IJBFMR 6 (2018) 7-21 ISSN 2053-1842

International Journal of Business and Finance Management Research
www.bluepenjournals.org/ijbfmr

Using artificial intelligence technology for corporate financial diagnostics

Wei-Yuan Lin, Yuan-De Chu* and De-Yuan Liao

Department of Economics, Soochow University, No. 56, Section 1, Kueiyang Street, Chungcheng District, Taipei City 100, Taiwan.

Article History
Received 24 January, 2018
Received in revisedform 17 June, 2018
Accepted 25 June, 2018

Keywords:
Fruit fly optimization algorithm,
Financial crisis prediction,
Back propagation network,
Logit regression.

Article Type:
Full Length Research Article

ABSTRACT

The early warning of financial crisis is a great concern to scholars and experts. Economists have been trying to use various econometric models to do related research and gain many concrete achievements. At the same time, artificial intelligence experts have joined the discussion. In this paper, we will use a new technique of artificial intelligence, called the fruit fly optimization algorithm (FOA), which combines the support vector regression (SVR) and generalized regression neural network (GRNN) to create new financial crisis prediction models, with the hope that it will be useful for academia and practitioners. In addition to the traditional methods of artificial intelligence - back propagation network (BPN) and GRNN, this study also constructed the financial crisis warning models of FOAGRNN and FOASVR respectively, and compared them with the traditional Logit Regression model. The empirical results obtained show that the performance of prediction by all the financial crisis models is good, specifically, FOASVR model is the best, followed by FOAGRNN and BPN, and the last is Logit Regression. Therefore, this study has found out that the use of FOA combined with other artificial intelligence models (SVR and GRNN) is helpful to improve the performance of financial crisis early warning.

INTRODUCTION

From the existing financial distress prediction researches, it can seen that the models used for artificial intelligence system have comparatively more predictive effects than the models that used traditional econometric methods. Traditional econometric logit method has its own advantages and disadvantages that in corporate financial distress prediction, and these methods have some basic assumptions. For example, generally, their statistical methods need to set financial variables or error terms to be normal distribution, with these assumptions, the accuracy of financial distress prediction can be improved. However, according to past researches, they show that not all models can satisfy these assumptions.

Therefore, the artificial intelligence method is helpful to solve these problems. This study did not only use well known back propagation network (BPN) but also introduced a new method-fruit fly optimization algorithm (FOA), combining it with generalized regression neural network (GRNN) and support vector regression (SVR) to predict corporate financial distress. It is deeply believed that this will very helpful for enterprises and future researches.

LITERATURE REVIEW

Since Beaver (1966) used financial ratios as predictors of failure, Altman (1968) used discriminant analysis to predict corporate bankruptcy and Ohlson (1980) used probabilistic prediction of bankruptcy; a lot of researchers have used many mathematical methods and computer

*Corresponding author. E-mail: 0928chudavid@gmail.com. Tel: +886-932-397151; Fax: +886 -2-2550-4538.

techniques in this field. Altman et al. (1994), Coats and Fant (1993), Huang et al. (2004), Odom and Sharda (1990), Rumelhart et al. (1986), Specht (1990), Specht (1991), Tam and Kiang (1992), Pan et al. (2013), Wu et al. (2015), Lin and Pan (2008), Lin et al. (2011), Jan (2016), Lin (2016), Yang et al. (1999), Sun et al. (2014), Zhao and Guo (2014) and Wei et al. (2016) used neural network analysis while Doumpos and Zopounidis (2009), Drucker et al. (1997), Hua et al. (2007), Huang et al. (2004), Kim and Sohn (2010), Vapnik (1995) and Vapnik (1998) used vector machine or support vector machine regression. However, only a few scholars (Chen et al., 2013; Pan, 2012, 2014) used FOA. The contribution of this study is to combine FOA with GRNN and SVR to construct financial distress prediction models.

RESEARCH METHODS

In this study, a variety of artificial intelligence methods to build a variety of enterprise prediction model were considered. The correct prediction rates of them all with the traditional econometric method-Logit Regression-were compared. In this study, package EViews was used to establish the Logit model, and also our own Matlab programs were written to construct BPN, 2D-FOAGRNN and 3D-FOASVR.

Logit model

Regression model involving nominal scale variable is an example of a broader class of models known as qualitative response regression models. One of them is called binary response regression model and it can be estimated by the Logit Regression model (Aldrich and Nelson, 1984) as shown in (Equations 1-3).

$$L=\log\left(\frac{\pi}{1-\pi}\right)=x'\beta+u \quad (1)$$

$$\text{where } 1-\pi=P(y=0|\,x)=\frac{1}{1+\exp(x'\beta)} \quad (2)$$

$$\pi=P(y=1|\,x)=\frac{x'\beta}{1+\exp(x'\beta)} \quad (3)$$

and u denotes error term, it follows iid normal and $(\frac{\pi}{1-\pi})$ is called the odds ratio.

Backpropagation network

Back propagation (BP) is by far the most popular and widely used network-learning algorithm (Rumelhart et al., 1986). It is a more complex gradient descent algorithm than the Wirdow-Hoff learning rule, but it does essentially the same thing to build an adaptive system that minimizes an error signal (e_j) by using gradient descent.

Firstly, the learning process of BP is to set the weights of each neuron in the initial network are given by random number. Secondly, the samples to be trained are placed on the input X_i layer. After the learning process, the output value ($\hat{y}_j$) can be calculated and the difference between the output value and the training sample target value (y_j) is calculated. By this difference, the weight of each neuron in the network can be adjusted. Then we repeat the action until the network converges. The mathematical formula is shown as following (Equations 4-7).

The error function is:

$$F(e_j(t))=\frac{1}{2}\left[e_j(t)\right]^2=\frac{1}{2}\left[y_j(t)-\hat{y}_j(t)\right]^2=\frac{1}{2}\{y_j(t)-f[(U_j(t)]\}^2=\frac{1}{2}\{y_j(t)-f[\textstyle\sum_{i=1}^{m}\omega_{ij}(t)O_i(t)+\theta_j(t)]\}^2 \quad (4)$$

And the weights adjusted rule is as shown below:

$$\Delta\omega_{ij}=-\eta\frac{\partial E_j(t)}{\partial\omega_{ij}(t)}=-\eta\frac{\partial E_j(t)}{\partial e_j(t)}\times\frac{\partial e_j(t)}{\partial\hat{y}_j(t)}\times\frac{\partial\hat{y}_j(t)}{\partial U_j(t)}\times\frac{\partial U_j(t)}{\partial\omega_{ij}(t)}$$
$$=-\eta\times e_j(t)\times(-1)\times f'\left[U_j(t)\right]\times O_i(t) \quad (5)$$

Where η is called a learning rate. Let f is sigmoid function, then:

$$\Delta\omega_{ij}=\eta e_j(t)\times f\left[U_j(t)\right]\{1-f\left[U_j(t)\right]\}\times O_i(t) \quad (6)$$

Let $\delta_j(t)=e_j(t)\times f\left[U_j(t)\right]\{1-f\left[U_j(t)\right]\}$

Then $\Delta\omega_{ij}(t)=\eta\,\delta_j(t)\times O_i(t)$ (7)

In other words, the weights of BP network are adjusted by the outputs backward.

General regression neural network

Specht (1990) was the first to publish probabilistic neural network (PNN). PNN is only suitable for classification problems, but it cannot solve the problem of continuous variable. Therefore, in 1991, Specht (1991) published general regression neural network learning algorithm, it evolved from probabilistic neural networks. Besides, this algorithm not only can do classification problems, but also do dynamic model prediction and control. By the way, it has better ability to deal with linear or non-linear regression problems.

