Applied Soft Computing 11 (2011) 5477–5484

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

# Credit risk evaluation using neural networks: Emotional versus conventional models

Adnan Khashman *

*Intelligent Systems Research Group (ISRG), Near East University, Lefkosa, Mersin 10, Turkey*

ARTICLE INFO

*Article history:*
Received 16 May 2009
Received in revised form
20 November 2010
Accepted 1 May 2011
Available online 17 May 2011

*Keywords:*
Credit scoring
Artificial neural networks
Emotional back-propagation algorithm (EmBP)
Emotional neural networks

ABSTRACT

Credit scoring and evaluation is one of the key analytical techniques in credit risk evaluation which has been an active research area in financial risk management. Artificial neural networks (NNs) have been considered to be accurate tools for credit analysis among others in the credit industry. Lately, emotional neural networks (EmNNs) have been suggested and applied successfully for pattern recognition. In this paper we investigate the efficiency of EmNNs and compare their performance to conventional NNs when applied to credit risk evaluation. In total 12 neural networks; based equally on emotional and conventional neural models; are arbitrated under three learning schemes to classify whether a credit application is approved or declined. The learning schemes differ in the ratio of training-to-validation data used during training and testing the neural networks. The emotional and conventional neural models are trained using real world credit application cases from the Australian credit approval datasets which has 690 cases; each case with 14 numerical attributes; based on which an application is accepted or rejected. The performance of the 12 neural networks will be evaluated using certain criteria. Experimental results suggest that both emotional and conventional neural models can be used effectively for credit risk evaluations, however the emotional models outperform their conventional counterparts in decision making speed and accuracy, thus, making them ideal for implementation in fast automatic processing of credit applications.



## 1. Introduction

Credit scoring is a method of predicting potential risk corresponding to a credit portfolio. Models based on this method can be used by financial institutions to evaluate portfolios in terms of risk. Credit risk analysis is an important topic in the financial risk management, and has been the major focus of financial and banking industry [1]. Data mining methods, especially pattern classification, using real-world historical data, is of paramount importance in building such predictive models.

Due to financial crises and regulatory concerns of the Basel Committee on Banking Supervision [2,3], a regulatory requirement was made for the banks to use sophisticated credit scoring models for enhancing the efficiency of capital allocation. The Basel Committee, comprised of central bank and banking business representatives from various countries, formulated broad supervisory standards and guidelines for banks to implement. Due to changes in the banking business, risk management practices, supervisor approaches, and financial markets, the committee published a revised framework as the new capital adequacy framework, also known as Basel II [3]. The commencement of the Basel II requirement, popularization of consumer loans and the intense competition in financial market has increased the awareness of the critical delinquency issue for financial institutions in granting loans to potential applicants [4].

Credit scoring tasks can be divided into two distinct types [4–6]. The first type is application scoring, where the task is to classify credit applicants into “good” and “bad” risk groups. The data used for modeling generally consists of financial information and demographic information about the loan applicant. In contrast, the second type of tasks deals with existing customers and along with other information, payment history information is also used here. This is distinguished from the first type because this takes into account the customer’s payment pattern on the loan and the task is called behavioral scoring. In this paper, we shall focus on application scoring.

In credit scoring; a scorecard model lists a number of questions (called characteristics) for loan applicants who provide their answers based on a set of possible answers (called attributes). As a credit scoring method, neural network models are quite flexible as they allow the characteristics to be interacted in a variety of ways. They consist of a group or groups of connected characteristics. A single characteristic can be connected to many other characteristics, which make up the whole complicated network structure. They outweigh decision trees and scorecards because they do not assume uncorrelated relations between characteristics. They also do not suffer from structural instability in the same way as deci-

* Tel.: +90 392 223 6464; fax: +90 392 223 6622.
*E-mail addresses:* khashman@ieee.org, amk@neu.edu.tr

1568-4946/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.asoc.2011.05.011

sion trees because they may not rely on a single first question for constructing the whole network. However, the development of the network relies heavily on the qualitative data that are solicited to specify the interactions among all characteristics [7].

The use of neural networks in business applications has been previously investigated by several works [6,8–28]. More recently, the work in [29] used Kohonen and error back-propagation neural networks as consumer credit scoring models for financial institutions where data usually used in previous research is not available. This work suggested that the error back-propagation neural network showed the best results. In another work [30], three two-stage hybrid models of logistic regression-artificial neural network was proposed to construct a financial distress warning system suitable for Taiwan's banking industry, and to provide an optimal model of credit risk for supervising authorities, analysts and practitioners in conducting risk assessment and decision making. In [31], a back propagation based neural network was used to classify credit applicants. In [32], a credit scoring algorithm based on support vector machines was proposed to decide whether a bank should provide a loan to the applicant. In [33], a reassigning credit scoring model involving two stages was proposed. The classification stage is constructing an ANN-based credit scoring model, which classifies applicants with accepted (good) or rejected (bad) credits. The reassign stage is trying to reduce the Type I error by reassigning the rejected good credit applicants to the conditional accepted class by using the CBR-based classification technique. Lately, in [34,35] we investigated the use of conventional neural networks for the task credit risk evaluation with successful results that indicate the potential of using supervised neural models for such a task.

