EVALUATING CREDIT RISK USING ARTIFICIAL NEURAL NETWORKS

QEETHARA K. AL-SHAYEA[1], and GHALEB A. EL-REFAE[2]

[1, 2] MIS Department
Al Zaytoonah University of Jordan
Amman, JORDAN
[1]kit_alshayeh@yahoo.com
[2]ghalebelrefae@yahoo.com

ABSTRACT

In credit business, banks are interested in learning whether a prospective consumer will pay back their credit. The goal of this paper is to classify the credit risk which an applicant can be categorized as a good or bad consumer using artificial neural networks, to enable all parties to take remedial action. The Feed-forward back propagation neural network is used to classify a consumer into two classes depending on selected parameters. One of the classes is credit worthy and likely to repay its financial obligation. The other class which is not credit worthy and whose applications for credit will be rejected due to a high possibility of defaulting on its financial obligation. Two well known and available datasets have been used (German and Australian dataset) to test the proposed neural network. The results of applying the artificial neural networks methodology to classify credit risk based upon selected parameters show abilities of the network to learn the patterns. In German dataset, the percent correctly classified in the simulation sample is approximately 77 percent. While, in Australian dataset, the percent correctly classified in the simulation sample is approximately 86 percent. The proposed neural network is evaluated using confusion matrices.

Keywords: Artificial Neural Networks, Credit Scoring, Business Intelligence, Feed-forward Back Propagation Neural Network and Receiver Operating Characteristic (ROC) Curves.

1.0 INTRODUCTION

The recent upsurge in research activities into artificial neural networks (ANNs) has proven that neural networks have powerful pattern classification and prediction capabilities (Zhang, 2004). ANN has been applied to a variety of business areas such as finance, auditing, accounting, management, decision making, marketing and production.

Kashman (2009) proposed a novel approach to credit risk evaluation using a neural network was presented. In our approach we train a three-layer supervised neural network, which is based on the back propagation learning algorithm, following seven learning schemes. Also, Kashman (2010) described a credit risk evaluation system that uses supervised neural network models based on the back propagation learning algorithm.

Bahrammirzaee (2010) presented a comparative research review of three famous artificial intelligence techniques, i.e., artificial neural networks, expert systems and hybrid intelligence systems, in financial market. Bahrammirzaee *et al.* (2011) designed a hybrid intelligent system for credit ranking using reasoning-transformational models. Expert system as symbolic module and artificial neural network as non-symbolic module are components of this hybrid system.

Marcano-Cedeno *et al.* (2011) presented an algorithm with the application of an ANN training Algorithm inspired by the neurons' biological property of metaplasticity. This algorithm is especially efficient when few patterns of a class are available, or when information inherent to low probability events is crucial for a successful application, as weight updating is overemphasized in the less frequent activations than in the more frequent ones.

2.0 ARTIFICIAL NEURAL NETWORKS

An Artificial Neural Network (ANN) is inspired by the biological brain, which consists of billions of interconnected neurons working in parallel. An (ANN) is a network of highly interconnecting processing elements (neurons) operating in parallel. These elements are inspired by biological nervous systems. As in nature, the connections between elements largely determine the network function. A subgroup of processing element is called a layer in the network. The first layer is the input layer and the last layer is the output layer. Between the input and output layer, there may be additional layer(s) of units, called hidden layer(s). Figure 1 represents the structure of typical neural network. Learning in ANNs takes place through an iterative training process during which node interconnection weight values are adjusted. Initial weights, usually small random values, are assigned to the interconnections between the ANN nodes. You can train a neural network to perform a particular function by adjusting the values of the connections (weights) between elements.
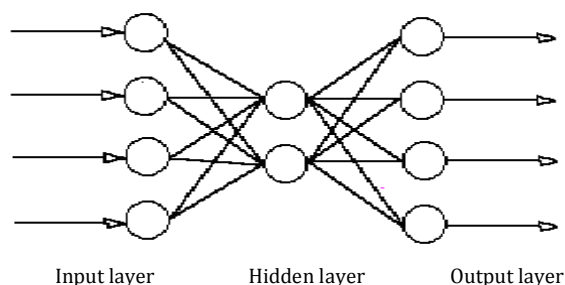

Input layer   Hidden layer   Output layer

Figure 1: Structure of typical neural network.

2.1 The Proposed Classification Model

The proposed model is aimed to classify applicants into two classes depending on selected parameters. This study identifies applicants into likely to repay its financial obligation or not likely to repay using the proposed neural network.