The learning process of GRNN is different from that of BP, and the weight of each neuron in the network is determined by the output and input of the sample to be trained. Some of the differences are as follows:

i. The learning process has nothing to do with the recall process.
ii. The weight of each neuron in the initial network is not necessary.
iii. The difference between the output value of the inference and the target value of the training sample is not used to modify the weight of each link in the network.
iv. No repetitive learning process.
v. The purpose of the learning process is to find the best spread parameter (σ).
vi. The number of neurons in the network is related to the training samples.

GRNN is a radial basis of the neural network. GRNN has a strong non-linear mapping ability, a flexible network structure, a high degree of fault tolerance and robustness which are suitable for solving non-linear problems. GRNN has a stronger advantage over Radial Basis Function (RBF) network in terms of approximation ability and learning speed. The network finally converges to the optimized regression surface, and the prediction effect is better when the sample data is few. In addition, the network can handle unstable data. Therefore, GRNN has been widely used in signal process, structural analysis, education industry, energy, food science, control decision system, drug design, financial field, bioengineering and other fields.

Let the joint probability density function of the random variable x and y be *f*(x, y). The conditional mean is:

$$\hat{Y} = \mathrm{E}\left({}^{y}/_{X}\right) = \frac{\int_{-\infty}^{\infty} y f(x,y)dy}{\int_{-\infty}^{\infty} f(x,y)dy} \quad (8)$$

Where $\hat{Y}$ is the predicted output of Y under the condition input vector X is given. Using the Parzen non-parametric estimation, the density function $\hat{f}$(X, y) can be estimated from the sample data set.

$$\hat{f}(\mathrm{X},\mathrm{y}) = \frac{1}{n(2\pi)^{\frac{p+1}{2}}\sigma^{p+1}} \sum_{i=1}^{n} exp$$

$$-\frac{(x-x_i)^T(x-x_i)}{2\sigma^2}\Big] * exp\Big[-\frac{(X-Y_i)^2}{2\sigma^2}\Big] \quad (9)$$

Where p is the dimension of the random variable x; the spread σ is the width coefficient of the Gaussian function, which is called the smoothing factor. In this case, x_i and Y_i are the observed values of the random variables x and y; n is the sample size. $f(X,y)$ is substituted by $\hat{f}(\mathrm{X},\mathrm{y})$ into equation, then the output $\hat{Y}(X;\sigma)$ of the available network is:

$$\hat{Y}(X;\sigma) = \frac{\sum_{i=1}^{n} Y_i exp\left[-\frac{(x-x_i)^T(x-x_i)}{2\sigma^2}\right]}{\sum_{i=1}^{n} exp\left[-\frac{(x-x_i)^T(x-x_i)}{2\sigma^2}\right]} \quad (10)$$

Let $\omega_i = \frac{exp\left[-\frac{(x-x_i)^T(x-x_i)}{2\sigma^2}\right]}{\sum_{i=1}^{n} exp\left[-\frac{(x-x_i)^T(x-x_i)}{2\sigma^2}\right]}$ (11)

Then:

$$\hat{Y}(X;\sigma) = \sum_{i=1}^{n} \omega_i Y_i \quad (12)$$

where $\sum_{i=1}^{n} \omega_i = 1$

That is, the estimated value $\hat{Y}(X;\sigma)$ is the weighted average of all the observed values Y_i, with the weighting factor ω_i.

Support vector regression

Support Vector Machine (SVM) is a machine learning system developed by Vapnik (1995), based on the statistical learning theory in 1995. It is often used in the pattern recognition, text classification and other fields. The theory is to use the linear function hypothesis space learning system in high dimensional feature space, and it belongs to a learning system from the optimization theory and structural risk minimum training algorithm.

In 1997, Drucker et al. (1997) proposed a new regression technique extended from support vector machine that is called SVR. In the view of data mining, the regression method is used to deal with the problem of prediction. By using the regression which is the use of a series of existing values to predict another continuous value, the regression functions obtained by different loss functions are also different. In the SVR, the ε denotes insensitive loss function parameter. It means that the estimate of the distance in the real value of the estimate in the allowable range of ε can ignore its error value. The technology is mainly for the known information to predict unknown variables, and it can be divided into the following two, one for the linear SVR, and the other for the non-linear SVR. Thenon-linear SVR can handle the optimization of the problem as below:

Let $x \in R^n$ and $y \in R^1$

Consider the following real function: a vector x is mapped into some a prior chosen Hibert space, where we define functions that are linear in their parameters in (Equation 13).

$$y = f(x,\omega) = \sum_{i=1}^{\infty} \omega_i \emptyset_i(x), \quad \omega = (\omega_1, \cdots, \omega_{N,\ldots}) \in \Omega \qquad (13)$$

In Vapnik (1998) the following method for estimating functions is based on the training dataset $(x_1,\omega_1), \cdots, (x_l,\omega_l)$. We want to minimize the following function (Equation 14):

$$R(\omega) = \frac{1}{\ell}\sum_{i=1}^{l} |y_i - f(x_i,\omega)|_\varepsilon + \gamma(\omega,\omega), \qquad (14)$$

where

$$|y_i - f(x_i,\omega)|_\varepsilon = \begin{cases} 0 & \text{if } |y_i - f(x_i,\omega)| < \varepsilon, \\ |y_i - f(x_i,\omega)| - \varepsilon & \text{otherwise}, \end{cases} \qquad (15)$$

(ω,ω) is the inner product of two vectors, and γ is some constant. It was shown that the function minimizing this functional has a form (Equation 16):

$$f(x,\alpha,\alpha^*) = \sum_{i=1}^{\ell}(\alpha_i^* - \alpha_i)(\Phi(x_i),\Phi(x)) + b \qquad (16)$$

where $\alpha_i^*, \alpha_i \geq 0$ with $\alpha_i^*\alpha_i$=0 and $(\Phi(x_i),\Phi(x))$ is the inner product of two elements of Hibert space.

To find the coefficients α_i^* *and* α_i, one has to solve the following quadratic optimization problem (Equation 17-18):

$$W(\alpha^*,\alpha) = -\varepsilon\sum_{i=1}^{l}(\alpha_i^* + \alpha_i) + \sum_{i=1}^{l} y_i(\alpha_i^* - \alpha_i) - \frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j)(\Phi(x_i),\Phi(x_j)) \qquad (17)$$

with constraints,

$$\sum_{i=1}^{l}(\alpha_i^* - \alpha_i) = 0, \quad 0 \leq \alpha_i^*,\alpha_i \leq C, \quad i = 1,\cdots,\ell. \qquad (18)$$

To evaluate the inner products $(\Phi(x_i),\Phi(x))$, we can use Hilbert space theory to guarantee that a symmetric function $K(u,v)$ has the following expansion:

$$K(u,v) = \sum_{k=1}^{\infty} a_k \emptyset_k(u)\, \emptyset_k(v) \qquad (19)$$

with positive coefficients $a_k > 0$, that is to guarantee that $K(u,v)$ is an inner product in some feature space Φ, it is necessary and sufficient that the conditions:

$$\int K(u,v)g(u)g(v)dudv > 0 \qquad (20)$$

be valid for any non-zero function g on the Hilbert space (Mercer's theorem). Therefore, in the SV method, one can replace (Equation 16) with

$$f(x,\alpha,\alpha^*) = \sum_{i=1}^{l}(\alpha_i^* - \alpha_i)K(x,x_i) + b \qquad (21)$$

where the inner product $(\Phi(x_i),\Phi(x))$ is defined through a kernel $K(x_i,x)$. To find coefficients α_i^* and α_i one has to maximize the function,

$$W(\alpha^*,\alpha) = -\varepsilon\sum_{i=1}^{l}(\alpha_i^* + \alpha_i) + \sum_{i=1}^{l} y_i(\alpha_i^* - \alpha_i) - \frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j)K(x_i,x_j) \qquad (22)$$

