The Study of Information Security Risk Assessment Based on Support Vector Machine

Haiyan Xie [1,a], Ying Wang[1] and Xianghong Zhang [1]

[1]*Department of Mathematics, Dalian Maritime University, 116026 Liaoning, Dalian, China*

Abstract. This paper establishes an information security evaluation model based on support vector machine by analyzing the factors affecting information security risk assessment. We compared and analyzed the different kernel functions of support vector machine by MATLAB, the experimental results show that the radial basis kernel function can minimize the training error and make the training result more accurate. Meanwhile, we compared the information security risk assessment based on support vector machine and BP neural network, the experimental results show the former has less error and shorter time. Therefore, the information security evaluation model based on support vector machine is feasible.

1 Introduction

With the advent of information age and the rapid development of information technology, information system has become an indispensable tool in social life. But the security issues that arise in the accompanying information systems are also becoming increasingly prominent. Therefore, in order to resolve the problem of information security, we must put do well in accurate and effective information security risk assessment in the first place. In the modern world, information security risk assessment has become a hot topic in various fields of the world[1-7].

Support vector machine is an emerging artificial intelligence learning algorithm, which is a new hotspot after neural network. It has the advantages of simplicity, speed, universal popularity, high training efficiency, and global optimization. The paper introduces the concept of information security risk assessment, analyzes the working principle of support vector machine and the working principle of BP neural network, and applies the two methods to information risk assessment to resolve the problem of information risk assessment.

2 Page information security risk assessment concepts and methods

Information security risk assessment refers to the process of a scientific and fair comprehensive evaluation of information systems and security attributes such as the confidentiality, integrity and availability (CIA) of information systems and information that are processed, transmitted and stored by the information technology standards, according to the relevant information technology standards.

The essence of security is that information has its own vulnerabilities, threats and values. There are several important factors in information security risk assessment: threat identification, vulnerability identification, asset identification. Information risk assessment provides a basis for the determination of security strategy and the establishment of information security system. There are many ways to evaluate the security of information, and many scholars have studied it in [8-9].

3 The introduction of theory

3.1 Support vector machine theory

Support vector machine is a machine learning method based on the principle of structural risk minimization [10], which is uses statistical theory to resolve fitting accuracy and generalization problems, this technique is widely used in pattern recognition, information security, and data fitting.

For nonlinear classification problems, support vector machine maps the training samples x to a high-dimensional space H with a nonlinear function $\varphi(\cdot)$, so that the training set which obtained in the H is linearly separable, thus obtaining the following classification hyperplane

$$w \cdot \phi(x) + b = 0 . \tag{1}$$

Under the KKT condition, the dual problem of the optimal classification hyperplane problem can be described as:

[a] Corresponding author: winteriscoming@sina.com

$$\min \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l} y_i y_j \alpha_i \alpha_j \phi(x_i)\cdot\phi(x_j) - \sum_{j=1}^{l}\alpha_j = \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l} y_i y_j \alpha_i \alpha_j K(x_i,x_j) - \sum_{j=1}^{l}\alpha_j \qquad (2)$$

$$s.t. \sum_{i=1}^{i} y_i \alpha_i = 0 \qquad 0 \le \alpha_i \le C, i=1,2,...,l \qquad (3)$$

Among $\alpha_i, i=1,2,...,l$ is the Lagrange multiplier, $K(x_i,x_j)=\phi(x_i)\cdot\phi(x_j)$ called the kernel function.

The kernel function is one of the most important core parts of the support vector machine, the mainly function of the kernel function is to map the parameter samples of the nonlinear class problem to the high-dimensional space to realize the linear classification form in the support vector machine. There are many kernel functions of support vector machines. Common support vector machine kernel functions include linear kernel functions, polynomial kernel functions, Gauss kernel functions, radial basis kernel functions, and neural network kernel functions.

This paper mainly uses three methods to carry out experiments, linear kernel functions, polynomial kernel functions, radial basis kernel functions. Its algorithm process is:

The first step is to give a group input samples $x_i, i=1,2,...,l$ and its corresponding expected output $y_i \in \{1,-1\}$.