In general, the use of neural networks for credit scoring and evaluation has been shown to be effective over the past decade. The capability of neural networks in such applications is due to the way the network operates, and the availability of training data. This is more evident when using multilayer perceptron networks based on the error back propagation learning algorithm [36]. When feeding the information from a credit applicant to the neural network, attributes (applicant's answers to a set of questions (or characteristics)) are taken as the input to the neural network and a linear combination of them is taken with arbitrary weights. The attributes are linearly combined and subject to a non-linear transformation represented by a certain activation function (sigmoid function in this work), then fed as inputs into the next layer for similar manipulation. The final function yields values which can be compared with a cut-off for classification. Each training case is submitted to the network, the final output compared with the observed value and the difference, the error, is propagated back though the network and the weights modified at each layer according to the contribution each weight makes to the error value [37]. In essence the network takes data in attributes space, transforms it using the weights and activation functions into hidden value space and then possibly into further hidden value space; if further layers exist, and eventually into output layer space which is linearly separable.

Despite their successful application to credit scoring and evaluation, neural networks may not deliver robust "judgment" on whether an applicant should be granted credit or not. This problem arises from different reasons and partly depends on the chosen real world dataset for training and validating the trained neural network. Many of the previous works, which we described earlier on in this paper, suffer from problems despite the demonstrated successful implementations of the neural networks.

The first problem when using neural networks is the use of a high ratio of training-to-validation datasets. Depending on which dataset is used (the Australian credit dataset [38] is used in this work), a high ratio of training-to-validation data may not yield meaningful learning; for example, previously adopted ratios of training-to-validation (training:validation) datasets include: 80%:20% [4,24], 71%:29% [39], 68%:32% [17], 70%:30% [17,27,40,41], 69%:31% [29], 67%:33.3% [28], and 62%:38% [42]. A more appropriate ratio would be closer to 50%:50% as used in [43], or a lower ratio of training-to-testing dataset; i.e. less training and more validation data.

The second problem with using neural networks for credit evaluation is normalization of the input data. The values fed to the input layer of a neural network are usually between '0' and '1'. This is not a problem when using a neural network for image processing for example, since all input values would be representing the image pixel values, which in turn have a more uniform distribution and a finite difference between the lowest and the largest pixel value [44]. However, with credit evaluation, the numerical values (input values) representing the attributes of a credit applicant vary marginally in value, and if a simple normalization process is applied to the whole dataset, say by dividing each value in the set by the largest recorded value, then much information would be lost across the different attributes. For example, the highest value recorded in the Australian dataset is 100,001 (case 501, attribute 14); if all values within the dataset are divided by this maximum value, much of the input data would be closer to '0' value, which does not represent the attributes, thus leading to inefficient neural network training. Therefore, normalization of the credit application input data should be carefully performed, while maintaining the meaning of each attribute.

Another problem with using neural networks in financial applications is the computational cost. The simplest multilayer perceptron neural network has three layers (input, hidden and output). Much of the previously suggested neural network models for credit evaluation use two hidden layers. The problem here is the more layers are added, the higher the computational cost is, and thus, the higher the processing time.

In this paper, we aim to address the above problems when designing neural network (NN) models with application to credit risk evaluation. Moreover, we follow a novel approach to using neural networks for credit evaluation; namely by using an emotional neural network (EmNN) model. This approach is inspired by the fact that credit application cases vary marginally from one applicant to another. Normally, the decision upon accepting or rejecting a credit application is made by a human expert. When we employ conventional NNs to make the decision, they can be efficient, but they lack the human emotional factors. We intend to investigate the effects of these emotional factors, as modeled in the EmNNs, with applications like credit evaluation, and compare the performance of the EmNN to a conventional supervised neural network based on the back propagation (BP) learning algorithm [45].

The EmNN is based on the EmBP learning algorithm [46]. This emotional neural network has two emotional responses (anxiety and confidence) that change during the learning phase. If learning is meaningful, then the anxiety level decreases while the confidence levels increases as learning progresses. There are also two sets of extra emotional weights which conventional NNs do not possess; these together with the conventional weights form the memory of the EmNN after training. Emotional neural networks have been successfully implemented to various application involving pattern recognition and classification [47,48].

In our investigation we use the Australian credit dataset [38] that contains 690 real world application-decision cases. To address the general problem of learning data ratios, we train 12 neural networks (six emotional and six conventional models) using three learning schemes. The learning schemes differ in the ratio of training-to-validation ratios; or learning data ratios. The three schemes are: LS1 (43.5%:56.5%), LS2 (50%:50%), and LS3 (56.5%:43.5%). The lower the ratio, the more challenging it is for a neural network, but the more robust and meaningful the learning is. We compare the performance of the 12 neural network models

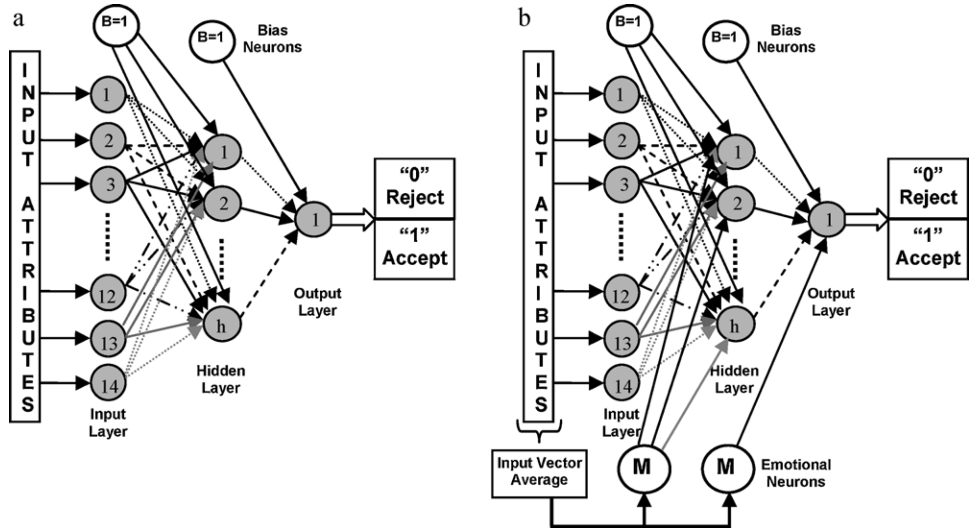


**Fig. 1.** Credit evaluation system neural network topologies: (a) conventional model – NN and (b) emotional model – EmNN.