Solving (Equation 22) with constraints (Equation 18) determines the Lagrange multipliers, α^*,α and the regression is given by (Equation 13), we can find.

$$y = f(x) = \sum_{SVs}(\bar{\alpha}_i - \bar{\alpha}_i^*)K(x_i,x) + \bar{b} \qquad (23)$$

where

$$\bar{b} = -\frac{1}{2}\sum_{i=1}^{l}(\alpha_i - \alpha_i^*)(K(x_i,x_r) + K(x_i,x_s)) \qquad (24)$$

Performance of the SVR depends on its kernel functions and corresponding parameter sets. Among different kinds of the kernel functions, we choose radial basis function for its outstanding performance and relatively short operation time. The formulation of the radial basis function is as follows:

$K(x_i, x_j) = \exp(-\gamma||x_i - x_j||^2)$,

γ is a parameter that can be designed by user. Two parameters, C and γ of SVR (Xian et al., 2006) are tuned by the FOA in this paper.

Fruit fly optimization algorithm

Evolutionary algorithm is a common term for the concept of survival of the fittest and disqualification of the fittest in Darwinian evolution. Some algorithms used the above concept to actually simulate the natural evolution process, such as the earlier Professor Holland's genetic algorithm (Holland, 1975). Then researchers gradually shifted the focus of evolution to animal foraging behavior and group behavior, the earliest as ant colony algorithm (ACO) proposed by Dorigo and Gambardella (1997). Ant's body releases pheromones secretions when it moves and that can be used to find a short path, and it's the optimization. After that, Eberhart and Kenndy (1995) published particle swarm optimization (PSO) to simulate feeding behavior of birds. Using the repeated operation to find the bird which is nearest the food in its surrounding area, the way that they find food is used to solve the optimal problem. These two algorithms are developed by animal foraging and group behavior, thus, scholars call them group intelligence or group intelligence algorithm. Due to they must be through the repeated operation and search to get the optimal solution, it also belongs to the field of evolutionary algorithm.

FOA is a new algorithm created by a Taiwanese scholar (Pan, 2012). It is based on the feeding behavior of fruit fly to find optimization. The olfactory organs of fruit fly can keenly collect all kinds of smell floating in the air, then use the keen vision to find the location of food and peers gathered, and fly to that direction.

FOA is one of group intelligence, and it belongs to the field of evolutionary computation and artificial intelligence. This research method has no domain limitation in application, and it can be applied to financial, mechanical and electrical engineering, knowledge management and medical fields. It can match with other technology, such as general neural network, SVR, Bayes' theorem, decision tree and fuzzy mathematics. Besides, it is highly flexible and can be used as required in different areas, or mix with different algorithms. This study merged FOA, General Regression Neural Network and SVR to become the method of bankrupt prediction. FOA is combined with the two non-linear methods-GRNN and SVR to form the model of predictions in this study, as shown below.

FOAGRNN

This section used 2D fruit fly algorithm (Pan, 2014, 2012) to adjust the spread of general regression neural network. If the spread value is small, the radial basis function is steeper, and the vector neurons closer to the input weights will have a larger output than the other neurons, the network will react to the target vector closest to the design input vector. The network behavior is to find a weighted average between the target vectors, and the input vector of its design is the closest to the new input vector. Spread value increases gradually, more neurons contribute to average value, and then the network function will be more smooth. The FOAGRNN method can be shown as follows:

Step 1. Randomly set the location of initial fruit fly:

Number of overlapping generation: maxgen=100

Population size: sizepop=100

X_axis=rand(); Y_axis=rand() (25)

Give fruit fly random directions and distances to search for food by using olfaction:

X_i= X_axis + Random value

Y_i= Y_axis + Random value (26)

In the beginning, fruit fly does not know the location of food. First, we should estimate the distance from the origin of the two dimensions (Dist). Then we calculate the taste concentration determination value S_i which is the reciprocal of the distance.

$$\mathrm{Dist}_i = \sqrt{x_i^2 + y_i^2}\ ;\ S_i = \frac{1}{\mathrm{Dist}_i} \quad (27)$$

Step 2. The taste concentration determination value is substituted into the GRNN spread

p=S_i; net=newgrnn(tr1,t1,p) (28)

Step 3. Input the training data to get the network output value, then use the target value to calculate the sum of square error (Fitness), the value is smaller the better. Finding the fruit fly with the highest scent concentration in the population, it is to find the minimum value of sum of square error (SSE).

yc=sim(net,tr2) (29)

% difference between the network output value and the target value

```
Y=yc-t2;
for ii=1:row1
g=g+y(ii)^2                                   (30)
end.
```

Step 4. Retaining the best scent concentration value and x, y coordinates, fruit fly population fly to location by sight in the mean while.

```
[bestSmellbestIndex]=min(Smell)
Smellbest=bestSmell
X_axis=X(bestIndex)                           (31)

Y_axis=Y(bestIndex)
```

Step 5. Using the function sim to test the performance of the neural network, we set the result as yc.

yc =sim(net,tr2) (32)

And then calculate its difference as (Equation 33). % difference between net output and target output;

y=yc-t2; (33)

Step 6. Finding overlapping generation optimization, we repeat Steps 2 to 5 and determine the scent concentration is better than former generation or not. If it is better than former generation, we execute step 6.

FOASVR

The SVR model used in this empirical study is to use Pan (2012) of the "latest evolutionary computing technology – FOA". We focus on SVR of C and ε two parameters to do overlapping generation dynamic fine-tuning. In this study, the two-dimensional space of FOA is transformed into 3D space to build the bankruptcy prediction. After the establishment, the training data will be trained by this research model, and we use the test data to compare and analyze to determine whether accurate.

The steps of the hybrid method (FOASVR) are shown below:

Step 1. Input independent variables and target variable.

Step 2. Normalize the input values (adjusted to [0,1]).

Step 3. The data is divided into two parts, the data in the previous part as a training data, and the latter part as a test data.

Step 4. random initial fruit flies population position.
Number of overlapping generation maxgen=100
Population size sizepop=100
X_axis=rand();Y_axis=rand();Z_axis=rand()

Step 5. FOA results found, optimal C value and ε value of SVR, this two parameters values into the model test.

Step 6. Make the prediction result Y_hatand compute and compare their differences with the actual target value. Then do overlapping generation search as previously FOA described.

There are many types of kernel functions, the main three kinds of methods are: linear, polynomial, and RBF. This paper uses the most commonly used RBF. RBF itself is non-linear, and it can transfer data from the original space conversion to a higher dimension of space to deal with non-linear problems, and we use the FOA to adjust its parameters C and ε.

EMPIRICAL STUDY

In this study, we use our Matlab software programs to construct the FOAGRNN models, FOASVR, BP to predict. However, the traditional econometric Logit is analyzed by EViews.

Data variables and descriptive statistics

This research was focused on the default company and normal company with same industry, the same period, the same capital scale, and match them up at the ratio of 1:2. There were 120 default companies and 240 normal companies in this empirical study.

Due to the fact that operating cycle of the construction industry is longer than that of other general industries, the financial industry (including banking, securities and insurance) is different from other general industries. Therefore, these industries were excluded from this study. In this paper, 80% of the total sample was treated as training samples, 20% as test samples. The chosen ten variables are shown in Table1.

