

# An Application Interface Design for Backpropagation Artificial Neural Networks

Volkan Cavus, and Adem Tuncer

**Abstract**—In this study, a computer based application interface has been designed for educational purpose of modelling artificial neural network using backpropagation algorithm. The software is intended to be of use both for the novice and for the experienced user. Data selection, number of neurons in the hidden layer, momentum rate, learning rate, activation function and the number of data to be trained can be determined by the user. Using these data, training of the ANN successfully achieved and be tested at the end of the training. It's observed that objective study is able to find solutions to real time ANN problems.

**Keywords**—Artificial Neural Networks, Backpropagation Algorithm, User Interface Design.

## I. INTRODUCTION

A neural network is a computing system made up of a number of simple, highly interconnected processing elements, which process information by their dynamic state response to external inputs [1]. Artificial neural network (ANN) has the abilities to be able to learn, explore and then generalize the data to produce new information, and to detect the relationships between variables. These abilities are very difficult to accomplish with traditional programming methods. ANN was developed for difficult or impossible events to program and is seen as a branch of science concerned with information processing. Despite their development inspired by the structure of the human brain, some features of the ANN are not the same as a biological neural network. However, there is no biological response of some features, but still shows a large proportion of similarities with the brain. ANNs are successfully applied to many areas such as computer science, finance, medicine, engineering, geology and physics.

User interface design focuses on interaction between the user and an application and the interface are easy to access, understand and use for the user's needs. The user interface design helps users to run their programs quickly and easily. In this study, a user interface has been designed for educational purpose of modelling ANN using backpropagation algorithm. Data in the data set can be taken and required parameter settings can be done and so training

Volkan Cavus is with Institute Of Natural Sciences, Yalova University, Yalova, 77200 Turkey (e-mail: cvsvlkn@gmail.com).

Adem Tuncer is with Department of Computer Engineering, Yalova University, Yalova, 77200 Turkey (corresponding author's phone: +90 226 815 5347; e-mail: adem.tuncer@yalova.edu.tr).

and testing operations can be performed easily with backpropagation ANN in the proposed application interface. In addition, the actual values and the values obtained from the training process can be seen in the interface application in order to evaluate the results. A user, who has a basic level of knowledge about ANN, can easily run ANN using the developed interface.

## II. ARTIFICIAL NEURAL NETWORKS

The human brain consists of a large number, more than a billion of neural cells. Learning in biological systems occurs with synaptic connections between the neurons. This is true of ANN as well. ANN is an artificial illustration of the human brain that tries to imitate its learning process [2]. It has the ability to learn; recall and generalize from the data which are used to train the system through trials like a human brain. It is often used for predicting the future like weather, harvest, energy consumption, earthquakes. It requires two parts, one of which is the training and the other is the testing. It uses samples to establish the relationship events, and decides to solve problems that will occur after learning the relationships and comments.

ANN is formed in three layers, called an input layer, an output layer and one or more hidden layers between the input and output [3]. In input layer, a neuron receives input data from outside of the network. The results of the network are in the output layer. It has the network's predictions or classifications. The hidden layer is located between the input layer and the output layer. Data from the input layer is processed and transferred to the output layer in the hidden layer. Each layer consists of one or more nodes, represented in Fig. 1. The lines between the nodes indicate the flow of information from one node to the next.

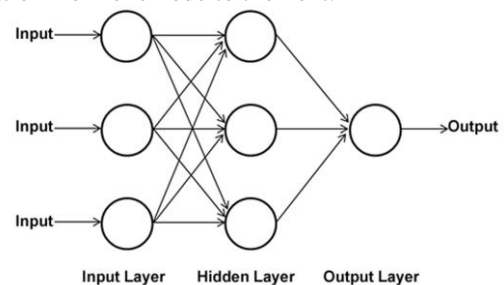


Fig. 1 Artificial neural network model

## III. BACKPROPAGATION ALGORITHM

There are many learning algorithms used in ANN.

Backpropagation networks are the most commonly used network because they offer good generalization abilities and are relatively straightforward to implement [4]. The main principle of the algorithm is to minimize the error between the actual output value and the output value of the ANN. The error is calculated by the difference between the actual output value and the ANN output value. If there is a large error, then it is fed back to the ANN to update synaptic weights in order to minimize the error. This process continues until the minimum error is reached [5].

The backpropagation algorithm consists of two steps, including forward and backward calculation. In the forward calculation step, outputs at the output layers are determined by using input layer. In the backward calculation step, the error values are propagated back through the network to determine how the weights should be changed during training.

