A Venture Capital Assessment Model based on the Genetic Neural Network

Tieying Liu, Yuting Fan, Xinzhu Xu
Inner Mongolia university institute of compute, Hohhot, China
cslty@imu.edu.cn, rubyFanYT@gmail.com

Abstract—With the rapid development of economy, there are greater challenges in Venture Capital. In the face of the venture project selecting, scientific evaluation of investment venture is an important issue in venture investment, and it directly affects the success or failure of investment. In this paper, we analyzed the venture evaluation index system, and apply a genetic algorithm (GA) to the processes of the neural network construction and training, and then we set up an evaluation model of the genetic algorithm and artificial neural network. In order to solve the problems of venture-benefit assessment in the venture investment, we presented a method which is under the order of solving the problems of venture-benefit assessment in the model of the genetic algorithm and artificial neural network. In the construction and training, and then we set up an evaluation algorithm (GA) to the processes of the neural network. The theoretical results show that the proposed model can effectively improve evaluation efficiency, and increase the rationality and correctness of venture evaluation system.

Keywords—Venture capital, Venture evaluation, Benefit valuation, Index system

I. INTRODUCTION

As a new financial industry, venture capital–with the unique power—impacts on the global economic development [1-3]. It is an organic process which composed by links such as raising funds, screening project, implementing investment and out of investment. In financial markets, Venture Capital (VC) has the characteristics of “high-risk”, and investors take high-risks with the purpose of gaining “high-return”, but not all of the venture investment projects can obtain high returns. The success of an investment project largely depends on the reliability of evaluation results. Venture investors must choose a potential projects in many investment projects as an investment. Thus venture capital project evaluation is one of key factors deciding whether the investment is successful. Moreover, it is also one of the important research contents of venture capital.

Over the last few decades, many of risk evaluation approaches have been proposed widely [4-7], most of them have been affected by random factors deeply, and their evaluation results are subjective [5]. The artificial neural network evaluation method can effectively overcome these defects, so it was widely accepted. However, when we use the BP algorithm to train neural network, there have three disadvantages: slow convergence speed of learning process; falling into local optimal solution easily, and lack of robustness that network performance is more sensitive to its initial setting [8]. To solve these problems, the paper provides an evaluation model which combines the genetic algorithm and the BP neural network.

In this paper, the genetic algorithm neural network evaluation model (GA-BP) can respectively estimate the venture and benefit of the investment project, and then get comprehensive evaluation results.

II. EVALUATION MODEL

In the process of combining the genetic algorithm (GA) and the neural network, as a subject model, the neural network implements evaluation function, the Genetic algorithm is auxiliary, and implements the optimization of the neural network. On the basis of existing models and algorithms, we develop a BP neural network, and apply the pointed and improved GA to the network training, and then we establish a venture assessment model and related algorithms.

A. Improved GA-BP algorithm

• According to the index system and assessment level, the import and export cell have been ascertained. Further defining concealed layer node list, the BP network of three layers have been set up;
• Using the GA to search the BP neural network weights which accord with fitness function
• Taking optimizing weights which is searched by the GA as initial weights
• Training the network by using the BP algorithm
• Judging the Network performance
• Using the GA to optimize the network weights again and taking the result as new initial weights, then return to step 4;
• Repeat these steps until the network is mature;
• Collecting the data of actual evaluation object and valuating it using the mature network after training

B. The structure of the neural network and evaluation method

We built a network of three layers, G(m,n,s;W), where m is the number of input nodes, n the number of output nodes, s the number of concealed layer nodes, and w a weight set.

Benefit and venture are both exist in venture capital. There is relevance between the benefit and venture in venture capital project. The key of venture capital evaluation is how to balance the benefit and risk. In this paper, venture capital project evaluation need evaluate the risk and benefit separately, and get the evaluation results of whole venture capital project by integrating the risk and benefit index.

