Neural network optimization based on data coding and genetic algorithm

Haoran Song,

Research School of Computer Science, Australian National University, Canberra,
2600 ACT, Australia
u6688461 @anu.edu.au

Abstract. Sometimes the neural network has to accept a large amount of input information. For example, when the input information is a high-definition image, the amount of input information may reach tens of millions. It is very difficult work for the neural network to directly learn from tens of millions of information sources. Sometimes the neural network work also receive some information processed by users, which needs to be decrypted or pre-processed to be better learned by the neural network. If the most representative information can be presented, and then put into the neural network for learning, it will be more conducive to the learning of the neural network. In the regression operation, we can preprocess the input data according to the characteristics of the data. In this article, we use the discretization of the data and select the data according to the data correlation. In the classification operation, equilateral coding was performed on the output data. In addition, this paper also process genetic algorithm to initalize hyperparameter of the nerual network. After the corresponding data preprocessing and genetic algorithm, the training speed and accuracy of the neural network have been significantly improved. In the real world, data often lacks certain interesting attribute values, is extremely susceptible to noise, and data sets often come from multiple heterogeneous data sources. Low-quality data will lead to low-quality neural network learning results. So we need a certain encoding and decoding strategy to ensure meaningful output.

Keywords: Data decryption, data preprocessing, equilateral coding, genetic algorithm

1 Introduction

With the development of neural network technology, neural networks are required to play a role in more and more fields. It means that, according to the needs of the field, the neural network needs to build more complex models so that it can process a variety of complex data. But in the field of natural sciences, it is not that the more complex the model, the more popular it is. The more complicated the model, the worse the interpretation is. If your complex model is not supported by theory, it is obviously very difficult to find out the meaning of the subject. Therefore, the decryption of data and the understanding of the relationship between data is one of the important research directions at this stage. Filter out meaningful data to reduce the complexity of the neural network without affecting the performance of the neural network. The selected data set "oil-well" was obtained from three oil wells in the northwestern continental shelf off Western Australia [4]. The complex and professional attributes of oil wells and the processed normalized data can help us study the data processing of neural networks. In this paper, a most basic back propagation neural network model is established. Only the input and output layers and a hidden layer are established. The focus is on studying the effect of input and output data processing on the performance of the neural network. In this paper, regression prediction is made for the porosity (PHI) attribute of the data, and classification prediction is made for the Lithofacies attribute. When performing regression analysis, the attributes are encoded according to the relationship between the attributes and the meaning of the representative. In classification, equilateral coding is used to improve the classification ability of the model. It is expected that the above encoding operations can have a positive effect on the training accuracy and efficiency of neural networks. In addition, this paper also uses genetic algorithm to optimize the neural network parameters, the purpose is to further improve the accuracy and convergence speed of the neural network.

2 Method

2.1 Basic Neural Network

• Initialization parameters

This paper establishes a simple neural network model, which consists of an input layer, a hidden layer, and an output layer. The BP neural network in this paper adopts the method with momentum gradient descent as the training method of the network, and its training function is the Adam function. The performance function is the mse function. The learning rate $\mu = 0.02$, and the maximum number of training is 1000. The initial weights and thresholds are the system default values.

• The number of hidden neurons

The hidden layer nodes of the BP neural network are usually determined by trial and error. First set fewer hidden layer nodes to train the network, then gradually increase the number of hidden layer nodes, use the same sample set for training, and finally select the minimum network error The number of corresponding hidden layer nodes[7]. Tested from 4-100 hidden neurons, the results are as follows：

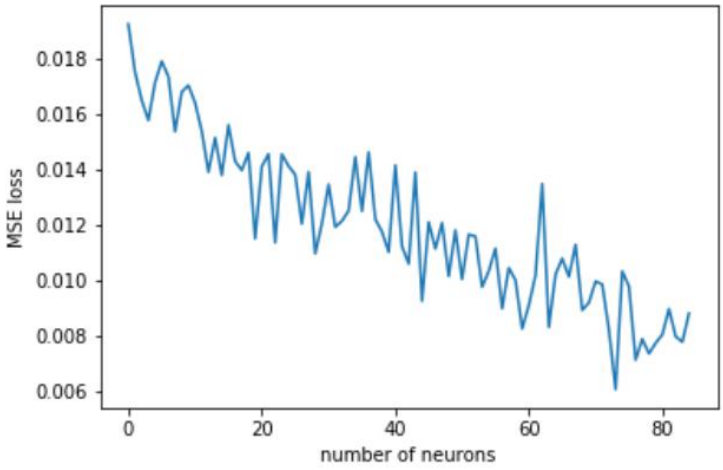

Fig. 1. Testing Hidden Neurons

As it is shown in Fig.1 , 73 is selected to be the number of neurons of hidden layer.