under the different learning schemes and then determine the ideal neural model and learning scheme. As a solution for the data normalization problem, we use a simple but efficient normalization procedure that is applied automatically when reading the input data for each numerical attribute value separately. This assures that the 14 input values representing the different attributes are meaningful for the neural network models after normalization. Finally, we maintain simplicity when designing the emotional and conventional neural network models, by using a single hidden layer, and a single neuron at the output layer; thus minimizing the computational and time costs.

The structure of the paper is as follows: in Section 2 a brief explanation of the credit risk evaluation dataset is presented. In Section 3 the credit evaluation system is described; showing input data normalization procedure and the design strategy of the different neural network models. In Section 4 the results of training and testing (validating) the neural models using the different learning schemes are presented; and a comparison between the 12 neural models implementation results is provided. Finally, Section 5 concludes this work and suggests future work.

## 2. Credit risk evaluation dataset

For the implementation of experiments we use the Australian credit dataset; available publicly at UCI Machine Learning data repository [38]. This is a real world dataset that has been successfully used for credit scoring and evaluation systems in many previous works [4,22,23,27,29,39,43,49].

The Australian credit data consists of 307 instances of creditworthy applicants ("good", "accept") and 383 instances where credit is not creditworthy ("bad", "reject"). Each instance or case contains six nominal, eight numeric attributes (inputs), and one class attribute (output: accept or reject) – see Table 1. This dataset is interesting because there is a good mixture of attributes: continuous, nominal with small numbers of values, and nominal with larger numbers of values. There are also a few missing values. To protect the confidentiality of data, the attributes names and values have been changed to meaningless symbolic data. The details of the attributes are available at the repository [38]. Table 2 shows examples of the dataset attributes' numerical representation for the first 10 cases; these numerical values are not normalized. Once normalization to values between "0" and "1" is completed, the values are used as the input data to the neural network. The dataset is divided into two subsets in our work; the training dataset and the validation or testing dataset. We follow three learning schemes with different training-to-validation data ratios in order investigate an ideal learning data ratio for the implementation of the credit risk evaluation system.

## 3. Credit risk evaluation system

A neural network-based credit risk evaluation system would consist of two phases. Firstly, data normalization phase, where each attribute (A1–A14) within the dataset is normalized separately. The output of this phase provides normalized numerical values representing a credit applicant's case, which is used in the second phase; evaluating the applicant's attributes and deciding whether to accept or reject the application using a neural network model. Once the neural network converges to a set value of minimum error and learning is accomplished, the second phase consists only of one forward pass that yields the evaluation result.

### 3.1. Input data normalization

This phase is a data preparation phase for neural network training and classification/evaluation. Here, the input data (attribute numerical values) are separately normalized to values between "0" and "1". This is achieved by finding the maximum or highest value within each attribute for all 690 cases in the dataset, and dividing all the values within that same attribute by the maximum. Table 3 shows the maximum values for each input attribute. The highest value in the whole set belongs to attribute A14 (100,001) which is a large value in comparison to the other values in the set, and thus, if used for normalizing the remaining input data values, it would lead to normalized values close to "0". This could cause insufficient learning of the neural network, as the applicant's attributes would not be appropriately represented. Table 4 shows the normalized input values (attributes) of the first 10 cases in the dataset.

### 3.2. Neural network arbitration

During this phase a supervised neural network (emotional or conventional) is used. The emotional neural network (EmNN) is based on the emotional back propagation (EmBP) learning algorithm [46], whereas the conventional neural network is based on the back propagation (BP) learning algorithm [45]. The back propagation-based algorithms are chosen due to their implementation simplicity and the availability of sufficient dataset for training

**Table 1**
Australian credit data set attributes. Inputs: (A1–A14), output (A15). The labels have been changed for the convenience of the statistical algorithms. For example, attribute 4 originally had 3 labels p, g, gg and these have been changed to labels 1, 2, 3 [38].

| Attribute | Type and/or value |
|---|---|
| A1 | 0, 1 categorical (formerly a, b) |
| A2 | Continuous |
| A3 | Continuous |
| A4 | 1, 2, 3 categorical (formerly p, g, gg) |
| A5 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14 categorical (formerly ff, d, i, k, j, aa, m, c, w, e, q, r, cc, x) |
| A6 | 1, 2, 3, 4, 5, 6, 7, 8, 9 categorical (formerly ff, dd, j, bb, v, n, o, h, z) |
| A7 | Continuous |
| A8 | 1, 0 categorical (formerly t, f) |
| A9 | 1, 0 categorical (formerly t, f) |
| A10 | Continuous |
| A11 | 1, 0 categorical (formerly t, f) |
| A12 | 1, 2, 3 categorical (formerly s, g, p) |
| A13 | Continuous |
| A14 | Continuous |
| A15 | 1, 0 class attribute (formerly +, −) |

**Table 2**
Examples of pre-normalization numerical values representing input/output attributes for the first 10 cases in the Australian credit dataset [38].