On the basis of \( m \) benefit indexes and \( n \) risk indexes, the \( kth \) evaluate index weight that after training is \( \theta_k \ (k = 1,2,\ldots,M + N) \) the score is \( P_i \) which depends on the characteristics of the index, and the quantified benefit and risk indexes scores of the \( j \)th invest scheme are \( I_{j / k}, R_{j / k} \ (j = 1,2,\ldots,L) \) respectively in \( L \) candidate venture capital scheme.
Among this, 
\[ I_j = \sum_{k=1}^{M} P_k \theta_k, \quad R_j = \sum_{k=1}^{N} P_k \theta_k. \]

At last, according to the investment appetites of investors, to entrust the benefit and risk the appetite weight \( W_{i,j} \) and \( W_{R,i} \). Among this, 
\[ W_{k,j} = 1 - W_{i,j}. \]

Then the last comprehensive evaluate score of the \( jth \) venture scheme is: 
\[ V_j = W_{i,j} \sum_{k=1}^{M} P_k \theta_k + W_{R,i} \sum_{k=1}^{N} P_k \theta_k. \]

C. The node of the hidden layer in the neural network

A method that can determine the number of the node in the hidden layer has been presented in Ref. [9]. The number can be calculated using the following formula in the method:
\[ s = \sqrt{n + m + a}, a \in (1,10), \]
In our method, we made some improvement of it:
\[ N_0 = \sqrt{n + m}. \]

Given:
\[ N_0 = \sqrt{n + m}, \]
we add one node to gain \( N_1 \). Then we choose the same sample for training, and use the same node that is not tested to test at the same time.

Our test data supports the following conclusions:
\[ \bullet \text{If } R_j, \text{ the correct recognition rate of } N_1 \text{ greater than } N_0, \text{ then } N_j \text{ adds one node until } R_{(j+1)} < R_k, \text{ and find } s = \sqrt{n + m + k}. \]
\[ \bullet \text{If } R_j, \text{ the correct recognition rate of } N_1 \text{ less than } N_0, \text{ then } N_j \text{ shrinks by one until } R_{(j-k+1)} > R_{(N_0-k)}, \text{ and find } s = \sqrt{n + m} - (k + 1). \]
\[ \bullet \text{If } R_j, \text{ the correct recognition rate of } N_1 \text{ equals } N_0, \text{ then } s = \sqrt{n + m}. \]

D. The encoding of the genetic algorithm (GA) and definition of operator

1) The genetic algorithm is coding to neural network

Given:
\[ \alpha_1, \alpha_2, \ldots, \alpha_s, w_{1,1}, w_{1,2}, \ldots, w_{1,s}, b_1, \ldots, w_{2,s}, \ldots, w_{s,1}, w_{s,2}, \ldots, w_{s,s}, b_m, \ldots, w_{m,1}, b_{m+1}, \ldots, \]
where, \( \alpha_i \in \{0,1\} \), \( w_{k,j} \) is the link weight from unit \( AN_{k-1} \) to \( AN_{k,j} \) is the threshold of the \( jth \) unit in the output layer. \( b_i \) is the threshold of the \( ith \) unit in the hidden layer,
\[ i \in \{1,2,\ldots,n\}, \quad j \in \{1,2,\ldots,s\}. \]

2) The process of optimization adopt the adaptive variable coding

In the process of operation, we change the definition interval appropriately according to the excellent individual variable scale. And we adjusted the contractive amplitude of the interval for contracting the sliding interval close to the optimal value, and then improved the accuracy and efficiency of searching.

We have coded it by adaptive variable as follows:
\[ I_{\text{min}} = I_{\text{max}} + \frac{I_{\text{min}} - I_{\text{max}}}{D} \]
\[ I'_{\text{max}} = I_{\text{max}} + \frac{I_{\text{max}} - I_{\text{min}}}{D} \]
where \( I_{\text{min}}, I_{\text{max}} \) is the sliding interval of an essential variable before coding; \( I'_{\text{min}}, I'_{\text{max}} \) is the sliding interval of an essential variable after coding; \( v_{\text{min}}, v_{\text{max}} \) are the minimum and maximum of the essential variable in the sub-population; \( D \) is the contraction parameters. \( D \in (2,4) \), Generally, \( D \) is equal to 4 in the beginning of the algorithm. We reduced the value of \( D \) appropriately to expand the search range for increased speed, when the population is contracted gradually close to the optimal value.