2.1 Equilateral coding

The network is a classifier and the output vector is very sparse which leads to learning difficulties, because tss can be improved by pushing the output vector to all 0s [1]. Equilateral coding is a method of normalizing data composed of classes into an array of floating-point values. It encodes nominal values for input into machine learning. Sometimes it produces better results than a one-hot coding[5]. During the training process, the output neurons are continuously checked according to the ideal output value provided in the training set. The error between the actual output and the ideal output is expressed in increments. This limits the number of neurons that cause single-hot coding wrong answers [5]. The case of oil-well study has to deal with classes of size 3. In this example, the data will be normalized between 0 and 1. If the ideal value is class 1, and actually class 2, which is shown as following:

Class	Output 0	Output 1	Output 2
Class #1	0.9	0.1	0.1
Class #2	0.1	0.9	0.1
Class #3	0.1	0.1	0.9

Fig. 2. One of 3 One Hot Encoding

Only one output neurons are incorrect. However, the entire group of neurons is part of the answer. Equilateral coding attempts to spread this error to more neurons. For this, the outputs should be provided with a unique set of values. Each set of values should have equal Euclidean distance. Equal distances ensure the same error weights [3]. And only N-1 dimensions are needed to place N points equidistantly (you can make an equilateral triangle in 2D), so equilateral coding will generate a look up table of (n-1) values for each (n) class. In the oil-well data set, rocks need to be divided into three categories. Therefore, a 2-dimensional equilateral coding is required, the coding is as follows (define the coding in the range of 0.1-0.9):

Class	Output 0	Output 1
Class #1:	0.15	0.30
Class #2:	0.85	0.30
Class #3:	0.50	0.90

Fig. 3. Equilateral Encoding of 3 classes

To retrieve the category, you need to calculate the Euclidean distance between the output vector and the above value. R.A. Bustos and T.D. Gedeon gave the calculation method in his article:

$$\text{Output category} = \text{MIN}(\text{distance } (U_i))$$

where

$$\text{distance } (U_i) = \sqrt{\sum_{j=1}^{4} 3\,(y_j - u_{j_i})^2}$$

where

$$U_i = (u_{j_S},\ u_{j_{DS}},\ u_{j_{WDS}},\ u_{j_{WS}},\ u_{j_{RF}})$$

Fig. 4. calculate Euclidean distance[1]

Because the output is encoded as a 2-dimensional vector, and the output of the simple neural network is "dimension = category", that is to say, the output of the neural network before adjustment is 3-dimensional, therefore, it is necessary to rewrite the output and loss function of the neural network to receive the output of the encoded two-dimensional vector.

2.2 Data correlation

If the correlation of some dimensional inputs is too strong, then the weights connected to these input neurons in the network actually play a similar role, and the effort to adjust the relationship between these weights when training the network is wasted. Therefore, it is necessary to detect the correlation between the data. The Correlation in Pandas can clearly see the correlation between the data as follows:

	GR	RDEV	RMEV	RXO	RHOB	NPHI	PEF	DT	Phi	logK	FLAG
GR	1.000000	0.254695	0.422141	0.689452	0.512372	0.370755	-0.297580	-0.099455	-0.629267	-0.465054	-0.328427
RDEV	0.254695	1.000000	0.937310	0.554654	0.126055	0.135985	0.128550	-0.236076	-0.241658	-0.251146	-0.117109
RMEV	0.422141	0.937310	1.000000	0.690733	0.141283	0.148150	-0.020499	-0.167324	-0.303792	-0.365901	-0.098265
RXO	0.689452	0.554654	0.690733	1.000000	0.435113	0.362406	-0.102363	-0.211125	-0.488222	-0.465580	-0.274419
RHOB	0.512372	0.126055	0.141283	0.435113	1.000000	0.264341	0.196182	-0.623235	-0.423729	-0.298469	-0.237949
NPHI	0.370755	0.135985	0.148150	0.362406	0.264341	1.000000	0.336415	0.240860	-0.070558	-0.223034	-0.173015
PEF	-0.297580	0.128550	-0.020499	-0.102363	0.196182	0.336415	1.000000	-0.133127	0.264797	0.000348	0.138284
DT	-0.099455	-0.236076	-0.167324	-0.211125	-0.623235	0.240860	-0.133127	1.000000	0.377967	0.200646	0.137166
Phi	-0.629267	-0.241658	-0.303792	-0.488222	-0.423729	-0.070558	0.264797	0.377967	1.000000	0.449463	0.412158
logK	-0.465054	-0.251146	-0.365901	-0.465580	-0.298469	-0.223034	0.000348	0.200646	0.449463	1.000000	-0.181690
FLAG	-0.328427	-0.117109	-0.098265	-0.274419	-0.237949	-0.173015	0.138284	0.137166	0.412158	-0.181690	1.000000

Fig. 5. Correlation of attributes

It can be found that the correlation between RMEV and RDEV reaches 0.94, RDEV is deep resistivity, and RMEV is shallow resistivity. So one of the two variables can be deleted. And because the correlation of RMEV to FLAG is -0.098, and the correlation of RDEV to FLAG is -0.12, it can be seen that RDEV has a higher impact on the rock category, and choose to retain RDEV. In addition, although the correlation between DT and PEF is not high, for FLAG, which is the type of rock, their degree of influence is the same, so you can choose one of the two.

2.3 Sparse data encoding

The literature uses the sparse vector encoding in the Geology descriptor attribute to distinguish popular types and rare types. In the Geology descriptor attribute, although the data range is from 10-90, the popular types are only 50, there are three values of 70 and 90, so three dimensions can be used in a four-dimensional vector to represent these three values. Other rare types are concentrated in the one-dimensional vector, which helps to enhance the characteristics of the data. In the oil-well data set, all the data are discretion and plotted, as shown in the following figure:

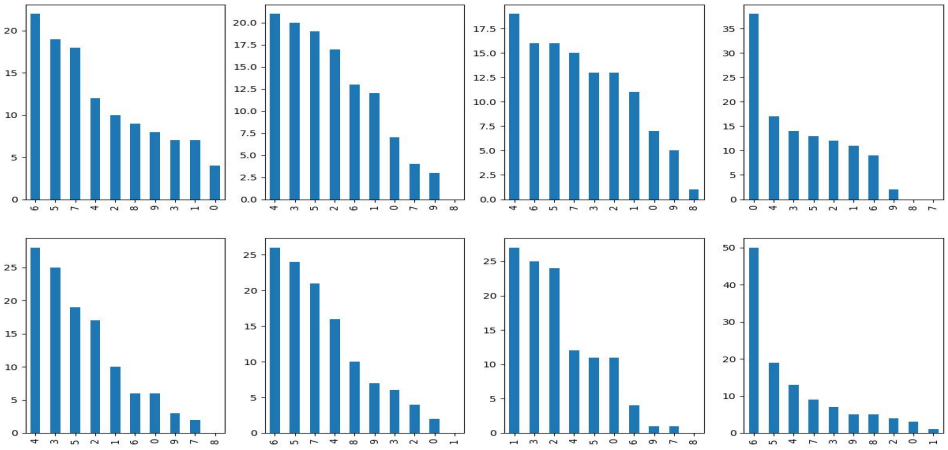

Fig. 6. Discretion data

It can be found from the above figure that in the oil-well data set, most of the data is evenly distributed, and there are no types are quite common and the others are rare. Two attributes that are relatively close to sparse data were screened out and tested (the third and fourth attributes in the second row of the above figure are 'PEF' and 'DT'). It can be considered that the 1, 3, and 2 types of data in the 'PEF' attribute are quite common and others are rare, so according to the method proposed by R.A. Bustos and T.D. Gedeon, the data can be encoded as Table 1[1]:

Table 1. Sparse coding in 'PEF'

Value	G1	G2	G3	G4
0	0.9			
1		0.9		
2			0.9	
3				0.9
4	0.9			
5	0.9			
6	0.9			
7	0.9			
8	0.9			
9	0.9			

Table 2. Sparse coding in 'DT'

Value	G1	G2
0	0.9	
1	0.9	
2	0.9	
3	0.9	
4	0.9	
5	0.9	
6		0.9
7	0.9	
8	0.9	
9	0.9	

In the 'DT' attribute, it can be considered that only category 6 is a common category, so based on the method proposed by R.A. Bustos and T.D. Gedeon, vectors can be changed to two-dimensional vectors, which is shown in Table 2[1].

The advantage of sparse representation is to reduce the complexity of the representation. The reason for more straightforwardness is actually to reduce the coefficient parameters. Through sparse representation, the information can be made full use of contained in the data. In addation, it can also remove redundant data information and maximize the use of data. Therefore it can be assumed that after sparse data encoding, the training results of the neural network will have a positive impact.

2.4 Genetic algorithm

Aiming at the problems that the simple neural network is easy to fall into the local minimum and the convergence speed is slow[7], a genetic algorithm is proposed to optimize the neural network and used for Phi prediction in the oil-well

data set. A neural network is used to establish the Phi regression prediction model. The genetic algorithm is used to optimize the initial weights and thresholds of the neural network.

The implementation steps of genetic algorithm are as follows:

• Individual coding and population initialization

The individual contains the weight and threshold of the entire neural network. As can be seen from the above, the neural network designed in this paper is 3 layers, in which the number of hidden layer neurons is 73, so the weight and threshold of the neural network are composed of four parts:

Table 3. Individual coding

Parts	Size
Hidden weight	(73,7)
Hidden bias	(73)
Predict weight	(1,73)
Predict bias	(1)

In this paper, individuals are encoded using real-number encoding[6]. The size of the population has a great influence on the global search performance of the genetic algorithm. The size of the population should be selected according to the specific problem. Therefore, this paper tested the initial population size.

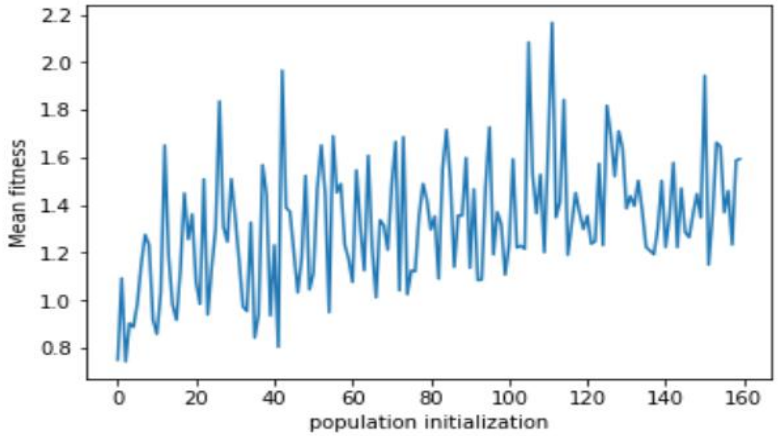

Fig.7. The mean fitness of initial population size from 0-160

As it is shown in Fig.7 , 110 is selected as the initial population size.

• Setting of fitness function

In this paper, the fitness function is set to the inverse of the neural network MSE:

$$f = \frac{1}{MSE} \tag{1}$$

It can be seen from the fitness function that the smaller the prediction error of the neural network, the larger the corresponding fitness function and the better the adaptability. In addition, in the test, it is found that the reciprocal of MSE will have a phenomenon that the fitness gap between individuals is not obvious, so this article expands MSE to 10 times and then takes the reciprocal, the purpose is to better distinguish the differences between individuals.

