PSO-C5.0 model is more accurate, and the number of feature subsets used is also less, so that the model has less running time and memory. On the contrary, when the number of feature subsets reaches the maximum, the accuracy of the model is reduced, so that it is known that there is a higher accuracy without using more features, and some features may affect the accuracy of the model. The PSO-CART model is less accurate, and due to using more features, the runtime and memory of the model will increase accordingly. Overall, the performance of the proposed model is better both in terms of accuracy and the number of feature subsets used.



(a) PSO-CART (b) PSO-C5.0

Figure 5. Accuracy of different subsetSize.

The ROC curve is used to reflect the comprehensive evaluation of the sensitivity and specificity of the classification results. The false positive rate is plotted on the horizontal axis and the true positive rate is plotted on the vertical axis. When the positive and negative sample distributions in the test data are transformed, the ROC curve can have good performance and it will remain unchanged. AUC is the area under the ROC curve and has a value between 0.1 and 1. As a numerical value, the quality of the classifier can be evaluated visually. The larger the value, and the performance of the classifier is better. The classification performance is reflected in the position of the ROC curve near the upper left corner, and the value of AUC will be larger. Figure 6 shows the comparison of the ROC curve of PSO-CART and PSO-C5.0. The AUC value under the PSO-CART model is 0.865, while the AUC value under the PSO-C5.0 model is 0.964, so it can be seen that the PSO-C5.0 model performs better. As can be seen from the figure, no curve will always be better than other ones, and the curve crossover will generally occur. This means that when you are interested in a particular area of the curve, you can use this to calculate the AUC value at a certain sensitivity and specificity.

We can see that the PSO-C5.0 algorithm proposed does improve the area of the ROC curve to a certain extent by observing the trend and the concavity and convexity of each curve, which is better than the other algorithm to verify the effectiveness of the algorithm. Moreover, the values of sensitivity and specificity will mutually restrict the trend of the ROC curve, which will affect the value of AUC in this respect. Overall, the ROC curve of the PSO-C5.0 model is higher than the PSO-CART model in many places, indicating that the PSO-C5.0 model is more efficient than other algorithms.

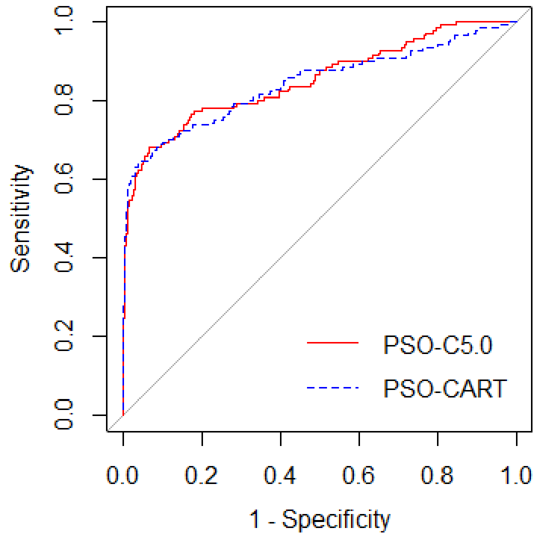


Figure 6. ROC curve.

6. Conclusions and future works

In this paper, a new predictive classification model based on particle swarm optimization algorithm and C5.0 decision tree is proposed, which is applied to prediction classification of big data in security and protection industry. In this model, the C5.0 decision tree algorithm is first used as a classifier, in which repeated tenfold cross-validation is used and then continuously tuned according to the custom fitness function based on particle swarm optimization. In addition, the classification accuracy, the reduced number of feature subset, specificity and sensitivity under different algorithms are compared. Finally, the proposed model has higher accuracy through the optimal value of the particle position, the error rate of the cost under different iterations and the trend and the concavity and convexity of ROC curve. The results show that the learning method has a good effect on the predictive classification, which may provide guidance for the prediction of crime tendency of high-risk personnel.

Although the proposed algorithm is superior to other algorithms, this is only the result of experimental verification on data set which is generated by simulation. There are some shortcomings to be further study in the future. The focus of the latter research will be on real data set to verify its classification performance. At the same time, our training sample data is too small, it is necessary to increase more data sets to verify. In addition, the decision tree should be used to mine the characteristics of people that have a criminal tendency. The future work would build the high-risk personnel risk prediction model with higher accuracy, and the decision tree can be used to excavate the general characteristics of people that have a criminal tendency.

Acknowledgments

The authors would like to appreciate all anonymous reviewers for their insightful comments and constructive suggestions to polish this paper in high quality. This research was supported by the

National Key Research and Development Program of China (No. 2018YFC0810204), National Natural Science Foundation of China (No. 61872242, 61502220), Shanghai Science and Technology Innovation Action Plan Project (17511107203, 16111107502) and Shanghai key lab of modern optical system.

Conflict of interest

All authors declare no conflicts of interest in this paper.