In this study, 10 input variables (X1, X2, X3, X4, X5, X6, X7, X8, X9, X10) and one output variable (y) were selected for analysis. Firstly, the input variables of data are described. Secondly, the graph of the frequency of all variables in Figure 1 are plotted. Then, the descriptive statistics is shown in Table 2.

The predicted output values of the above models are classified as 0 (default company) if they are less than or equal to 0.5; and 1 (normal company) for values greater than 0.5 to predict these models. All variables (x_i, y_i,) are normalized as shown in (Equation 34).

$$x_i' = \frac{x_i - x_{min}}{x_{max} - x_{min}} \text{ and } y_i' = \frac{y_i - y_{min}}{y_{max} - y_{min}} \quad (34)$$

The empirical results of various models

Logit model

The 10 variables of the sample data are normalized, and they are divided into two parts, the previous one 288 data as a training sample, the latter 72 data as a validation. Implementation results of training of logit model are shown in Table 3.

As shown in Table 3, the value of the LR statistic is about 194.05, and the p-value is near zero, thus the null hypothesis is rejected. Therefore from this empirical study it can be concluded that the logit method is suitable for this financial distress predicting model. Besides, another goodness of fit measure C was also used, which is defined below as performance of the prediction:

$$\text{Correct rate } C = \frac{number\ of\ correct\ predictions}{total\ number\ of\ observations} \quad (35)$$

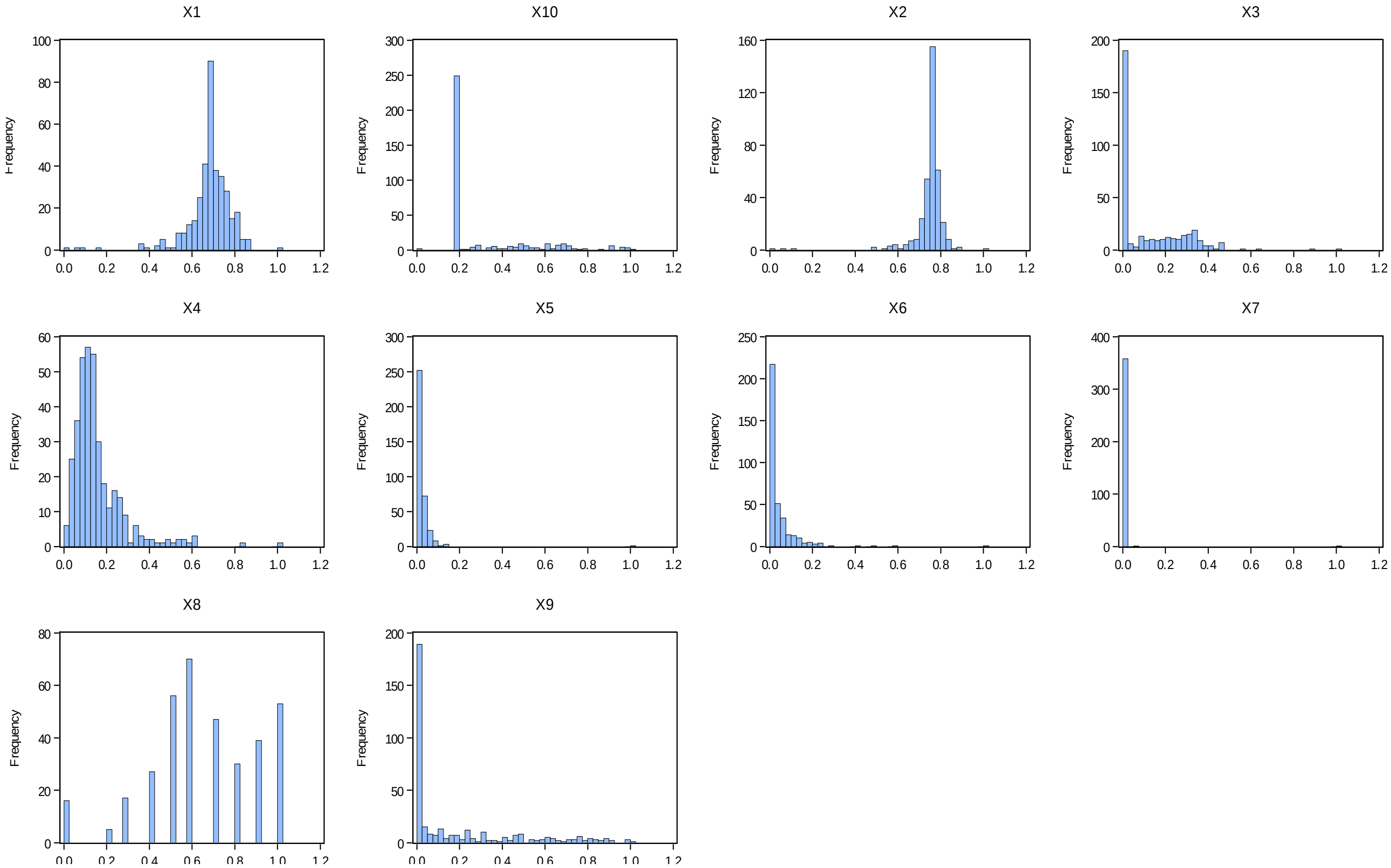


Figure 1. Frequency distribution of all variables.

Table 1. Input variables defined.

Variable types	Independent variables	Variable names
Financial variables	X1	Return on assets (ROA)
	X2	Operating profit margin (OP%)
	X3	Tax rate
	X4	Current ratio
	X5	Debt/equity (DE)
	X6	Contingent liabilities/equity
	X7	Net operating cycle
	X8	Taiwan Corporate Credit Risk Index (TCRI) credit rating
External rating variable	X9	Equity Pledge Ratio of Directors and Supervisors
Corporate governance variables	X10	Director Compensation/Net Icome

Source: Taiwan Economic News (TEJ).

Table 2. Statistics of all input variables.

Independent variables	No. of obs	Mean	Median	Max	Min	Std. Dev.	Skewness	Kurtosis
X1	360	0.681819	0.688838	1	0	0.105201	-2.51131	15.25314
X2	360	0.748735	0.761459	1	0	0.081193	-5.49372	46.44414

Table 2. Contd.

X3	360	0.121021	0.000345	1	0	0.158887	1.428503	5.949635
X4	360	0.156998	0.125605	1	0	0.122235	2.697684	13.66468
X5	360	0.023554	0.014265	1	0	0.056391	14.59782	251.3322
X6	360	0.040857	0.008481	1	0	0.084066	5.920434	56.20882
X7	360	0.014981	0.011793	1	0	0.05213	18.81855	356.0669
X8	360	0.644167	0.6	1	0	0.25139	-0.49374	3.037733
X9	360	0.181425	0.0052	1	0	0.271497	1.431511	3.804831
X10	360	0.300225	0.18018	1	0	0.215845	1.640116	4.603797

Table 3. Empirical results of logit model.