• Individual choice

Individual selection can be based on probability values, the formula is as follows:

$$\mathrm{Pi} = \frac{\mathrm{F_i}}{\sum_{i=1}^{k} \mathrm{F_i}} \tag{2}$$

Where Fi is the fitness value of individual i, k is population size

• Cross operation and mutation operation

The optimal individual does not perform crossover operations, but directly replicates into the next generation. For other individuals, the crossover probability P_c is used to perform a crossover operation on the 2 individuals, generating another 2 new individuals and replacing the parents. Similarly, the optimal individual does not perform mutation operations, but directly copies to the next generation. For other individuals, the mutation probability P_m is used to perform mutation operations to produce another new individual and replace the parents. In this experiment, P_c =0.8, P_m =0.005.

• Selection of iteration times

In this paper, the impact of the number of iterations on the results is tested in the range of 0-400 generations. The test results are as follows:

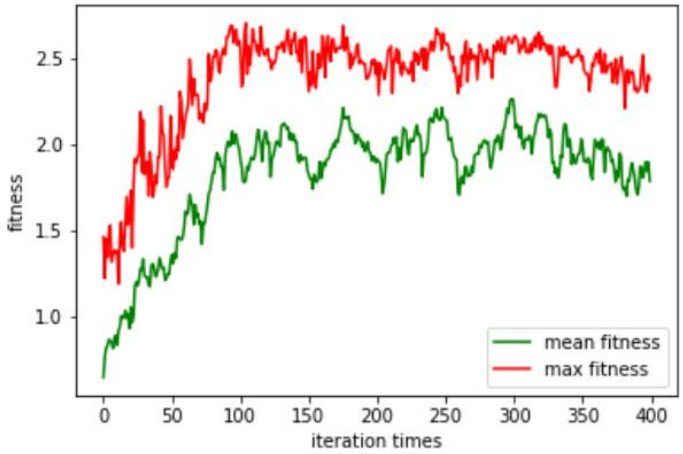

Fig.8. The fitness in each generation from 0-400

It can be found that after the individual has evolved to 200 generations, the fitness is unstable and is gradually declining. In this paper, 170 generations are selected as the final number of iterations.

3 Result

3.1 Classification

Firstly, observe the output of the ordinary neural network. After 1000 steps of training, the accuracy test is performed. The accuracy of the training set can reach 0.85, while the accuracy of the test set is only 0.68. It can be found that the model has overfitting. In order to alleviate the overfitting, the following adjustments have been made to the neural network:

• Using the Adam optimizer, adding weight_decay = 0.01, the accuracy increased from 0.68 to 0.72. It can be found that the phenomenon of overfitting has been improved a little but not greatly improved. It can be speculated that a large reason is the lack of training data.

On this basis, the output vector is equilaterally encoded, After testing, the accuracy of the training set can reach 0.92, and the accuracy of the test set can also be increased to 0.82. The neural network optimization and its effects in the classification stage are shown in the following table:

Table 4. Optimization Strategies And Results

Optimization Strategy	Acc in train set	Acc in test set
Hidden neuron: 80-100	0.85	0.68
Using Adam, weight decay = 0.01	0.88	0.72
Equilaterally encoding	0.92	0.80
Dropout = 0.2, lr = 0.1	0.92	0.82

The results show that the lack of training data will have a certain impact on the prediction of the neural network, but the optimization effect of equilateral coding can make up for such deficiencies to a certain extent.

3.2 Regression

Firstly, observe the output of the ordinary neural network. This article uses MSE loss as the criterion. The initial regression model, after 1000 steps of training, the MSE loss of the training set is 0.029, and the ARsquare value of the test set is 0.056.

On this basis (including the above-mentioned operations to deal with overfitting), the correlation degree was screened. According to the above, the RMEV and DT attributes were removed. The test result is: the MSE loss in the training set decrease to 0.019, and which in the test set is 0.051. It can be seen that after processing relatively irrelevant data, the

training effect of the neural network has been significantly improved. Finally, tried the sparse coding method, after replacing the original data, using the same neural network configuration, after training the same number of steps. It is found that compared with the original neural network, the accuracy of the training set and the test set has not increased to a large extent. On the contrary, in some cases, even the situation is not as good as before.