a) The definition of the fitness function
\[ E = \sum_{p=1}^{l} \left( \frac{1}{2} \sum_{i=1}^{n} (y_{pi} - o_{pi}) \right)^2 \]
where \( y_{pi} \) and \( o_{pi} \) are respectively the desired output and the actual output of the \( pth \) sample of the \( ith \) output unit in the neural network. \( t \) is the number of sample, and \( n \) the number of output unit.

The fitness function is defined as follows:
\[ f = \frac{1}{E} (1 + e^{s+\alpha(1-N)}) \]
where \( N_0 \) is the number of 1 in the controlling gene, i.e. such as the BP net above, \( N_0 = \sum_{i=1}^{N} a_i \),
\[ N_{\text{max}} \] is the max number of the node in the hidden layer, i.e. \( N_{\text{max}} = s \); changed with the practical situation, \( \alpha \) is the weighting parameter of the node in the hidden layer.

b) The genetic algorithm's crossover and mutation
rate and the choice of operators
The values of crossover and mutation rate are as follows:

\[ p_c = \begin{cases} 
   k_c \left( f_{\text{max}} - f_c \right) / \left( f_{\text{max}} - f_{\text{avg}} \right), & f_c \geq f_{\text{avg}} \\
   k_c, & f_c < f_{\text{avg}} 
\end{cases} \] (2.1)

\[ p_m = \begin{cases} 
   k_m \left( f_{\text{max}} - f_i \right) / \left( f_{\text{max}} - f_{\text{avg}} \right), & f_i \geq f_{\text{avg}} \\
   k_m, & f_i < f_{\text{avg}} 
\end{cases} \] (2.2)

where \( k_c \) and \( k_m \) are the constants that less than 1, \( f_c \) is the larger fitness value of the two individuals which will be crossed, \( f_i \) is the fitness of individual to mutate, \( f_{\text{max}} \) is the maximum colony adaptation degree in the contemporary population, and \( f_{\text{avg}} \) is the average colony adaptation degree in the contemporary population. We utilize the above conventional copy operator, crossover and mutation operator to constitute our algorithm.

## III. EXAMPLE ANALYSES

With the need of risk investment evaluation modeling, this paper uses stratified sampling. Based on invested project, the expert group graded it and got these data in tables 3.1 and 3.2.

### A. Risk evaluation training result of risk investment.

The network structure of risk evaluation model architecture is 6-3-1 type which includes 6 input nodes, 3 hidden layer nodes and 1 output value.

The realize process goes as follows,
- The GA creates 60 populations,
- Get the global optimum network value after 500 gaps of evolution,
- After 40899 trainings of the BP algorithm, the GA-BP algorithm ends.

Separately get this 10 items’ risk degree by using the above model to calculate the risk sample data, and the result is identical with expert’s. The result is shown in table 3.3

### B. Benefit assessing model training result of risk investment

The network structure of benefit assessing system model is 10-5-1 type which includes 10 input nodes, 5 hidden layer nodes and 1 output value.

The detail procedure is as follows,
- The GA creates 60 initial population,
- Get the global optimum network value after 500 searches for global optimization,
- After 4096 optimizing trainings of the BP algorithm, the GA-BP algorithm ends.

Separately get this 10 items’ benefit degree by using the above model to simulate the benefit sample data, and the result is identical with expert’s. The result is shown in table 3.4.