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C: Constant term	-4.53096	4.066233	-1.11429	0.2652
X1:Return on assets (ROA)	7.862751	4.917823	1.598827	0.1099
X2:Operating Profit Margin (OP%)	5.031582	7.107448	0.707931	0.4790
X3:Tax rate	-0.41101	1.652558	-0.24871	0.8036
X4: Current ratio	0.960073	2.556911	0.375482	0.7073
X5:Debt/equity(DE)	-21.20600	13.82218	-1.5342	0.1250
X6:Contingent liabilities/equity	0.639459	2.846221	0.224669	0.8222
X7:Net operating cycle	31.32657	63.57159	0.492776	0.6222
X8:TCRI credit rating	-5.75802	1.514086	-3.80297	0.0001
X9:Equity pledge ratio of directors and supervisors	-2.07294	0.740476	-2.79947	0.0051
X10:Director compensation/net Income	2.813738	1.943072	1.448088	0.1476
McFadden R-squared		0.529293		
S.D. dependent var		0.472225		
Akaike info criterion		0.675612		
Schwarz criterion		0.815517		
Hannan-Quinn criter.		0.731678		
LR statistic		194.055800		
Prob(LR statistic)		0		
Obs with Dep=0		96		
Obs with Dep=1		192		
Mean dependent var		0.666667		
S.E. of regression		0.296929		
Sum squared resid		24.422290		
Log likelihood		-86.288180		
Restr. log likelihood		-183.31610		
Avg. log likelihood		-0.299612		
Total obs		288		

Performance of GRNN by random search

First, we take a random method, select the threshold (spread) of GRNN to conduct empirical research. The results are shown in Table 4 and Figure 2 as follows: from our empirical results, we found that the threshold value and the RMSE predicted are displayed in the same direction. In other words, the threshold value is smaller and the predicted result is better. In this paper, we tested the FOA training threshold value of 0.0139 and the RMSE of 0.2013, its value is clearly small. This finding has shown that our FOA can be used to find the optimization

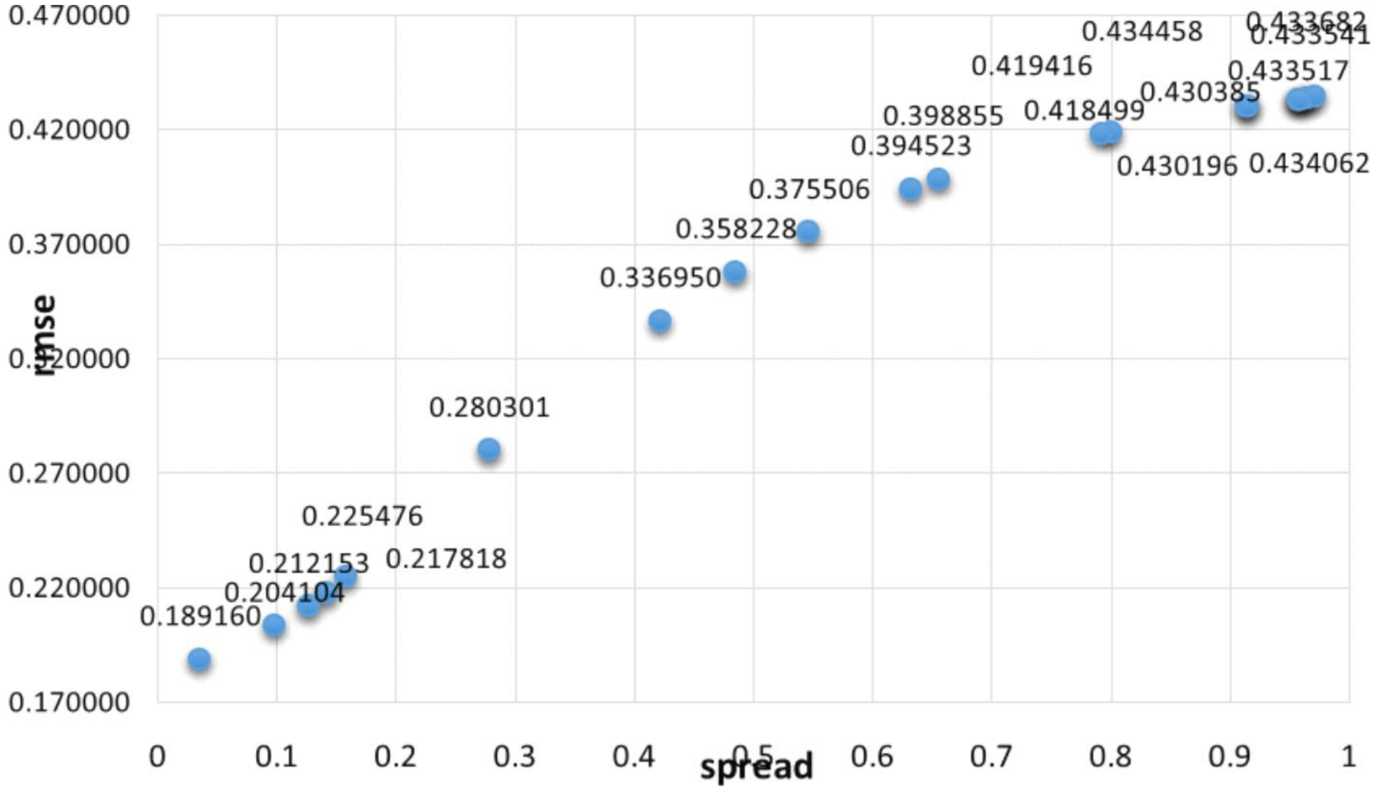


Figure 2. Performance of GRNN by random search.

Table 4. Performance of GRNN by random search.

Spread	RMSE	Spread	RMSE
0.126987	0.212153	0.957167	0.433517
0.913376	0.430196	0.485376	0.358228
0.632359	0.394523	0.80028	0.419416
0.09754	0.204104	0.141886	0.217818
0.278498	0.280301	0.421761	0.336950
0.546882	0.375506	0.915736	0.430385
0.957507	0.433541	0.792207	0.418499
0.964889	0.434062	0.959492	0.433682
0.157613	0.225476	0.655741	0.398855
0.970593	0.434458	035712	0.189160

predictor parameter of GRNN.

FOAGRNN

In the process of 20 times FOAGRNN repeated tests, one of the fruit fly flying route and RMSE is shown in Figure 3a and b. In Figure 3b, after about 10 iterations, the effect of convergence can be reached. The best RMSE value (0.012) and spread value (0.0139) of training data are obtained. And the best RMSE of testing data can also be calculated as 0.2013.

FOASVR

Similarly, after the 20 times of FOASVR experiments, the fruit fly flying route and RMSE are shown in Figures 4a