As mentioned above, genetic algorithms can be used to optimize neural network parameters. After searching for the best individual neural network parameters through genetic algorithm, the weights and thresholds obtained after optimization using genetic algorithm are brought into the BP neural network and retrained. The comparison between the training results and the ordinary neural network is as follows:

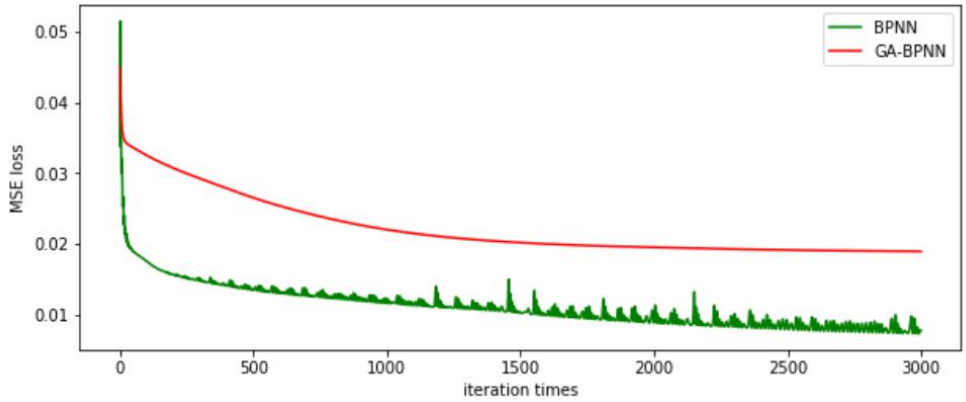

Fig.9. The MSE loss in each step in training set

The training results show that after the same training of 3000 steps, the loss of the training set drops to about 0.007, and the loss of the test set also drops to 0.029. In summary, in the regression operation, the data processing strategy and the results are as follows:

Table 5. Optimization Strategies And Results

Optimization Strategy	Acc in train set	Acc in test set
Simple NN	0.029	0.056
Correlation selection	0.019	0.051
Sparse Coding (best time)	0.021	0.048
Generic Algorithm	0.007	0.029

The results show that the attributes selected according to the correlation can indeed have a positive effect on the neural network, but the sparse coding does not have the expected performance in this data set. However, it cannot be concluded that sparse coding will not have a good impact on the neural network, and the specific situation needs further follow-up discussions. In addition, it is proved that GA does have a significant effect on the optimization of neural network parameters.

4 Discussion

In the case of classification, the accuracy of the designed simple neural network in the training set / test set is about 0.88 / 0.72, and after using the equilateral coding method in the literature.the accuracy of the training set / test set can be improved to 0.92 /0.82. Equilateral coding performs well in the oil-well data set, and has no effect on the training complexity, but if there are more types of classification, such as 7 types or more, then a 7-dimensional vector must be generated. Matching each category will increase the computational complexity of the neural network to some extent.

In the case of regression, the accuracy of the designed simple neural network in the training set / test set is about 0.92 / 0.78, and the accuracy of the training set / test set can be improved to 0.98 / 0.92 after the data has been processed accordingly. . In the process of feature selection, high-quality attributes are manually selected, which will bring some problems. For example, the writer does not understand this field, and the understanding of attributes is not appropriate enough, delete important parts, or retain redundant parts. In addition, in the oil-well data set, there are only 8-9 attributes that we need to analyze. If there are dozens or hundreds of attributes that need to be analyzed, it is impossible to manually select good attributes. So try to use PCA for dimension reduction and feature extraction. Although principal component analysis (PCA) is an unsupervised algorithm that creates a linear combination of original features, the new

main components are unexplainable, and the threshold for cumulative interpretability variance must be manually set or adjusted [2].

The performance of sparse coding can be analyzed for roughly two reasons. First, the sparse features of the data are not obvious. Although the two selected attributes meet the sparse features to some extent, it can be seen that the values of the categories 0, 4, and 5 in 'PEF' are also many, it can also be thought that they are not rare. Although category 6 in 'DT' is far beyond other categories, there are no rare occurrences in other categories, only relatively few. In the oil-well data set, the sparse boundaries of the data are not very obvious, resulting in the performance of sparse coding is not as we assumed. Secondly, the sparse coding of data makes the neural network lose some features. In fact, this is also related to the first reason. In the category of that is considered to be rare, there are also many features that need to be considered. When we perform sparse coding, it is equivalent to ignore these features, which also leads to unsatisfactory sparse coding.