| Cases | Input attributes | | | | | | | | | | | | | | Output |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 | A13 | A14 | A15 |
| 1. | 1 | 22.08 | 11.46 | 2 | 4 | 4 | 1.585 | 0 | 0 | 0 | 1 | 2 | 100 | 1213 | 0 |
| 2. | 0 | 22.67 | 7 | 2 | 8 | 4 | 0.165 | 0 | 0 | 0 | 0 | 2 | 160 | 1 | 0 |
| 3. | 0 | 29.58 | 1.75 | 1 | 4 | 4 | 1.25 | 0 | 0 | 0 | 1 | 2 | 280 | 1 | 0 |
| 4. | 0 | 21.67 | 11.5 | 1 | 5 | 3 | 0 | 1 | 1 | 11 | 1 | 2 | 0 | 1 | 1 |
| 5. | 1 | 20.17 | 8.17 | 2 | 6 | 4 | 1.96 | 1 | 1 | 14 | 0 | 2 | 60 | 159 | 1 |
| 6. | 0 | 15.83 | 0.585 | 2 | 8 | 8 | 1.5 | 1 | 1 | 2 | 0 | 2 | 100 | 1 | 1 |
| 7. | 1 | 17.42 | 6.5 | 2 | 3 | 4 | 0.125 | 0 | 0 | 0 | 0 | 2 | 60 | 101 | 0 |
| 8. | 0 | 58.67 | 4.46 | 2 | 11 | 8 | 3.04 | 1 | 1 | 6 | 0 | 2 | 43 | 561 | 1 |
| 9. | 1 | 27.83 | 1 | 1 | 2 | 8 | 3 | 0 | 0 | 0 | 0 | 2 | 176 | 538 | 0 |
| 10. | 0 | 55.75 | 7.08 | 2 | 4 | 8 | 6.75 | 1 | 1 | 3 | 1 | 2 | 100 | 51 | 0 |

**Table 3**
The highest value in each attribute of the 690 cases in the dataset. These values are used to normalize the input data prior to neural network processing.

| Input attribute | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 | A13 | A14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Maximum value | 1 | 80.25 | 28 | 3 | 14 | 9 | 28.5 | 1 | 1 | 67 | 1 | 3 | 2000 | 100, 001 |

and validation. Fig. 1 shows the topologies of the credit evaluation neural network models.

A total of 12 (six emotional and six conventional) neural network models are arbitrated to solve the credit risk evaluation problem. The input layers of all neural network models have 14 neurons each, which is according to the number of the credit applicant's attributes; each input neuron receives a normalized numerical value representing an attribute.

There is one hidden layer containing $h$ neurons; this number changes depending on the neural network model and is chosen after several experiments involving the adjustment of the number of hidden neurons from one to 50 neurons, with the aim of providing the neural model with meaningful learning while keeping the time cost to a minimum. The output layer has one single neuron, which uses binary output data representation; '1' for accepting or '0' for rejecting a credit application. A simple thresholding scheme is sufficient for a neural network's single output neuron to divide the feature space into two categories. A threshold value of 0.5 is used to distinguish between credit groups, good credit and bad credit. If the output result of the neural network is greater than or equal to 0.5, the presented case is assigned to one class (good, accept); otherwise it is assigned to the other class (bad, reject); hence, for conventional neural models:

$$\begin{aligned}&\text{Applicant}_i \in \text{Good credit class if}: NN_{out}(i) \geq 0.5\\&\text{Applicant}_i \in \text{Bad credit class if}: NN_{out}(i) < 0.5,\end{aligned} \quad (1)$$

**Table 4**
Normalized input data – attribute values for the first 10 cases in the Australian credit dataset.

| | Normalized input attributes | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 | A13 | A14 |
| 1. | 1 | 0.275140 | 0.409286 | 0.666667 | 0.285714 | 0.444444 | 0.055614 | 0 | 0 | 0.000000 | 1 | 0.666667 | 0.050000 | 0.012130 |
| 2. | 0 | 0.282492 | 0.250000 | 0.666667 | 0.571429 | 0.444444 | 0.005789 | 0 | 0 | 0.000000 | 0 | 0.666667 | 0.080000 | 0.000010 |
| 3. | 0 | 0.368598 | 0.062500 | 0.333333 | 0.285714 | 0.444444 | 0.043860 | 0 | 0 | 0.000000 | 1 | 0.666667 | 0.140000 | 0.000010 |
| 4. | 0 | 0.270031 | 0.410714 | 0.333333 | 0.357143 | 0.333333 | 0.000000 | 1 | 1 | 0.164179 | 1 | 0.666667 | 0.000000 | 0.000010 |
| 5. | 1 | 0.251340 | 0.291786 | 0.666667 | 0.428571 | 0.444444 | 0.068772 | 1 | 1 | 0.208955 | 0 | 0.666667 | 0.030000 | 0.001590 |
| 6. | 0 | 0.197259 | 0.020893 | 0.666667 | 0.571429 | 0.888889 | 0.052632 | 1 | 1 | 0.029851 | 0 | 0.666667 | 0.050000 | 0.000010 |
| 7. | 1 | 0.217072 | 0.232143 | 0.666667 | 0.214286 | 0.444444 | 0.004386 | 0 | 0 | 0.000000 | 0 | 0.666667 | 0.030000 | 0.001010 |
| 8. | 0 | 0.731090 | 0.159286 | 0.666667 | 0.785714 | 0.888889 | 0.106667 | 1 | 1 | 0.089552 | 0 | 0.666667 | 0.021500 | 0.005610 |
| 9. | 1 | 0.346791 | 0.035714 | 0.333333 | 0.142857 | 0.888889 | 0.105263 | 0 | 0 | 0.000000 | 0 | 0.666667 | 0.088000 | 0.005380 |
| 10. | 0 | 0.694704 | 0.252857 | 0.666667 | 0.285714 | 0.888889 | 0.236842 | 1 | 1 | 0.044776 | 1 | 0.666667 | 0.050000 | 0.000510 |