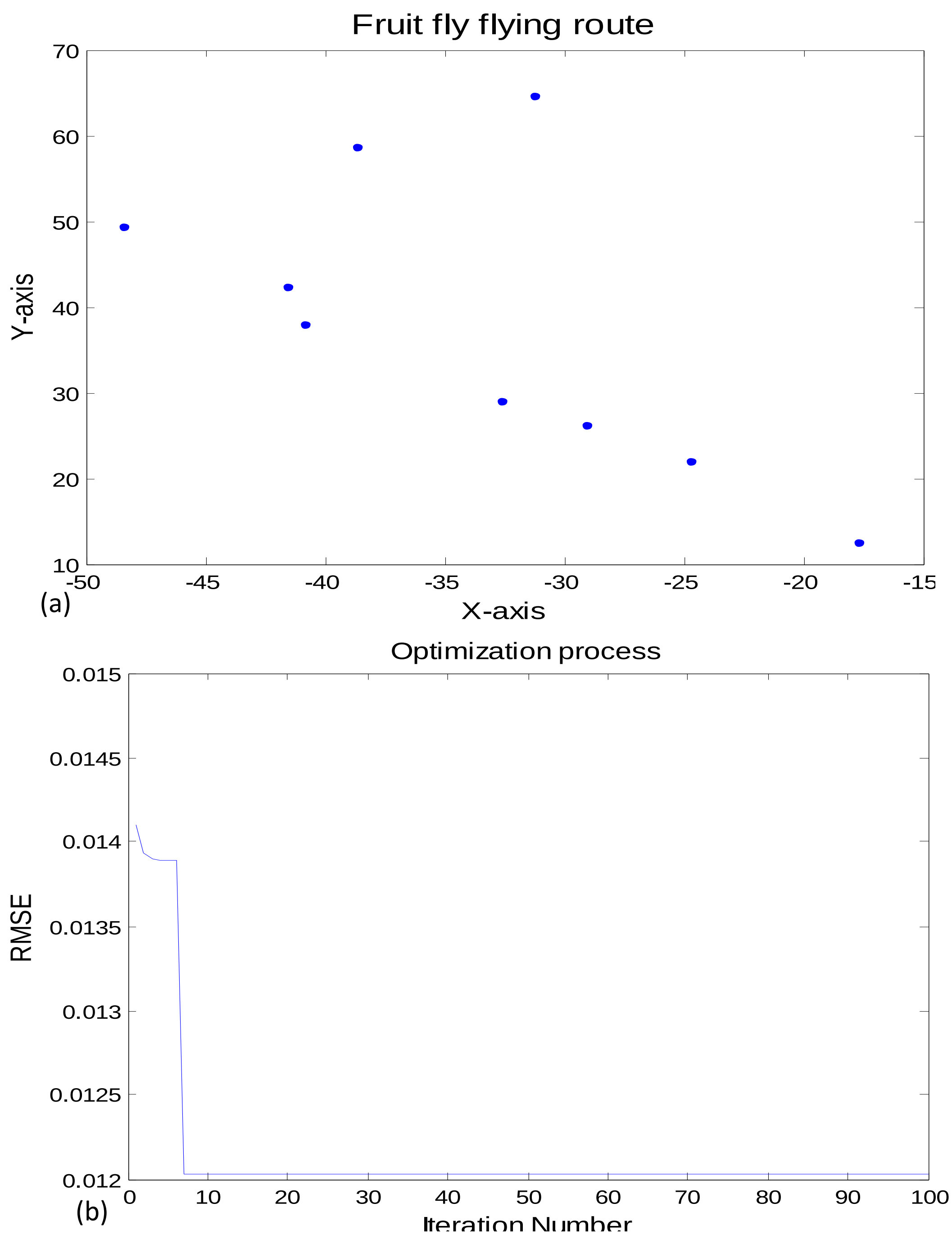


Figure 3. The Evolution path of FOAGRNN. **a,** The Fruit Fly 2D – route; **b,** the RMSE of the best fruit fly over time.

and b. After about 10 iterations, the effect of convergence can be reached in Figure 4b. The best RMSE value (0.0247) and SVR Parameters (Cbest =0.2742, ebest =0.0286) of training can be obtained. And the best testing RMSE can also be found as 0.2518. The operation screen of FOASVR is shown in Figure 5.

BPN

The BPN implementation of Matlab software is also shown in Figure 6. This experiment found that: the prediction results of BPN and its type I and type II error rates are shown in Table 5. The prediction accuracy (C)

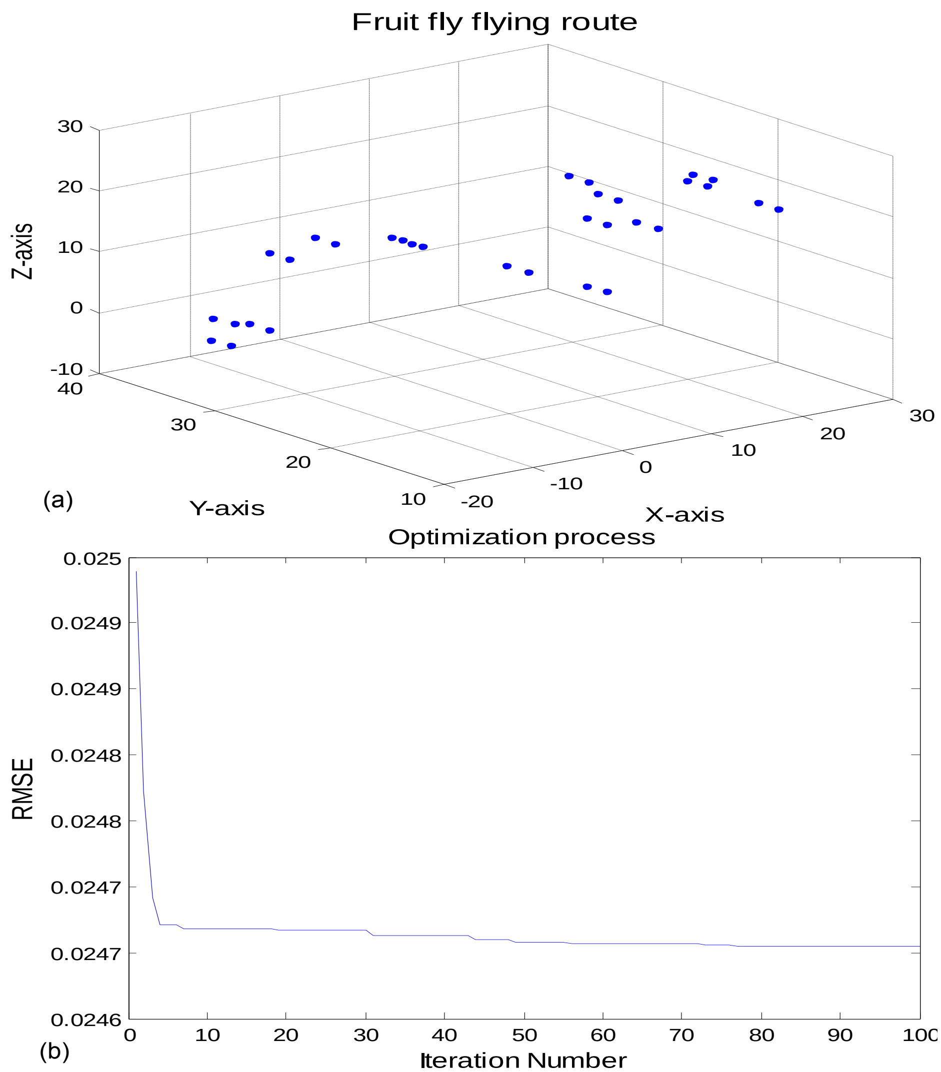
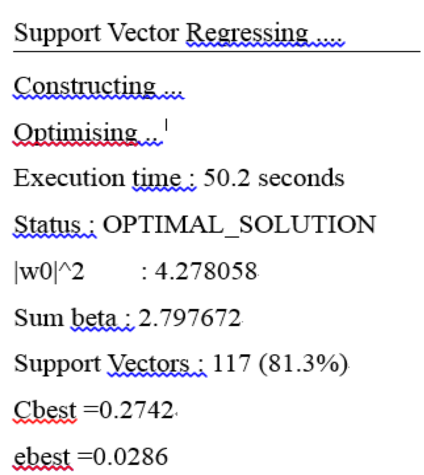


Figure 4. The Evolution path of 3D-FOASVR.**a,** the fruit fly 3D – route; **b,** the RMSE of the best fruit fly over time.


Support Vector Regressing ....

Constructing ...

Optimising ...

Execution time : 50.2 seconds

Status : OPTIMAL_SOLUTION

|w0|^2 : 4.278058

Sum beta : 2.797672

Support Vectors : 117 (81.3%)

Cbest =0.2742

ebest =0.0286

Figure 5. Operation screen of FOASVR.