Feed-forward neural networks are widely and successfully used models for classification and problem solving. A typical feed-forward back propagation neural network consists of three layers: the input layer, a hidden layer, and the output layer. A feed-forward fully connected network is trained in supervised manner (Al-Shayea *et al.,* 2010). It consists of three layers: the input layer, a hidden layer, and the output layer. A one hidden with 20 hidden layer neurons is created and trained. The input and target samples are automatically divided into training, validation and test sets. The training set is used to teach the network. Training continues as long as the network continues improving on the validation set. The test set provides a completely independent measure of network accuracy. The information moves in only one direction, forward, from the input nodes, through the hidden nodes and to the output nodes. There are no cycles or loops in the network. The proposed neural networks are shown in Figure 2 and Figure 3.
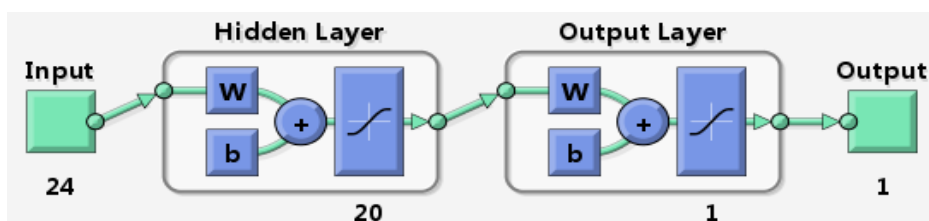

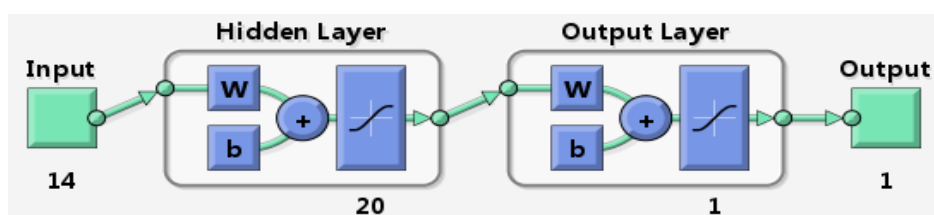**Figure 2:** The proposed classification neural network for German credit dataset.

Figure 3: The proposed classification neural network for Australian credit dataset.

It allows signals to travel one-way only; from source to destination; there is no feedback. The hidden neurons are able to learn the pattern in data during the training phase and mapping the relationship between input and output pairs. Each neuron in the hidden layer uses a transfer function to process data it receives from input layer and then transfers the processed information to the output neurons for further processing using a transfer function in each neuron. The output of the hidden layer can be represented as equation 1.

$$Y_{Nx1} = f(W_{NxM} X_{M,1} + b_{N,1})\tag{1}$$

Where,
Y is a vector containing the output from each of the N neurons in a given layer,
W is a matrix containing the weights for each of the M inputs for all N neurons,
 X is a vector containing the inputs,
b is a vector containing the biases and
f(·) is the activation function (Freeman and Skapura, 1991).

3.0 EXPERIMENTAL RESULTS

3.1 Data Analysis
Datasets used are a selection from UCI Machine Learning Repository. The classification performance was tested on two applicants' credit datasets. The two datasets are publicly available benchmark datasets, known as the Australian and German Credit Approval datasets, and they were also used in the Statlog project. Table 1 shows the datasets used for experiments.

Table 1: Datasets used for experiments.

Dataset	No. of samples	Classes
German	1000	700 good ; 300 bad
Australian	690	307 good; 383 bad

3.2 Performance Evaluation
Neural network toolbox from Matlab 7.9 is used to evaluate the performance of the proposed networks. A two-layer feed-forward network was created.

3.2.1 German dataset
A two-layer feed-forward network with 24 inputs and 20 sigmoid hidden neurons and linear output neurons was created. The dataset contains 1000 samples. 765 sample used in training the network while 235 samples used in testing the network. Training is done using scaled conjugate gradient back propagation network. The scaled conjugate gradient algorithm (SCG) developed by Moller (1993) was designed to avoid the time-consuming line search. This algorithm combines the model-trust region approach with the conjugate gradient approach.