### Table 3.3 The output values of training sample in this GA-BP risk evaluation model and the evaluate results of experts

<table>
<thead>
<tr>
<th>No.</th>
<th>Expert evaluation</th>
<th>Actual outcome</th>
<th>Expert evaluation</th>
<th>Actual outcome</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.65</td>
<td>0.64982</td>
<td>0.63</td>
<td>0.58746</td>
</tr>
<tr>
<td>2</td>
<td>0.61</td>
<td>0.60615</td>
<td>0.61</td>
<td>0.50842</td>
</tr>
<tr>
<td>3</td>
<td>0.82</td>
<td>0.82418</td>
<td>0.82</td>
<td>0.50842</td>
</tr>
<tr>
<td>4</td>
<td>0.75</td>
<td>0.51463</td>
<td>0.75</td>
<td>0.50842</td>
</tr>
<tr>
<td>5</td>
<td>0.51</td>
<td>0.51463</td>
<td>0.51</td>
<td>0.50842</td>
</tr>
<tr>
<td>6</td>
<td>0.50</td>
<td>0.64241</td>
<td>0.50</td>
<td>0.64241</td>
</tr>
<tr>
<td>7</td>
<td>0.49</td>
<td>0.5085</td>
<td>0.49</td>
<td>0.5085</td>
</tr>
<tr>
<td>8</td>
<td>0.47</td>
<td>0.50842</td>
<td>0.47</td>
<td>0.50842</td>
</tr>
<tr>
<td>9</td>
<td>0.46</td>
<td>0.4974</td>
<td>0.46</td>
<td>0.4974</td>
</tr>
<tr>
<td>10</td>
<td>0.45</td>
<td>0.4974</td>
<td>0.45</td>
<td>0.4974</td>
</tr>
</tbody>
</table>

### Table 3.4 The output values of training sample in this GA-BP benefit evaluation model and the evaluate results of experts

<table>
<thead>
<tr>
<th>No.</th>
<th>Expert evaluation</th>
<th>Actual outcome</th>
<th>Expert evaluation</th>
<th>Actual outcome</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.5686</td>
<td>0.57954</td>
<td>0.5686</td>
<td>0.57954</td>
</tr>
<tr>
<td>2</td>
<td>0.5438</td>
<td>0.52531</td>
<td>0.5438</td>
<td>0.52531</td>
</tr>
<tr>
<td>3</td>
<td>0.4974</td>
<td>0.51456</td>
<td>0.4974</td>
<td>0.51456</td>
</tr>
<tr>
<td>4</td>
<td>0.5571</td>
<td>0.51456</td>
<td>0.5571</td>
<td>0.51456</td>
</tr>
<tr>
<td>5</td>
<td>0.50842</td>
<td>0.51456</td>
<td>0.50842</td>
<td>0.51456</td>
</tr>
<tr>
<td>6</td>
<td>0.5085</td>
<td>0.51456</td>
<td>0.5085</td>
<td>0.51456</td>
</tr>
<tr>
<td>7</td>
<td>0.50842</td>
<td>0.51456</td>
<td>0.50842</td>
<td>0.51456</td>
</tr>
<tr>
<td>8</td>
<td>0.5085</td>
<td>0.51456</td>
<td>0.5085</td>
<td>0.51456</td>
</tr>
<tr>
<td>9</td>
<td>0.50842</td>
<td>0.51456</td>
<td>0.50842</td>
<td>0.51456</td>
</tr>
<tr>
<td>10</td>
<td>0.5085</td>
<td>0.51456</td>
<td>0.5085</td>
<td>0.51456</td>
</tr>
</tbody>
</table>

### IV. CONCLUSIONS

By using the neural networks and genetic algorithm, combined with the characteristics of risk investment, this paper proposes a risk investment evaluation model. The model comprehensively evaluates the data in two aspects, i.e., the investment risk degree and the future benefit degree of risk investment. Combined with the investment appetites of investors, it provides an almost perfect evaluation system as a reference for rational investing of risk investment. The Living example analysis shows that this model is effective in solving the project evaluation of risk investment problems.

### ACKNOWLEDGMENTS

This work is funded by a project: 'Chunhui Plan' coresear-ch project of Ministry of Education (Number: 22009-1-01016) in China.
REFERENCES


[18] LIU Guohua, BAO Hong, LI Wenxiao. Implementation of ANN-GA System Based on WWW. Computer Engineering. 2002,02

116