The second step is to select the appropriate kernel function $K(x_i,x_j)=\Phi(x_i)\Phi(x_j)$ and related parameters.

The third step is resolve the following optimization problem

$$\max W(\alpha)=\sum_{i=1}^{l}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{l}\alpha_i\alpha_j y_i y_j K(x_i,x_j) \qquad (4)$$

under the constraints $\sum_{i=1}^{l} y_i\alpha_i = 0$ and $0 \le \alpha_i \le C$, then we can obtain the optimal value α_i^*.

The fourth step is to calculate the optimal weight value according to $w^* = \sum_{i=1}^{l}\alpha_i y_i \Phi(x_i)$.

The fifth step is to calculate the result for the vector to be classified x according to

$$f(x)=\mathrm{sgn}\left\{\sum_{i=1}^{l} y_i\alpha_i^* K(x_i,x)+b^*\right\} \qquad (5)$$

and decide x belong to which category.

3.2 BP Neural Network Theory

The BP neural network either a single layer neural network or a multilayer neural network. The first layer is input layer, the last is output layer, and the middle is hidden layer. Suppose the number of neurons in the q layer is n_q. The connection weight coefficient of input the q layer and the i neuron is $\omega_{ij}^q, i=1,2,...,n_q,\ j=1,2,...,n_{q-1}$, the threshold is θ_i^q, the ideal output is $d_{pi}, i=1,2,...,n_q$. The input-output relation of the network is:

$$S_i^q = \sum_{j=1}^{n_q-1} w_{ij}^q x_j^{q-1} + \theta_i^q \qquad (6)$$

$$x_i^q = f(S_i^{(q)}) = \frac{1}{1+e^{-\mu S_i^{(q)}}} \qquad (7)$$

The fitting error function is :

$$E_p = \frac{1}{2}\sum_{i=1}^{n_Q}(d_{pi} - x_{pi}^Q)^2 \qquad (8)$$

The BP network weight correction is:

$$\omega_{ij}^{(q)}(k+1) = \omega_{ij}^{(q)}(k) + \alpha D_{ij}^{(q)}(k+1) \qquad (9)$$

among them,

$$D_{ij}^q = \sum_{p=1}^{P}\delta_{pi}^{(q)} x_{pj}^{(q)} \qquad (10)$$

$$D_{ij}^q = \sum_{p=1}^{P}\delta_{pi}^{(q)} x_{pj}^{(q)} \qquad (11)$$

$$\delta_{pi}^{(Q)} = \mu\left(d_{pi} - x_{pi}^{(Q)}\right)x_{pi}^{(Q)}\left(1 - x_{pi}^{(Q)}\right) \qquad (12)$$

4 The information security model based on support vector machine

4.1. Model establishment

By subdividing the three elements of information security, it can be divided into more specific factors, namely, lack of information, information leakage, information damage, and service busy; Network vulnerability, communication protocol vulnerability, hardware defects, software defects; environmental degradation, malicious attacks, management errors, communication interference, etc. There are many ways to obtain data, such as use the Delphi method [6] to obtain data, by inviting experts in risk assessment to let them evaluate certain information risk levels based on their own experience and actual situation on the spot [11]. It also could via sampling directly, sampling from financial and information reports, etc. In the current environment of information security risk assessment, most information data is confidential, so it is very difficult to obtain real data, so does hire experts. Therefore, this experiment‘s data adopt the data of paper in [12]. The sample data set is shown in Table 1.

Table 1 gives a total of 12 groups of samples, with 6 input neurons, marked as $R_1, R_2, ..., R_6$, which respectively represent "unauthorized access", "unauthorized access system resources", "data leakage", "denial of service", "unauthorized modification of data and software", "system function collapse". L_m is a security risk assessment value. When the system risk assessment value is L_m ≤0.3, it is a low risk system, when the system risk assessment value is 0.3< L_m ≤0.7, it is a moderate risk system, when the system risk assessment value is L_m >0.7, it is a high risk system. There were 50 training samples, including training input samples and test samples. Table 1 is the 12 system's sample data, and the latter 4 are the test samples.