Changes in weight rate are calculated in the backward calculation with (1), and the new weight rate is obtained by collecting with the previous weight rate.

$$\Delta W_{ij}(t) = \eta \delta_j X_i + \alpha \Delta W_{ij}(t-1) \quad (1)$$

Where,  $\Delta W(t)$  is weight change,  $\Delta W(t-1)$  is weight change in the previous iteration,  $\eta$  is learning rate,  $\delta$  is the error,  $X$  is

the input value, and  $\alpha$  is momentum.

The learning rate determines the size of the weight changes. The learning rate directly affects the performance of the algorithm. If the learning rate is set too high, the algorithm may become unstable. If the learning rate is too small, the algorithm will take too long to converge [6]. The learning rate typically takes a value between 0 and 1 [4]. The momentum is used to prevent the system from converging to a local minimum. The momentum is used to add a certain fraction of the weight change value to the next one. The momentum is typically takes a value between 0 and 1.

#### IV. DEVELOPED APPLICATION INTERFACE DESIGN

In this study, an application user interface which parameter settings of ANN using backpropagation can be made by the user has been designed. Developed application interface using C# is shown in Fig. 2.

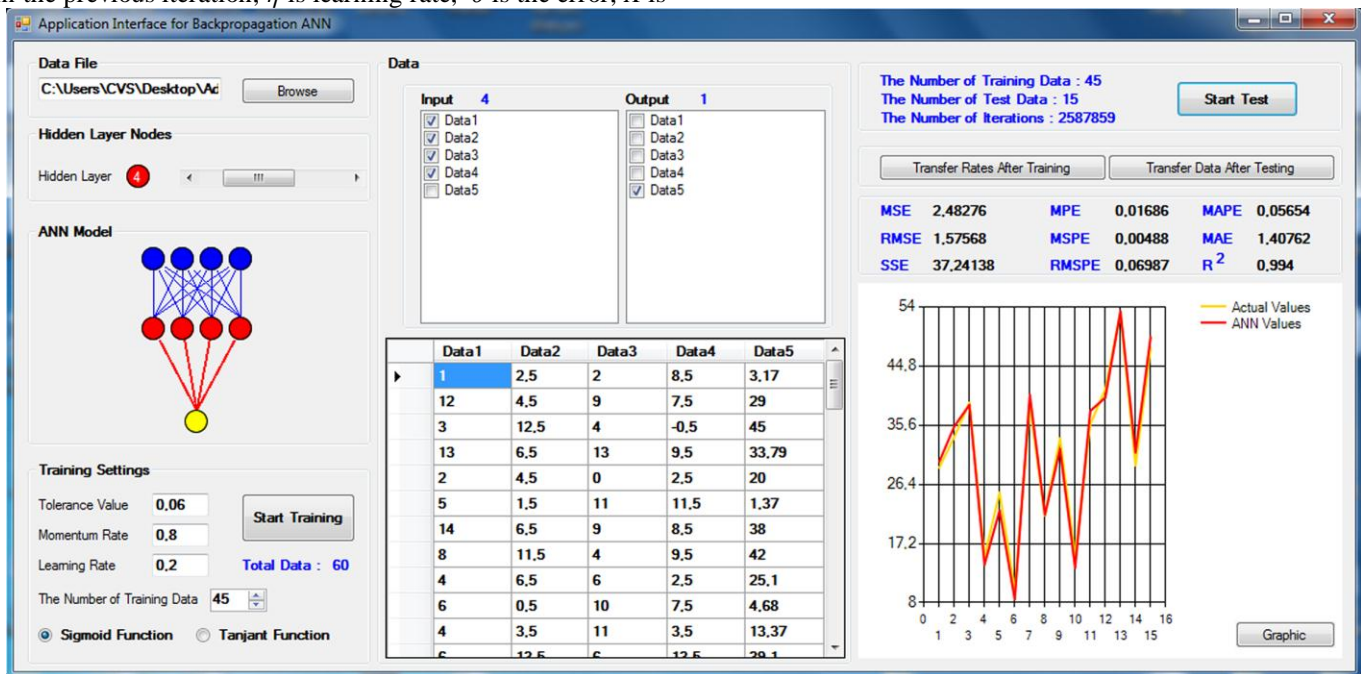


Fig. 2 Application interface for backpropagation ANN

Stored dataset is added to the application interface, and then necessary input and output fields for ANN are determined using the section shown in Fig. 3. The values specified in this section are defined as the number of neurons in the input and output layers of the backpropagation ANN.