References

1. A. Mehmood, I. Natgunanathan, X. Yong, et al., Protection of big data privacy, *IEEE Access*, **4** (2016), 1821–1834.
2. H. Cai, B. Xu, L. Jiang, et al., IoT-based big data storage systems in cloud computing: perspectives and challenges, *IEEE Internet Things*, **4** (2017), 75–87.
3. H. Chen, W. Chung, J. J. Xu, et al., Crime data mining: A general framework and some examples, *Computer*, **37** (2004), 50–56.
4. P. Hanchuan, L. Fuhui and D. Chris, Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy, *IEEE T. Pattern Anal.*, **27** (2005), 1226–1238.
5. Y. Liu, Z. Qin, Z. Xu, et al., Feature selection with particle swarms, *Comput. Inf. Proceedings*, **3314** (2004), 425–430.
6. M. Agaoglu, Predicting instructor performance using data mining techniques in higher education, *IEEE Access*, **4** (2016), 2379–2387.
7. T. Almanie, R. Mirza and E. Lor, Crime prediction based on crime types and using spatial and temporal criminal hotspots, *Comput. Sci.*, **5** (2015), 70–89.
8. S. C. Jeeva and E. B. Rajsingh, Intelligent phishing url detection using association rule mining, *Hum. Cent. Comput. Inf. Sci.*, **6** (2016), 1–19.
9. R. Sujatha and D. Ezhilmaran, A new efficient SIF-based FCIL (SIF âFCIL) mining algorithm in predicting the crime locations, *J. Exp. Theor. Artif. In.*, **28** (2015), 561–579.
10. V. Ingilevich and S. Ivanov, Crime rate prediction in the urban environment using social factors, *Procedia Comput. Sci.*, **136** (2018), 472–478.
11. M. A. Jalil, F. Mohd and N. Maizura, A comparative study to evaluate filtering methods for crime data feature selection, *Procedia Comput. Sci.*, **116** (2017), 113–120.
12. O. Kotevska, A. G. Kusne, D. V. Samarov, et al., Dynamic network model for smart city data-loss resilience case study: City-to-city network for crime analytics, *IEEE Access*, **5** (2017), 20524–20535.
13. K. Konstantina, T. P. Exarchos, K. P. Exarchos, et al., Machine learning applications in cancer prognosis and prediction, *Comput. Struct. Biot.*, **13** (2015), 8–17.
14. C. Mao, R. Lin, C. Xu, et al., Towards a trust prediction framework for cloud services based on PSO-driven neural network, *IEEE Access*, **5** (2017), 2187–2199.
15. S. M. Vieira, L. F. Mendon ça, G. J. Farinha, et al., Modified binary PSO for feature selection using SVM applied to mortality prediction of septic patients, *Appl. Soft Comput.*, **13** (2013), 3494–3504.

16. C. Wu, C. Luo, N. Xiong, et al., A greedy deep learning method for medical disease analysis, *IEEE Access*, **6** (2018), 20021–20030.
17. H. Wu, S. Yang, Z. Huang, et al., Type 2 diabetes mellitus prediction model based on data mining, *Inform. Med. Unlocked*, **10** (2018), 100–107.
18. H. Li, X. Mao, C. Wu, et al., Design and analysis of a general data evaluation system based on social networks, *Eurasip Wireless Commu. Netw.*, **1** (2018), 109–120.
19. X. Meng, Y. Huang, D. Rao, et al., Comparison of three data mining models for predicting diabetes or prediabetes by risk factors, *Kaohsiung Med. Sci.*, **29** (2013), 93–99.
20. Y. Shen, C. Wu, C. Liu, et al., Oriented feature selection SVM applied to cancer prediction in precision medicine, *IEEE Access*, **6** (2018), 48510–48521.
21. F. Han, C. Yang, Y. Wu, et al., A gene selection method for microarray data based on binary PSO encoding gene-to-class sensitivity information, *IEEE/ACM Trans. Comput. Biol. Bioinformat.*, **14** (2017), 85–96.
22. S. M. Vieira, J. Sousa and T. A. Runkler, Two cooperative ant colonies for feature selection using fuzzy models, *Expert Syst. Appl.*, **37** (2010), 2714–2723.
23. W. Ke, C. Wu, Y. Wu, et al., A new filter feature selection based on criteria fusion for gene microarray data, *IEEE Access*, **6** (2018), 61065–61076.
24. J. Kennedy and R. C. Eberhart, A discrete binary version of the particle swarm algorithm, 1997 IEEE International Conference on Systems, Man, and Cybernetics.Computational Cybernetics and Simulation, 2002.
25. S. R. Safavian and D. Landgrebe, A survey of decision tree classifier methodology, *IEEE T. Syst. Man Cybern. B Cybern*, **21**(1991), 660–674.
26. P. A. Chou, Optimal partitioning for classification and regression trees, *IEEE T. Pattern Anal.*, **13** (1991), 340–354.
27. J. R. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufmann Publishers Inc, 1992.
28. H. Masnadi-Shirazi and N. Vasconcelos, Cost-Sensitive Boosting, *IEEE T. Pattern Anal.*, **33** (2011), 294–309.
29. W. Hu, W. Hu and S. Maybank, AdaBoost-based algorithm for network intrusion detection, *IEEE T. Syst. Man Cybern. B Cybern*, **38** (2008), 577–583.