Table 1. Sample data set

M	R1	R2	R3	R4	R5	R6	L_m
1	0.4	0.3	0.4	0.6	0.5	0.3	0.38
2	0.3	0.5	0.8	0.4	0.2	0.5	0.45
3	0.5	0.6	0.2	0.8	0.7	0.5	0.62
4	0.4	0.3	0.4	0.2	0.3	0.4	0.38
5	0.3	0.2	0.3	0.2	0.2	0.4	0.26
6	0.4	0.3	0.4	0.6	0.5	0.4	0.47
7	0.7	0.6	0.8	0.7	0.8	0.6	0.75
8	0.3	0.2	0.3	0.2	0.4	0.2	0.24
9	0.5	0.4	0.5	0.6	0.4	0.4	0.48
10	0.5	0.5	0.6	0.8	0.4	0.5	0.56
11	0.6	0.7	0.8	0.5	0.7	0.7	0.72
12	0.3	0.2	0.4	0.5	0.3	0.2	0.35

4.2 Experiment analysis

4.2.1 The model of support vector machine

The experiment uses MATLAB as a test platform. The kernel functions used in training mainly include linear kernel functions, Gaussian kernel functions, and polynomial kernel functions.

The core program is as follows:

```
model=svmtrain(y,x,'-s 3 -t 2 -c 2.2 -g 2.8 -p 0.0001');
datat=etime(clock,t1).
```

The last 4 sets of data are used as test sample sets, and the test results are shown in Table 2.

Table 2. SVM testing result

Nuclear Category	Nuclear Width	Time Data	Real Data	Forecast Data	Average Absolute Error
Radial Basis	0.001	0	0.48	0.4812	1.33E-06
			0.56	0.5589	
			0.72	0.7188	
			0.35	0.3489	
Polynomial	0.001	0.015	0.48	0.4785	1.54E-06
			0.56	0.5613	
			0.72	0.7187	
			0.35	0.3493	
Linear	0.001	0	0.48	0.4784	0.000443
			0.56	0.5708	
			0.72	0.7198	
			0.35	0.3093	

As we can be seen from Table 2, when the core width is 0.001, use radial basis function not only takes less time, but also minimized the average error of training results, thus we can see that radial basis function has greater superiority than polynomial kernel function and linear kernel function. On the basis of radial basis function, change the nuclear width, and analyzed its error and time. The results are shown in Table 3. From the Table 3, we can see that the more nuclear width, the more time, the more error.

Table 3. Comparison of the results of changing the SVM nuclear width

Nuclear Width	Time Data	Real Data	Forecast Data	Average Absolute Error
0.1	0.002	0.4800	0.4828	0.1537
		0.5600	0.5446	
		0.7200	0.6202	
		0.3500	0.3857	
0.01	0.001	0.4800	0.4812	0.0347
		0.5600	0.5589	
		0.7200	0.7188	
		0.3500	0.3489	
0.001	0	0.4800	0.4805	0.0046
		0.5600	0.5598	
		0.7200	0.7198	
		0.3500	0.3489	

4.2.2 BP neural network model

(1) Establish a model. According to the sample data in Table 1, the risk factor is six, and the evaluation result is one, so the number of input neurons is six, and the number of output neurons is one. For the number of neurons in the hidden layer, there is no uniform formula that can be used. Usually, we could use the empirical formulas, such as $n_N = \sqrt{n+m} + a$, $n_N = \frac{n+m}{2}$, $n_N \geq \log_2 m$ etc. Here, the number of neurons in the hidden layer is a constant between 1 and 10, respectively. Enter and output the number of neurons. The model compares the results through multiple experiments, and finally selects 8 hidden neurons. The Sigmoid function is used as the transfer function, and the 3-layer BP neural network model, which is the input layer, an implicit layer and the output layer, is used to determine the BP network model.

(2) Selection of initial weights. The choice of initial weight determines BP neural network training time, complexity and optimization. The weight is too large, the weighted input and will enter the saturation region of the function, and the adjustment process will be affected. It is better that the output value of the neuron after initial weighting is close to zero. The initial weight is a number between -1 and +1 (except in special cases). Usually, calling the MATLAB toolbox existing function, we can

use newff.m to initialize the weight of the hidden layer [22].