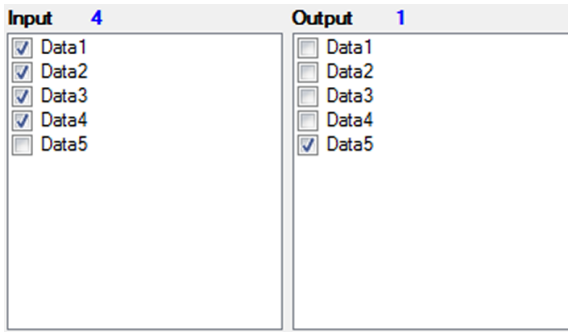


Fig. 3 Input and output fields

Fig. 4 shows the training settings section. In this section, the training is started by determining activation function, ANN parameters, and number of training and testing data in the data set. The most popular activation functions for backpropagation networks are the sigmoid function and tangent function. In this study, both functions were used.

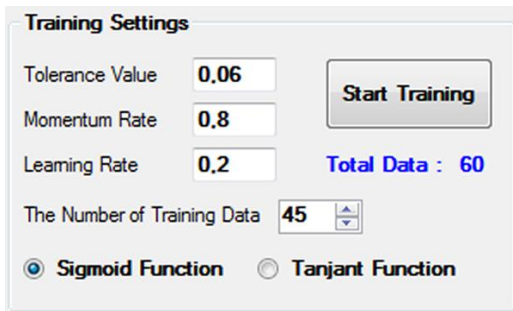


Fig. 4 ANN training settings

Min-Max normalization method is used to convert data input and output values in ANN to values between 0 and 1. This method is given in (2).

$$x = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}} \quad (2)$$

Where  $x$  is normalized data,  $x_i$  is the input value,  $x_{\min}$  is the smallest value within the data set and  $x_{\max}$  is the largest value within the data set.

The actual values and the values obtained after training is completed are also given on the interface screen in order to evaluate the results in shown Fig. 5. The error values and the error rates can be seen on this section using “Graphic” button.

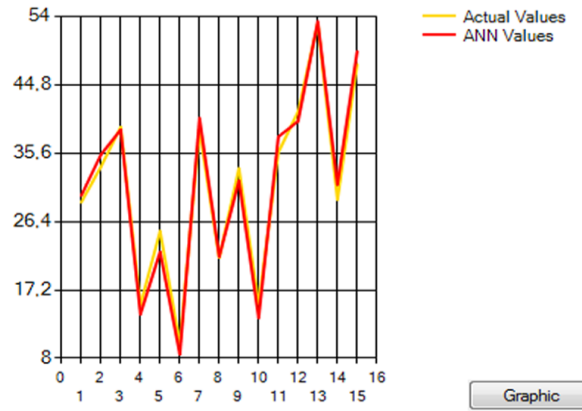


Fig. 5 Graph of actual and ANN values

In order to test the success of the proposed study, the sample has been tested on a real data set. When looking at the chart for the sample data set, it is obviously seen that the actual values and the values produced by the ANN are close to each other. This shows that developed training set works successfully.

Some of the performance criteria values used to measure the success of the ANN are used in the developed interface. Thus, the success of the training can be analyzed based on multiple performance criteria. Performance criteria values can be reached after training and testing on the interface screen shown in Fig. 6.

<b>MSE</b>	<b>2.48276</b>	<b>MPE</b>	<b>0.01686</b>	<b>MAPE</b>	<b>0.05654</b>
<b>RMSE</b>	<b>1.57568</b>	<b>MSPE</b>	<b>0.00488</b>	<b>MAE</b>	<b>1.40762</b>
<b>SSE</b>	<b>37.24138</b>	<b>RMSPE</b>	<b>0.06987</b>	<b>R<sup>2</sup></b>	<b>0.994</b>

Fig. 6 Performance criteria values

MSE(Mean Square Error), RMSE(Root Mean Square Error), SSE(Sum Square Error), MPE(Mean Percentage Error), MSPE(Mean Square Percentage Error), RMSPE(Root Mean Square Percentage Error), MAPE(Mean Absolute Percentage Error), MAE(Mean Absolute Error), and R<sup>2</sup>(Correlation Coefficient) values are obtained as performance criteria values.

The flow chart of backpropagation ANN that used in the study is given in Fig. 7.