The results of applying the proposed neural networks to distinguish between good and bad applicant based upon selected parameters showed very good abilities of the network to learn the patterns. The network was simulated in the testing set (i.e. cases the network has not seen before). The results were good; the network was able to classify 80.4 % of the cases in the training set. Figure 4 shows the training state values. Best validation performance is 0.13428 at epoch 13 as shown in Figure 5. The mean squared error (MSE) is the average squared difference between outputs and targets. Lower values are better while zero means no error.
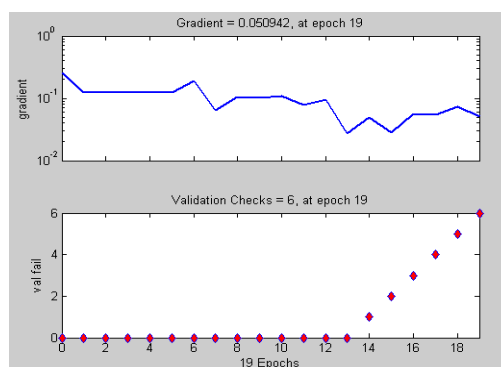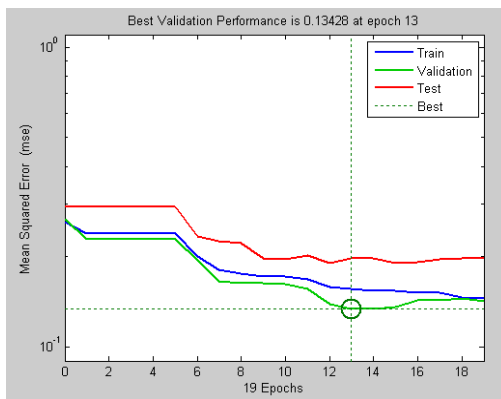

Figure 4: The training state values.

Figure 5: The best validation performance.

Figure 6 shows the confusion matrices for training, testing, and validation, and the three kinds of data combined. The diagonal cells in each table show the number of cases that were correctly classified, and the off-diagonal cells show the misclassified cases. The blue cell in the bottom right shows the total percent of correctly classified cases (in green) and the total percent of misclassified cases (in red).

The Receiver Operating Characteristic (ROC) curve is used to inspect the classifier performance more closely as shown in Figure 7. By definition, a ROC curve shows true positive rate versus false positive rate (equivalently, sensitivity versus 1–specificity) for different thresholds of the classifier output. You can use it, for example, to find the threshold that maximizes the classification accuracy or to assess, in more broad terms, how the classifier performs in the regions of high sensitivity and high specificity.
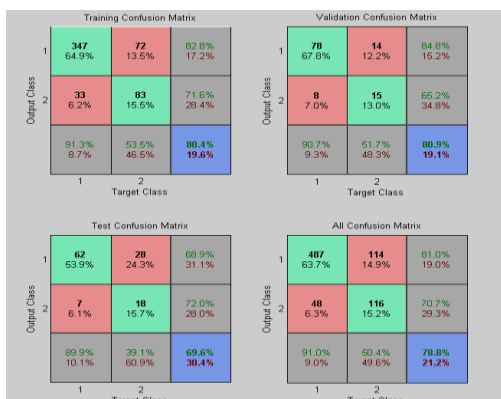

Figure 6: The confusion matrices for training, testing, and validation, and the three kinds of data combined.
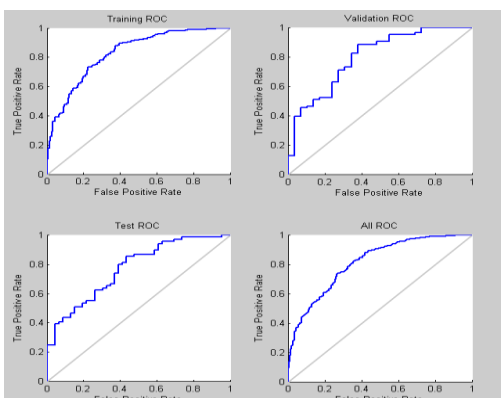

Figure 7: The Receiver Operating Characteristic.

The percent correctly classified in the simulation sample by the feed-forward back propagation network is 76.6 percent as shown in Figure 8. Figure 9 shows the ROC curve for simulation sample.
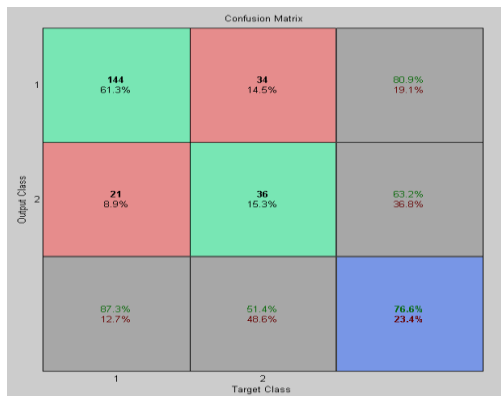
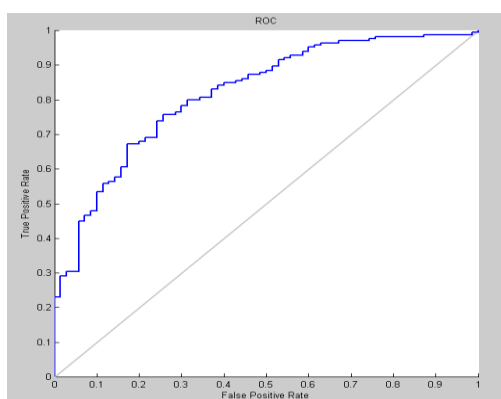**Figure 8:** The feed-forward back propagation network.