**Table 5**
Conventional neural networks final training parameters (required error = 0.007, maximum allowed iterations = 25,000).

| Neural network model | NN-1 | NN-2 | NN-3 | NN-4 | NN-5 | NN-6 |
|---|---|---|---|---|---|---|
| Input layer nodes | 14 | 14 | 14 | 14 | 14 | 14 |
| Hidden layer nodes | 10 | 9 | 10 | 9 | 10 | 9 |
| Output layer nodes | 1 | 1 | 1 | 1 | 1 | 1 |
| Learning coefficient | 0.0075 | 0.00935 | 0.0085 | 0.009 | 0.0077 | 0.0068 |
| Momentum rate | 0.90 | 0.67 | 0.87 | 0.88 | 0.89 | 0.90 |
| Obtained error | 0.0161 | 0.007 | 0.0093 | 0.007 | 0.0224 | 0.007 |
| Performed iterations | 25,000 | 13,841 | 25,000 | 6292 | 25,000 | 16,592 |
| Learning data ratio | 300:390 | 300:390 | 345:345 | 345:345 | 390:300 | 390:300 |

and for emotional neural models:

$$\begin{array}{l}\text{Applicant}_i \in \text{Good credit class if}: EmNN_{out}(i) \geq 0.5 \\ \text{Applicant}_i \in \text{Bad credit class if}: EmNN_{out}(i) < 0.5,\end{array} \quad (2)$$

where $NN_{out}(i)$ is the output of the conventional neural network and $EmNN_{out}(i)$ is the output of the emotional neural network; obtained when the attributes of the *i*th case (applicant) are presented to the network. This is basically the output credit decision associated with applicant *i*.

During the learning phase, the number of hidden layer neurons, the learning coefficient, and the momentum rate were adjusted during various experiments in order to achieve the required minimum error value of 0.007 which was considered as sufficient for this application. Table 5 lists the final parameters of the six conventional neural network models, whereas Table 6 lists the final parameters of the six emotional neural network models.

## 4. Implementation, analysis and experimental results

As one of our objectives is to investigate an ideal training-to-validation data ratio (or learning data ratio), we follow three learning schemes to train the neural network models. The learning schemes differ in their learning data ratios. For example, learning scheme 1 (LS1) uses a ratio of (300:390); i.e. the first 300 credit application cases are used for training the network, while the remaining 390 cases are not exposed to the neural network during training; as they are used to test or validate the network's classification capability upon completion of training. The training-to-validation ratios; or learning data ratios were 300:390 (43.5%:56.5%), 345:345 (50%:50%), and 390:300 (56.5%:43.5%). These learning data ratios were chosen as they are closer or equal to 50%:50% ratio which we consider as appropriate. A high learning data ratio (i.e. more training data than validation) may yield incomprehensive biased learning of the network model; whereas, a low learning data ratio (i.e. less training data than validation) may not provide the neural network with meaningful learning.

There are two questions that arise when analyzing the work in this paper. Firstly, why using 12 neural networks, and secondly, how do we evaluate their performance. To clarify the first question, there are 12 neural networks based equally on six emotional models and six conventional models. This number of implemented models is a result of our approach to investigating an ideal learning scheme and neural model for credit risk evaluation. In our approach we first define the learning schemes (LS1, LS2, and LS3) based on the three different learning data ratios. We then train and validate both emotional and conventional neural network models under each scheme. It was observed that emotional and conventional models perform differently under different schemes and with different learning parameters. Therefore, under each learning scheme, we found the optimum configuration for; say the emotional model, then trained the conventional model using the same configuration. The experiments are repeated to find an optimum configuration for the conventional model, and then apply the same configuration using the emotional model. As a result, four (two emotional and two conventional) neural networks are implemented under each of the three schemes; in total 12 neural networks are investigated.

The second question concerns our method of evaluating the experimental results. To compare the performance of the 12 neural networks under the three learning schemes, performance evaluation criteria are required. Our first criterion was based on training the neural networks for a chosen number of iterations; 25,000 iterations were considered sufficient, as a higher number of iterations would lead to increasing the computational costs.

The second criterion was based on the network converging to the required error value of 0.007; this indicates the accuracy of the trained neural model in classifying credit applications. The third criterion was the run time of the trained neural network; this is the time taken for a trained model to perform one forward pass, and to classify whether a credit application should be accepted or rejected. The final criterion, which is of utmost importance, is the validation dataset (V-dataset) accuracy rate; this is the accuracy rate of a trained neural model using credit application cases that were not exposed to the neural network during training. Testing a trained neural model using the validation dataset provides more objective indication to whether the neural model learning is robust or not.