In the literature, some good data encoding methods are mentioned, but they are not adopted due to the limitation of the data set or the poor performance.

• Periodic data encoding: the literature sets total activation 2 for valid aspects, and encodes the output in 9 directions into a 4-dimensional vector output, so the early condition of the encoding is the cyclic nature of the information. In the oil-well data set, there are no similar properties, so this method is not used[1].

• Statistical Z function: In the literature, the numerical range of the data is very large, or some data ranges from 0-1, and some data ranges from 0-100. Different evaluation indicators often have different dimensions and dimensional units. This situation will affect the results of data analysis. In order to eliminate the dimensional influence between indicators, data standardization needs to be performed to solve the comparability between data indicators . However, in the oil-well data set, all data are subjected to 0-1 normalization, and there is no need for standardized processing.

The following table compares the gap between the training effects of GA-BPNN and BPNN:

Table 6. GA-BPNN VS BPNN

	BPNN	GA-BPNN
Convergence speed(Target MSE loss:0.02)	2000 steps	100 steps
Train set loss	0.019	0.007
Test set loss	0.048	0.029

It can be seen from the analysis of Table 6 that GA-BPNN not only has higher training accuracy than BPNN, but also has faster convergence speed than BPNN. If 0.02 is the target loss, BPNN needs about 2000 steps to reach, while GA-BPNN only needs about 100 steps. Therefore, genetic algorithm optimization not only accelerates the convergence speed of the network, but also improves the accuracy of prediction, and can predict Phi more accurately.

In summary, the data form of this data set is somewhat simple, and the room for modification and improvement is relatively small. Some methods have yet to be implemented using other data sets.

5 Conclusion and Future Work

This paper focuses on the influence of the input and output data encoding in the neural network on the training effect of the neural network. The experiment found that equilateral coding can improve the training effect of the neural network, and also alleviate the phenomenon of overfitting to a certain extent, but it is limited by the number of classification types; using the correlation between attributes to filter attributes can also effectively enhance the performance of the neural network. However, the limitation of manual screening is also one of the shortcomings. The research in this paper provides another possibility for data preprocessing. Good data preprocessing can greatly optimize the performance of neural networks. The quality of the data directly determines the prediction and generalization capabilities of the model. So that neural networks can be more widely used in various disciplines. In addition, this paper also uses genetic algorithms to initialize the parameters of the neural network, improve the performance of the neural network, and help the neural network to improve the accuracy of the prediction and the speed of convergence. However, the analysis of cross probability and mutation probability in genetic algorithms is not enough in this paper, and more detailed research in the future is needed. The future is the era of big data. Data processing is the core of the entire big data processing process. Preprocessing such a large amount of data is a very severe test for computing resources and processing algorithms. Therefore, the scalability algorithm is one of the future development goals of data preprocessing.

References

1. Bustos, R. A., & Gedeon, T. D. (1995). Decrypting Neural Network Data: A GIS Case Study. In Artificial Neural Nets and Genetic Algorithms (pp. 231-234). Springer, Vienna.
2. Dimensionality Reduction Algorithms: Strengths and Weaknesses, https://elitedatascience.com/dimensionality-reduction-algorithms
3. Guiver, J. P., & Klimasauskas, C. C. (1991). Applying neural networks, Part IV: improving performance. PC AI Magazine<, 5, 34-41.
4. H. Kuo, T.D. Gedeonc,& P.M. Wong.(1999). A Clustering Assisted Method for Fuzzy Rdc Extraction and Pattern Classification .IEEE Trans. on Fuzzy Systems (pp. 679-684).
5. Masters, T. (1993). Practical neural network recipes in C++. Morgan Kaufmann.
6. Montana D J, Davis L. Training Feedforward Neural Networks Using Neural Networks and Genetic Algorithm[C]. Proc. Of International Conference on Computing, Communications and Control Technologies. Austin, USA: [s. n.], 1989: 762-767.
7. Y.M. Gao., &R.J. Zhang.(2014). Forecast analysis of house prices based on genetic algorithm and BP neural network. Computer Engineering. Vol.40, (pp. 679-684).