Figure 9: The ROC curve for simulation sample

3.2.2 Australian dataset
A two-layer feed-forward network with 14 inputs and 20 sigmoid hidden neurons and linear output neurons was created .The dataset contains 690 samples. 500 sample used in training the network while 190 samples used in testing the network. Training is done using scaled conjugate gradient back propagation network. This algorithm combines the model-trust region approach with the conjugate gradient approach.

The results of applying the proposed neural networks to distinguish between good and bad applicant based upon selected parameters showed very good abilities of the network to learn the patterns. The network was simulated in the testing set. The results were good; the network was able to classify 89.7% of the cases in the training set. Figure 10 shows the training state values. Best validation performance is 0.024484 at epoch 15 as shown in Figure 11.
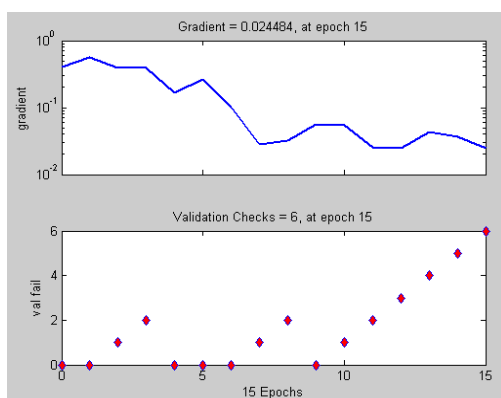

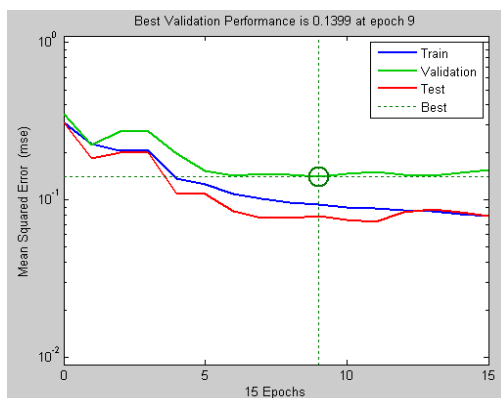**Figure 10:** The training state values.

Figure 11: The best validation performance.

Figure 12 shows the confusion matrices for training, testing, and validation, and the three kinds of data combined. The diagonal cells in each table show the number of cases that were correctly classified, and the off-diagonal cells show the misclassified cases. The blue cell in the bottom right shows the total percent of correctly classified cases (in green) and the total percent of misclassified cases (in red). The network's outputs are almost perfect, as you can see by the high numbers of correct responses in the green squares and the low numbers of incorrect responses in the red squares. The lower right blue squares illustrate the overall accuracies.
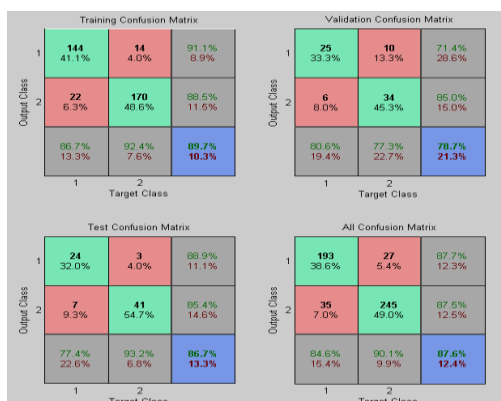

Figure 12: The confusion matrices for training, testing, and validation, and the three kinds of data combined.