Table 7 lists in detail the obtained results implementing the 12 neural models under the three learning schemes. Upon inspecting this table, we observe that emotional neural models (EmNN-1, EmNN-3, EmNN-5) and conventional neural models (NN-2, NN-4, NN-6) achieved the first and second performance evaluation criteria by converging to the required error value of 0.007 within

**Table 6**
Emotional neural networks final training parameters (required error = 0.007, maximum allowed iterations = 25,000).

| Emotional neural model | EmNN-1 | EmNN-2 | EmNN-3 | EmNN-4 | EmNN-5 | EmNN-6 |
|---|---|---|---|---|---|---|
| Input layer nodes | 14 | 14 | 14 | 14 | 14 | 14 |
| Hidden layer nodes | 10 | 9 | 10 | 9 | 10 | 9 |
| Output layer nodes | 1 | 1 | 1 | 1 | 1 | 1 |
| Learning coefficient | 0.0075 | 0.00935 | 0.0085 | 0.009 | 0.0077 | 0.0068 |
| Momentum rate | 0.90 | 0.67 | 0.87 | 0.88 | 0.89 | 0.90 |
| Obtained error | 0.007 | 0.0172 | 0.007 | 0.0162 | 0.007 | 0.0161 |
| Performed iterations | 11,834 | 25,000 | 17,615 | 25,000 | 8407 | 25,000 |
| Anxiety level | 0.010713 | 0.020957 | 0.010718 | 0.019887 | 0.010713 | 0.019868 |
| Confidence level | 0.248696 | 0.274807 | 0.248354 | 0.274975 | 0.247747 | 0.273451 |
| Learning data ratio | 300:390 | 300:390 | 345:345 | 345:345 | 390:300 | 390:300 |

**Table 7**
The three learning schemes and implementation results of the 12 credit risk evaluation neural network models. T-dataset: training dataset, V-dataset: validation dataset.

| Learning scheme and data ratio | Neural network model | Obtained error | Training time[a] (s) | Run time[a] (s) | T-dataset accuracy rate | V-dataset accuracy rate | Overall rate[b] (%) |
|---|---|---|---|---|---|---|---|
| LS1 (300:390) | EmNN-1 | 0.007 | 83.34 | $3.85 \times 10^{-5}$ | 297/300 (99%) | 316/390 (81.03%) | 88.84 |
| | EmNN-2 | 0.0172 | 165.58 | $3.85 \times 10^{-5}$ | 295/300 (98.33%) | 307/390 (78.72%) | 87.25 |
| | NN-1 | 0.0161 | 98.06 | $4.11 \times 10^{-5}$ | 294/300 (98%) | 301/390 (77.18%) | 86.23 |
| | NN-2 | 0.007 | 51.16 | $4.11 \times 10^{-5}$ | 300/300 (100%) | 309/390 (79.23%) | 88.26 |
| LS2 (345:345) | EmNN-3 | 0.007 | 131.5 | $8.99 \times 10^{-5}$ | 343/345 (99.42%) | 276/345 (80%) | 89.71 |
| | EmNN-4 | 0.0162 | 176.28 | $8.99 \times 10^{-5}$ | 340/345 (98.55%) | 275/345 (79.71%) | 89.13 |
| | NN-3 | 0.0093 | 102.03 | $9.28 \times 10^{-5}$ | 342/345 (99.13%) | 269/345 (77.97%) | 88.55 |
| | NN-4 | 0.007 | 25.89 | $9.28 \times 10^{-5}$ | 344/345 (99.71%) | 274/345 (79.42%) | 89.57 |
| LS3 (390:300) | EmNN-5 | 0.007 | 67.49 | $5.33 \times 10^{-5}$ | 387/390 (99.23%) | 233/300 (77.67%) | 89.86 |
| | EmNN-6 | 0.0199 | 180.84 | $5.33 \times 10^{-5}$ | 384/390 (98.46%) | 242/300 (80.67%) | 90.72 |
| | NN-5 | 0.0224 | 112.97 | $15.67 \times 10^{-5}$ | 381/390 (97.69%) | 237/300 (79%) | 89.57 |
| | NN-6 | 0.007 | 69.67 | $15.67 \times 10^{-5}$ | 388/390 (99.49%) | 241/300 (80.33%) | 91.16 |

[a] Using a 2.8 GHz PC with 2 GB of RAM, Windows XP Operating System and C++ programming language.
[b] Overall accuracy rate is obtained by combining accuracy rates of training and validation datasets.

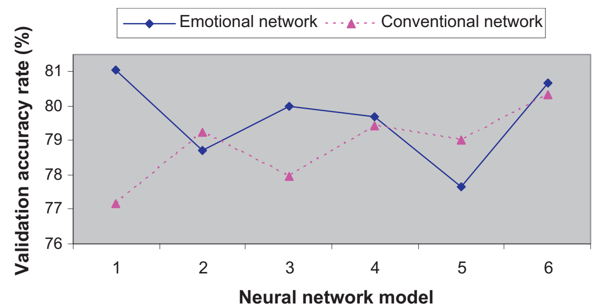


**Fig. 2.** Credit risk evaluation accuracy rates using validation datasets.

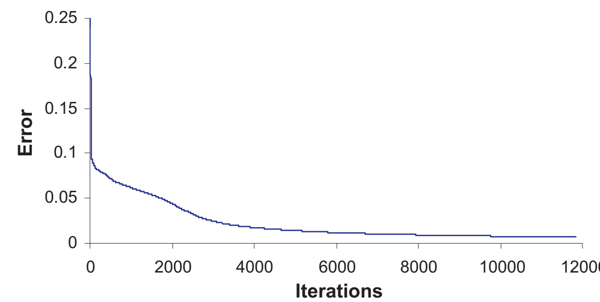


**Fig. 3.** Learning curve of emotional neural network (EmNN-1) under learning scheme LS1.

the maximum allowed 25,000 iterations. Out of these six neural models, and considering the third criterion of the quickest run time, the emotional neural network model (EmNN-1) outperforms the other five models yielding a credit application decision within $3.85 \times 10^{-5}$ s.