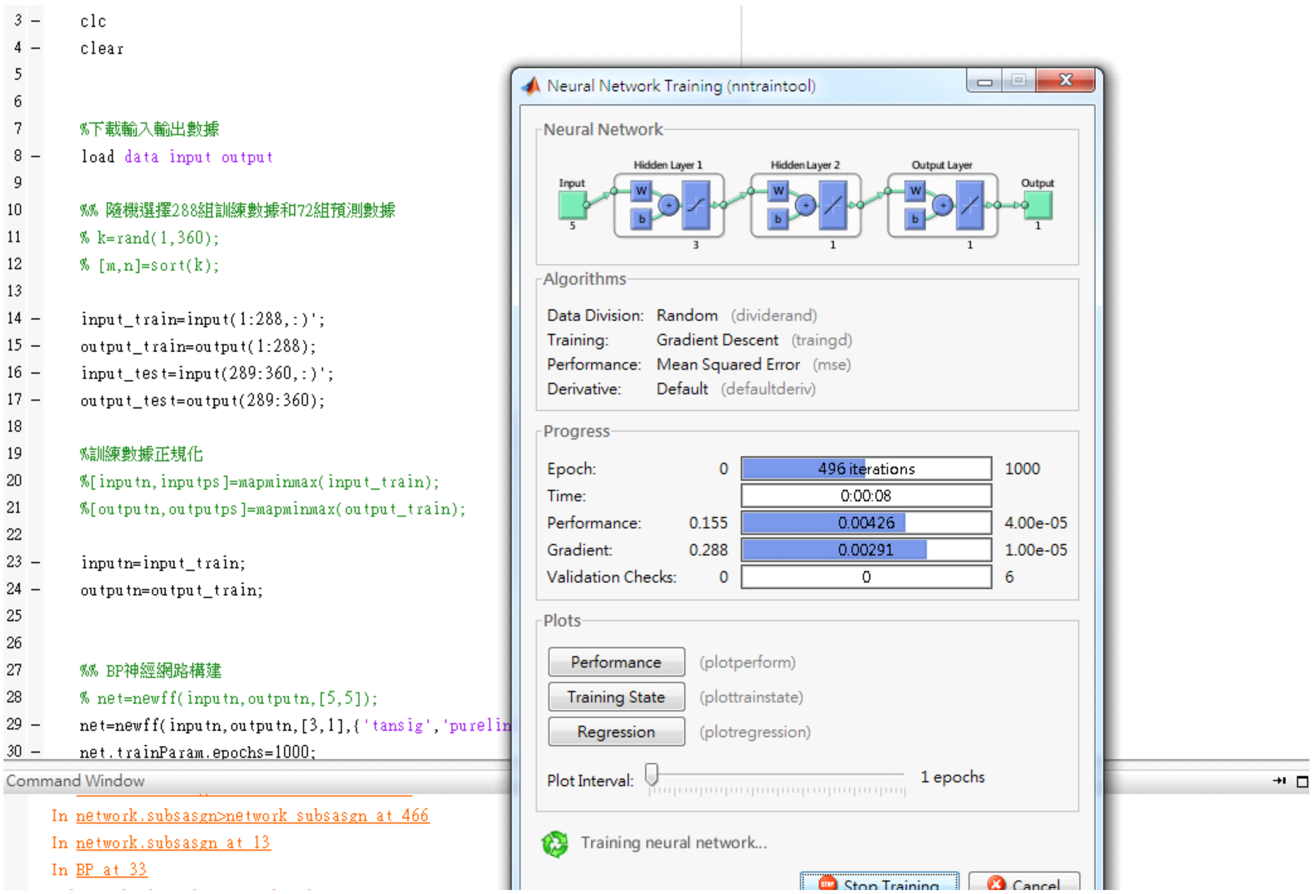


Figure 6. BPN model in Matlab screen.

Table 5. Performance of BPN.

Accuracy No.	Outside sample test of BPN				
	Accuracy rate	Type I errors no.	Type II errors no.	Type I error rate	Type II error rate
67	93%	3	2	0.042	0.028

of it is 93%, the number of type I errors is 3, type II errors is 2. And the error rate of type I and the type II are shown as0.042 and 0.028 respectively. The type II error rate is obviously smaller than the type I error rate in Table 5. At this point, we modify some of the default training parameters as follow:

net=newff(inputn,outputn,[3,1],{'tansig','purelin'}, 'traingd');
net.trainParam.epochs=1000;
net.trainParam.lr=0.1;
net.trainParam.goal=0.00004;
net.trainParam.lr_inc=1.05;

Performance comparison of models

The performance comparisons of all models are listed in Table 6. They are classified as three parts, Type I, Type II, and ROC curves as shown in Table 6.

Type I error: This paper further defines the performance and forecasting error rate by the prediction error, that is, Error rate = type I error + type Ⅱ error. The prediction error of each model is shown in Table 6. We found that type I error rate of FOAGRNN and FOASVR is equal to (1.4%) which is the lowest in these four kinds of model. And the other two models-BPN and logit are bigger

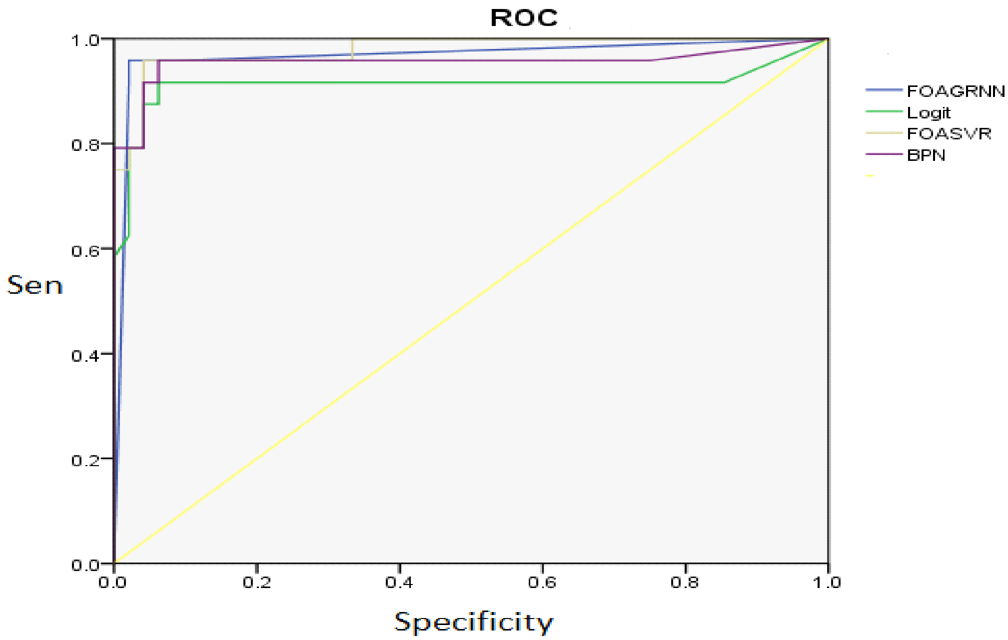


Figure 7. The ROC curves of the four models.

Table 6. Performance of these models.

Performance				
Data	Models			
	Logit	FOAGRNN	FOASVR	BPN
Training no.	288	288	288	288
Testing no.	72	72	72	72
Correct no.	66	68	69	67
Accuracy rate	92%	94%	96%	93%
Type I error	4.2%	1.4%	1.4%	4.2%
Type II error	4.2%	4.2%	2.8%	2.8%

(4.2%).

Type II error: The prediction type II error of each model is also shown below in Table 6. The experimental results show that the FOASVR and BPN are the lowest, both of which are (4.2%). However, logit and FOAGRNN is slightly larger (2.8%).

ROC curves: Bradley (1997) and Hand Hand and Till (2001) have pointed out that the area under the benchmark line and ROC curve is larger, the model classification ability is more accurate. It can be seen from Figure 7, the prediction abilities of FOASVR and FOAGRNN financial distress models are better than those of the other two models-BP and logit. However, it cannot be seen which of FOAGRNN or FOASVR model is more accurate from Figure 6. Thus, the need for further analysis by the numerical values in Table 7.