Figure 13 shows the classifier performance more closely by plotting a Receiver Operating Characteristic (ROC) curve.
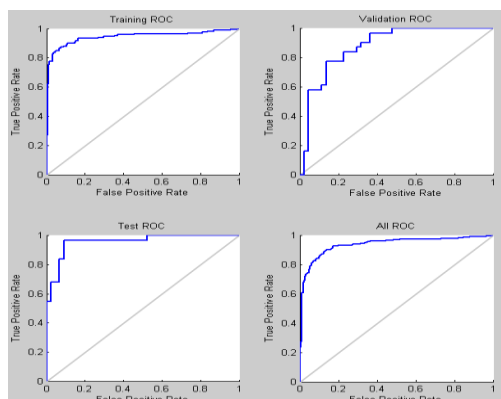

Figure 13: The Receiver Operating Characteristic.

The percent correctly classified in the simulation sample by the feed-forward back propagation network is approximately 86 percent as shown in Figure 14. Figure 15 shows the ROC curve for simulation sample.
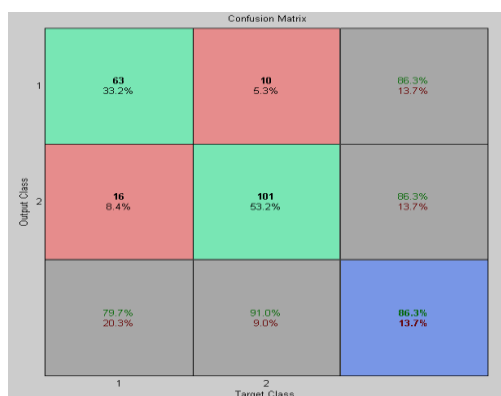

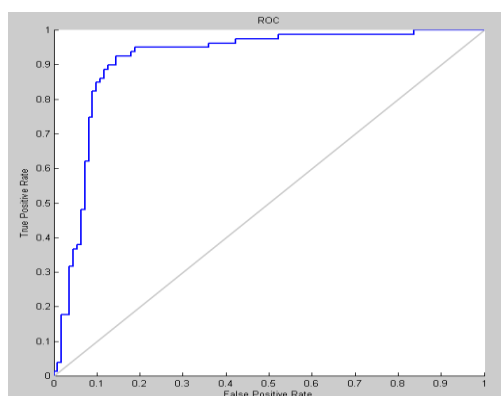**Figure 14:** The feed-forward back propagation network.

Figure 15: The ROC curve for simulation sample

4.0   CONCLUSIONS

This study aimed to test neural network model with learning algorithm: the feed-forward back propagation neural network with supervised learning in terms of their ability to classify. The results of applying the proposed neural networks to classify the credit risk if it is worthy or not worthy. Artificial neural networks showed significant results in classification of the application. The Receiver Operating Characteristic (ROC) curve is plotting to represent the classifier performance more closely.

REFERENCES
[1]   Al-Shayea, Q.K., El-Refae, G.A. and El-Itter, S.F. (2010): Neural Networks in Bank Insolvency Prediction. *International Journal of Computer Science and Network Security*, Vol. 10, No. 5, pp. 240-245.
[2]   Bahrammirzaee, A. (2010): A Comparative Survey of Artificial Intelligence Applications in Finance: Artificial Neural Networks, Expert System and Hybrid Intelligent Systems. *Neural Computing & Applications,* Vol. 19 No. 8, pp.1165-1195.
[3]   Bahrammirzaee, A., Ghatari, A., Ahmadi, P. and Madani, K. (2011): Hybrid Credit Ranking Intelligent System Using Expert System and Artificial Neural Networks. *Applied Intelligence*, Vol. 34 No.1, pp. 28-46.
[4]   Freeman, J.A. and Skapura, D.M. (1991): *Neural Networks: Algorithms, Applications and Programming Techniques*, Addison Wesley Longman.
[5]   Kashman, A. (2010): Neural Networks for Credit Risk Evaluation: Investigation of Different Neural Models and Learning Schemes. *Expert Systems with Applications*, Vol. 37 No.9, pp. 6233-6239.
[6]   Kashman, A. (2009): A Neural Network Model for Credit Risk Evaluation**.** *International Journal of Neural Systems*, Vol. 19, No. 4, pp. 285–294.
[7]   Marcano-Cedeno, A., Marin-De-La-Barcena, A., Jimenez-Trillo, J., Pinuela, J. and Andina, D. (2011): Artificial Metaplasticity Neural Network Applied to Credit Scoring. *International Journal of Neural Systems,* Vol. 21, No.4, pp. 311-317.

[8] Moller, M.F. (1993): A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning. *Neural Networks*, Vol. 6, pp. 525-533.

[9] Zhang, G.P. (2004): *Neural Networks in Business Forecasting*. Idea Group Inc.