When considering the fourth criterion (V-dataset accuracy rate), again emotional model (EmNN-1) yields the highest accuracy rate of 81.03%. Fig. 2 shows the validation dataset accuracy rates for the 12 neural models. It is worth noticing that overall accuracy rates (see Table 7) can be misleading, as they combine training dataset (T-dataset) accuracy rates with validation accuracy rates. T-dataset accuracy rates are obtained using data that has been exposed to the neural model during training, thus high T-dataset accuracy rates do not indicate that the neural model achieved meaningful learning. For example, an overall accuracy rate of 91.16% was obtained using conventional neural model (NN-6); this model, when compared to EmNN-1, is not considered as ideal because of its higher learning data ratio, higher run time, and lower V-dataset accuracy rate.

Based on these results and the performance evaluation criteria, the emotional neural network model EmNN-1 is considered as

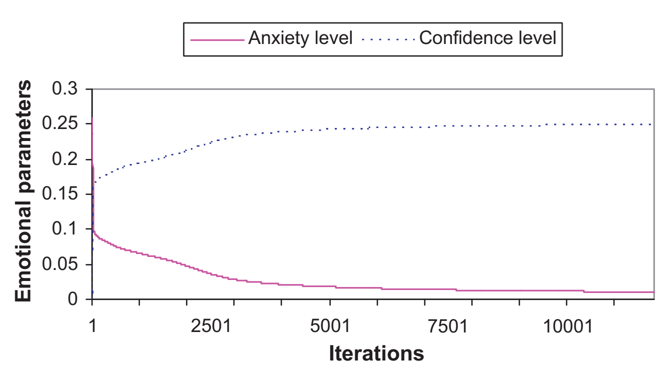


**Fig. 4.** Emotional parameters changes during EmNN-1 learning.

the most efficient when used for credit risk evaluation. The EmNN-1 emotional neural network model converged to the required error of 0.007 after 11,834 iterations and within approximately 84 s, whereas the run time for this emotional model was a fast $3.85 \times 10^{-5}$ s. The training dataset accuracy rate was 99%, the validation dataset accuracy rate was 81.03%, and the overall accuracy rate was 88.84%. Notice also that among the six emotional neural models, EmNN-1 has the least level of anxiety. Fig. 3 shows the learning curve of EmNN-1, whereas Fig. 4 shows the changes in anxiety and confidence levels during learning. Finally, we can also determine the ideal learning scheme for this particular application, which was LS1; where the training-to-validation data ratio is 300:390 (43.5%:56.5%).

## 5. Conclusions

This paper presented an investigation of the use of supervised neural network models for credit risk evaluation under different learning schemes. We also propose an efficient, fast and simple-to-use credit evaluation system, based on the results of our investigation. In our investigation we trained 12 neural networks; based on emotional and conventional neural models under three learning schemes. These schemes differ in the ratio of the number of credit application cases used for training, against those used for validation or testing. The use of high training-to-validation dataset ratio, or learning data ratio, has been a common problem with much of the previous research works on using neural networks for credit risk evaluation as discussed in Section 1. Another problem was the normalization of credit application input data, where we proposed a simple but efficient method of normalizing the input data (attributes of a credit applicant) prior to presenting it to the neural networks. We also maintained simplicity in designing the neural network models, in order to keep the computational and time costs to a minimum, thus, the use of one hidden layer; many previous works generally use two hidden layers for similar applications.

In order to implement the different neural networks for solving the problem of credit risk evaluation, we used the Australian credit dataset, which is a real world dataset that has been successfully used for credit scoring and evaluation systems in many previous works. The Australian credit data consists of 690 cases or instances: 307 instances of creditworthy applicants (“good” or “accept”), and 383 instances where credit is not creditworthy (“bad” or “reject”).

We also described in detail (Section 4) the criteria and considerations which are to be made in order to decide upon an ideal learning scheme and neural model. The criteria included maximum number of iterations during training, minimum error requirement, speed of trained neural models when making a decision on a credit application, and finally, the accuracy rate of the neural model.

Having compared the implementation results of the 12 neural networks; based on the performance evaluation criteria, it was concluded that emotional neural network model (EmNN-1) outperforms the remaining 11 neural networks when considering the four above criteria. The EmNN-1 was trained under learning scheme LS1 with training-to-validation data ratio of 300:390 (43.5%:56.5%), and is capable of deciding upon a credit application case within $3.85 \times 10^{-5}$ s, with a validation dataset accuracy rate of 81.03%. Therefore, we suggest that this emotional neural system for credit risk evaluation can be efficiently used in automatic processing of credit applications. Future work will focus on designing, training and implementing both emotional and conventional neural systems which could indicate the reason why a credit application had not been approved.

## Acknowledgement

The author would also like to thank Ross Quinlan for publicly providing the Australian Credit Approval Dataset at the UCI Machine Learning Repository.