The area under the ROC curve (AUC)is 97.8% for FOASVR, 96.7% for FOAGRNN, 95.6% for BPN, 91.3% for Logit respectively. The Gini coefficient is 95.6% for FOASRR, 93.4% for FOAGRNN, 91.2% for BPN, and 82.6% for logit respectively. It can be seen, both AUC and Gini coefficient values of FOASVR are greater than those of FOAGRNN.

The above analysis shows that both FOASVR and FOAGRNN models have the better ability to predict the enterprise's crisis, BPN model is second, and the Logit model is the last. From another point of view, the predictive ability of the model with artificial intelligence is more accurate than that of the traditional logit model, even the prediction power of all models are still good

Table 7. Performance of our predictive models.

Models	Empirical results				
	(%)			95% confidence interval	
	Gini coef.	AUC	Standard error	Low bound	Upper bound
FOAGRNN	93.4	96.7	2.7	91.3	1.0
Logit	82.6	91.3	5.3	81.0	1.0
FOASVR	95.6	97.8	1.6	94.8	1.0
BPN	91.2	95.6	3.6	88.5	1.0

enough (It is because we used RBF types as our functional forms in this paper).

Conclusion

The main contribution of this study is to combine the FOA with GRNN and SVR to construct the financial distress prediction model. GRNN and SVR model combined with FOA are helpful to enhance the performance of the corporate financial distress. The empirical results show that:

i. Comparison of the four methods found that FOASVR had the best predictive ability, the second FOAGRNN and BPN, and the last is Logit regression.
ii. Overall, FOASVR has a slightly better prediction accuracy and less error rate than the other three methods
iii. Through the ROC curve analysis results, FOASVR is still the best.

In conclusion, the three models that used artificial intelligence have better predictive performance than traditional logit model.

REFERENCES

Aldrich J. H. & Nelson F. (1984). Linear probability, logit and probit models, Sage Publications.

Altman E. I. (1968). Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. J. Financ. 23:589-609.

Altman E. I., Marco G. F. (1994). Varetto, corporate distress diagnosis: comparisons using linear discriminant analysis and neural networks. J. Bank Financ. 18:505-529.

Beaver W. H. (1966). Financial ratios as predictors of failure. J. Account. Res. Pp. 71-111.

Bradley A. P. (1997). The use of the area under the ROC curve in the evaluation of machine learning algorithms. Pattern Recogn. 30:1145-1159.

Chen P. W., Lin W. Y., Huang T. H. & Pan W. T. (2013). Using fruit fly optimization algorithm optimized grey model neural network to perform satisfaction analysis for E-business service. Appl. Math. Inform. Sci. 7:459-465.

Coats P. K. & Fant L. F. (1993). Recognizing financial distress patterns using a neural network tool. Financ. Manage. 22:142-155.

Dorigo M. & Gambardella L. M. (1997). Ant colony system: a cooperative learning approach to the traveling salesman problem. IEEE Trans. Evolut. Comput.1:53-66.

Doumpos M. & Zopounidis C. (2009). Monotonic support vector machines for credit risk rating. New mathematics and natural computation. 5:557-570.

Drucker H., Burges C. J., Kaufman L., Smola A. & Vapnik V. (1997). Support vector regression machines. Adv. Neural Inform. Process. Syst. 9:155-161.

Eberhart R. C. & Kenndy J. (1995). A new optimizer using particles warm theory, in: Proceedings of the sixth international symposium on micro-machine and human science, Nagoya, Japan. Pp. 39-43.

Hand D. J. & Till R. J. (2001). A simple generalization of the area under the ROC curve to multiple class classification problems. Machine Learning. 45(2):171-186.

Holland J. (1975).Adaptation in natural and artificial systems. University of Michigan Press, Ann Arbor, MI.

Hua Z. S., Wang Y., Xu X. Y., Zhang B. & Liang L. (2007). Predicting corporate financial distress based on integration of support vector machine and logistic regression. Expert Systems with Applications 33:434-440.

Huang Z., Chen H., Hsu C. J., Chen W. H. & Wu S. (2004). Credit rating analysis with support vector machines and neural network: A market comparative study. Decision Support Systems 37:543-558.

K. Y. Tam & Kiang M. (1992). Managerial applications of neural networks: The case of bank failure predictions. Manage Science. 38:926-947.

Kim H. S. & Sohn S. Y. (2010). Support vector machines for default prediction of SMEs based on technology credit. Eur. J. Oper. Res. 201:838-846.

Lin W. Y. & Pan W. T. (2008). The study of impact from the hybrid model combining ZSCORE and GAANFIS on business performance. J. Stat. Manage. Syst. 11:1195-1206.

Lin W. Y. (2016). A novel 3D fruit fly optimization algorithm and its applications in economics. Neural Comput. Appl. 27:1391-1413.

Lin W. Y., Mei K. C. & Lien C. C. (2011). Application of grey model neural network in performing financial characteristic study of enterprises in Taiwan. J. Stat. Manage. Syst. 14:751-766.

Odom M. D. & Sharda R. (1990). A neural network model for bankruptcy prediction. IJCNN International Joint Conference 2:163-168.

Ohlson J. A. (1980). Financial ratios and the probabilistic prediction of bankruptcy. J. Account. Res. 18:109-131.

Pan W. T. (2012). A new fruit fly optimization algorithm: Taking the financial distress model as an example. Knowl. Based Syst. 26:69-74.

Pan W. T. (2014). A new evolutionary computation - Fruit fly optimization algorithm, Second Edition, Taiwan.

Pan W. T., Su S. C., Kao S. L., Lin W. Y.& Huang T. H. (2013). Construction of enterprise operation performance prediction model across Taiwan strait using artificial intelligence technique. JDCTA. 7:929-938.

Rumelhart D. E., Hinton G. E. &Willams R. J. (1986). Learning representations by back propagating errors. Nature 323:533-536.

Specht D. F. (1990).Probabilistic neural networks and the polynomial adaline as complementary techniques for classification. IEEE Trans on Neural Networks.1: 111-121.

Specht D. F. (1991). A general regression neural network. IEEE Trans. Neural Networks. 2:568-576.

Sun Y., Lang M., Wang D., Liu L. (2014). A PSO-GRNN model for railway freight volume prediction: Empirical study from China Journal of Industrial Engineering and Management, May 2014.

Vapnik V. N. (1995). The nature of statistical learning theory, Springer, New York.

Vapnik V. N. (1998). Statistical learning theory, Springer, New York.

Wei W., Jiang J., Liang H., Gao L., Liang B., Huang J., Zang N., Liao Y., Yu J., Lai J., Qin F., Su J., Ye L. & Chen H. (2016). Application of a combined model with autoregressive integrated moving average (ARIMA) and generalized regression neural network (GRNN) in forecasting hepatitis incidence in Heng County, China, Jun, 2016.

Wiszniowski J. (2016). Applying the general regression neural network to ground motion prediction equations of induced events in the Legnica-Głogów Copper District in Poland, ActaGeophysica, Dec 2016.

Wu J., Peng D., Li Z., Zhao L. & Ling H. (2015). Network intrusion detection based on a general regression neural network optimized by an improved artificial immune algorithm, Mar 2015.

Xian W., Ao L., Zhaohui J. & Huanqing F. (2006). Missing value estimation for DNA microarray gene expression data by support vector regression imputation and orthogonal coding scheme. BMC Bioinformatics.

Yang Z. R., Platt M. B. & Platt H. D. (1999). Probabilistic neural networks in bankruptcy prediction. J. Bus. Res. 44:67-74.

Zhao H. & Guo S. (2014). Annual energy consumption Forecasting based on PSOCA-GRNN Model.