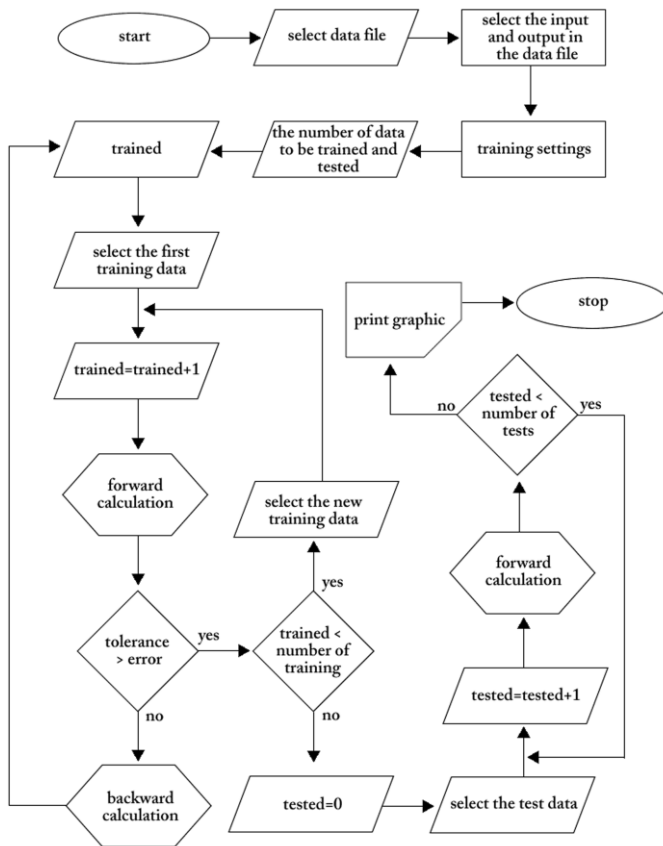


Fig. 7 Flow chart of the algorithm

## V. CONCLUSION

In this study, a computer based application interface is designed for educational purposes ANN using backpropagation. A user, who has a basic level of knowledge about ANN, can easily run ANN using the developed interface. When using the application interface, parameter entries or changes can be made such as neuron numbers of the hidden layer, momentum, learning rate and activation function parameters without writing any code or process. The actual values and obtained values after training are transmitted to the chart. It is easy to understand and interpret the ANN through the chart. Different experimental results demonstrate the success and validity of the study.

## REFERENCES

- [1] M. Caudill, "Neural Network Primer: Part I", *AI Expert*, Feb. 1989.
- [2] R. Kottaimalai, M. P. Rajasekaran, V. Selvam and B. Kannapiran, "EEG Signal Classification using Principal Component Analysis with Neural Network in Brain Computer Interface Applications", *2013 IEEE Int. Conf. on Emerging Trends in Computing, Communication and Nanotechnology (ICECCN 2013)*, Tirunelveli, India, March 2013, pp. 227-231.  
<http://dx.doi.org/10.1109/ICE-CCN.2013.6528498>
- [3] M. Alaraj, M. Abbod, Z. Hunaitii, "Evaluating Consumer Loans Using Neural Networks Ensembles", *Int. Conf. on Machine Learning, Electrical and Mechanical Engineering (ICMLEME'2014)*, Dubai, January 2014, pp. 55-61.
- [4] A. U. Khan, T. K. Bandopadhyaya, S. Sharma, "Comparisons of Stock Rates Prediction Accuracy using Different Technical Indicators with Backpropagation Neural Network and Genetic Algorithm Based Backpropagation Neural Network", *First International Conference on*

*Emerging Trends in Engineering and Technology 2008, ICETET '08*, Nagpur, Maharashtra, India, July 2008, pp. 575-580.  
<http://dx.doi.org/10.1109/ICETET.2008.59>

- [5] M. A. Gunler, "Identification of Facial Expressions Using Artificial Neural Networks (Ann)", *a Master's Thesis, Electrical & Electronics Engineering*, Atılım University, July 2012.
- [6] M. Mohebbi, H. A. Moghadam, "Real-Time Ischemic Beat Classification Using Backpropagation Neural Network", *IEEE 15th Signal Processing and Communications Applications*, 2007. SIU 2007, Eskisehir, Turkey, June 2007, pp. 1-4.

**Volkan Cavus** is currently a lecturer with the Sinop University, Sinop, Turkey and M.Sc. student with Institute Of Natural Sciences, Yalova University. His current research is artificial neural networks.

**Adem Tuncer** received the Ph.D. degree from Kocaeli University, Kocaeli, Turkey, in 2013. He is currently an Assistant Professor with the Department of Computer Engineering, Yalova University. His current research interests include heuristic optimization algorithms, artificial intelligence, genetic algorithms, path planning for mobile robots, fpga.