(3) The determination of expected error and learning rate. In BP neural network, the error value is a very important parameter, if you want to find a suitable error value, you need through training comparison to get. If the error is small, you can increase the number of neurons, but the number of neurons cannot be too much, so you should increase the number of nodes to increase the training time. If the learning rate is small, the training speed is slow and takes a long time, but finally you can obtain the convergence result. If the learning rate is large, it may make the training not stable enough. Therefore, under the choice of learning rate, the learning rate is more inclined to ensure the stability of training. Usually the learning rate is between 0.01 and 0.8.

4.2.3The result of experiment

BP neural network also uses MATLAB as the experimental platform, and calls the neural network toolbox for neural network training. The core training procedures are as follows:

```
t1=clock;
net=newff(P,T,[3,1]);
datat=etime(clock,t1)
net.trainparam.goal=0.0001;
net.trainparam.epochs=5000;
net=train(net,P,T).
```

By performing the weight training of the neural network, we can obtained the weights and thresholds of the input layer and the hidden layer. After completing the training, you can follow the risk level of existing risk factors for existing experts as input, to obtain the whole system risk assessment. The risk assessment results are shown in TABLE IV. For comparison, TABLE IV also shows the prediction results of the support vector machine.

Table 3. Comparison of BP neural network and SVM prediction results

Real Data	Forecast Result		Average Absolute Error		Training Times (Seconds)	
	BP	SVM	BP	SVM	BP	SVM
0.48	0.4784	0.4805	0.0922	0.0046	0.15	0
0.56	0.5708	0.5598				
0.72	0.7198	0.7198				
0.35	0.3093	0.3489				

It can be seen from the comparison of the results of SVM information risk assessment and BP neural network information risk assessment, the support vector machine method trains less error and takes less time, so, it is very effective to use the support vector machine method for information risk assessment.

5 Conclusion

This paper constructed an information security risk assessment model based on support vector machine (SVM) algorithm, and selected and experimented the kernel function of support vector machine. Finally, compared with BP neural network model, the experimental results show that the information security risk assessment model based on support vector machine algorithm is more effective.

Acknowledgment

This work was supported by the Fundamental Research Funds for the Central Universities under grant 3132018226 and National Natural Science Foundation of China under grants 51509030 and 61803065 .

References

1. Feng Denguo, Zhang Yang, Zhang Yu qing. Overview of Information Security Risk Assessment[J]. Transactions of Communications **25**, 7 (2004), pp.10-18.
2. Wei Chengye. Information Security Risk Assessment Model [J]. Network Security Technology and Application, **4** (2002), pp.10-15.
3. Angenent S. Parabolic Equations for Curves on Surfaces Part I. Curves with p-Integrable Curvature [J]. Annals of Mathematics, **132**, 3 (1990), pp.,451-483.
4. Zhu Xinming. Information security risk assessment risk analysis method [J]. information security and technology, **8** (2010), pp.87-89.
5. Ma Ligang, Xia Jun Li. Information security risk assessment [J]. modern computer (professional version). **1** (2006), pp.49-53.
6. Xu Aiting. The application and difficulty of Delphi method [J]. China Statistics. 2006(9), pp.57-59.
7. Wen DaShun. Overview of information security risk assessment [J]. China Science and technology information, **14** (2013),81-81.
8. Shen Shikai, She yu mei. Application of fuzzy neural network in information security risk assessment [J]. computer simulation, 2011, 28(10):91-94.Grayson M A. Shortening Embedded Curves [J]. Annals of Mathematics, **129**, 1 (1989), pp. 71-111.
9. Liu Qiong. Research and design of risk assessment system based on AHP [D]. Xi'an Electronic and Science University(2009) .
10. Zhang Xuegong. Statistical learning theory and support vector machine [J]. automation Journal, **26**, 1 (2000), pp.32-42.
11. Song Y, Shen Y, Zhang G, et al. The information security risk assessment model based on GA - BP[C]// IEEE International Conference on Software Engineering and Service Science. IEEE, 2017, pp.119-122.
12. Zhao Dongmei, Liu hai feng, Liu chen guang. Information security risk assessment based on BP

neural network [J]. computer engineering and Application, **1** (2007), pp.139-141.