## References

[1] K.K. Lai, L. Yu, S.Y. Wang, L.G. Zhou, Credit risk analysis using a reliability-based neural network ensemble model, Lect. Notes. Comput. Sci. 4132 (2006) 682–690.
[2] Basel Committee on Banking Supervision, Principles for the Management of Credit Risk, Basel, Switzerland, 2000, p. 11, http://www.bis.org/publ/bcbs75.pdf?noframes=1.
[3] Basel Committee on Banking Supervision, International Convergence of Capital Management and Capital standards: A Revised Framework, Basel, Switzerland, 2005, p. 15, http://www.bis.org/publ/bcbs118.htm.
[4] S.T. Li, W. Shiue, M.H. Huang, The evaluation of consumer loans using support vector machines, Expert Syst. Appl. 30 (4) (2006) 772–782.
[5] A. Laha, Building contextual classifiers by integrating fuzzy rule based classification technique and *k*-nn method for credit scoring, Adv. Eng. Inform. 21 (2007) 281–291.
[6] A. Vellido, P.J.G. Lisboa, J. Vaughan, Neural networks in business: a survey of applications (1992–1998), Expert Syst. Appl. 17 (1) (1999) 51–70.
[7] E.W.L. Cheng, Y.H. Chiang, B.S. Tang, Alternative approach to credit scoring by DEA: evaluating borrowers with respect to PFI projects, Build Environ. 42 (2007) 1752–1760.
[8] K.A. Smith, Introduction to Neural Networks and Data Mining for Business Applications, Eruditions Publishing, Australia, 1999.
[9] B.S. Ahn, S.S. Cho, C.Y. Kim, The integrated methodology of rough set theory and artificial neural network for business failure prediction, Expert Syst. Appl. 18 (2) (2000) 65–74.
[10] T.S. Lee, C.C. Chiu, C.J. Lu, Credit scoring using the hybrid neural discriminant technique, Expert Syst. Appl. 23 (3) (2002) 245–254.
[11] I. Becerra-Fernandez, S.H. Zanakis, S. Walczak, Knowledge discovery techniques for predicting country investment risk, Comput. Ind. Eng. 43 (4) (2002) 787–800.
[12] R. Malhotra, D.K. Malhotra, Evaluating consumer loans using neural networks, Omega 31 (2) (2003) 83–96.
[13] B. Baesens, R. Sentiono, C. Mues, J. Vanthienen, Using neural network rule extraction and decision tables for credit-risk evaluation, Manage. Sci. 49 (3) (2003) 312–329.
[14] Z. Huang, H. Chen, C.J. Hsu, W.H. Chen, S. Wu, Credit rating analysis with support vector machines and neural networks: a market comparative study, Decis. Support Syst. 37 (4) (2004) 543–558.
[15] T.S. Lee, I.F. Chen, A two-stage hybrid credit scoring model using artificial neural networks and multivariate adaptive regression splines, Expert Syst. Appl. 28 (2005) 743–752.
[16] D. West, S. Dellana, J. Qian, Neural network ensemble strategies for financial decision applications, Comput. Oper. Res. 32 (10) (2005) 2543–2559.
[17] N.C. Hsieh, Hybrid mining approach in design of credit scoring model, Expert Syst. Appl. 28 (2005) 655–665.
[18] W. Huang, Y. Nakamori, S.Y. Wang, Forecasting stock market movement direction with support vector machine, Comput. Oper. Res. 32 (10) (2005) 2513–2522.
[19] J.H. Min, Y.C. Lee, Bankruptcy prediction using support vector machine with optimal choice of kernel function parameters, Expert Syst. Appl. 28 (4) (2005) 603–614.
[20] B. Baesens, T.V. Gestel, M. Stepanova, D. Van den Poel, J. Vanthienen, Neural network survival analysis for personal loan data, J. Oper. Res. Soc. 56 (9) (2005) 1089–1098.
[21] M.K. Lim, S.Y. Sohn, Cluster-based dynamic scoring model, Expert Syst. Appl. 32 (2) (2007) 427–431.
[22] D. Martens, B. Baesens, T.V. Gestel, J. Vanthienen, Comprehensible credit scoring models using rule extraction from support vector machines, Eur. J. Oper. Res. 183 (2007) 1466–1476.
[23] C.L. Huang, M.C. Chen, C.J. Wang, Credit scoring with a data mining approach based on support vector machines, Expert Syst. Appl. 33 (4) (2007) 847–856.
[24] H. Abdou, J. Pointon, A. Elmasry, Neural nets versus conventional techniques in credit scoring in Egyptian banking, Expert Syst. Appl. 35 (3) (2008) 1275–1292.
[25] E. Angelini, G. Di Tollo, A. Roli, A neural network approach for credit risk evaluation, Quart. Rev. Econ. Finance 48 (2008) 733–755.
[26] L. Yu, S.Y. Wang, K.K. Lai, Credit risk assessment with a multistage neural network ensemble learning approach, Expert Syst. Appl. 34 (2) (2008) 1434–1444.
[27] C.F. Tsai, J.W. Wu, Using neural network ensembles for bankruptcy prediction and credit scoring, Expert Syst. Appl. 34 (4) (2008) 2639–2649.
[28] R. Setiono, B. Baesens, C. Mues, Recursive neural network rule extraction for data with mixed attributes, IEEE Trans. Neural Netw. 19 (2) (2008) 299–307.
[29] M. MŠušteršic, D. Mramor, J. Zupan, Consumer credit scoring models with limited data, Expert Syst. Appl. 36 (3) (2009) 4736–4744.
[30] S.L. Lin, A new two-stage hybrid approach of credit risk in banking industry, Expert Syst. Appl. 36 (4) (2009) 8333–8341.

[31] C.M. Wang, Y.F. Huang, Evolutionary-based feature selection approaches with new criteria for data mining: A case study of credit approval data, Expert Syst. Appl. 36 (3) (2009) 5900–5908.
[32] X. Xu, C. Zhou, Z. Wang, Credit scoring algorithm based on link analysis ranking with support vector machine, Expert Syst. Appl. 36 (2) (2009) 2625–2632.
[33] C.L. Chuang, R.H. Lin, Constructing a reassigning credit scoring model, Expert Syst. Appl. 36 (2) (2009) 1685–1694.
[34] A. Khashman, A neural network model for credit risk evaluation, Int. J. Neural Syst. 19 (4) (2009) 285–294.
[35] A. Khashman, Neural networks for credit risk evaluation: Investigation of different neural models and learning schemes, Expert Syst. Appl. 37 (9) (2010) 6233–6239.
[36] S. Haykin, Neural Networks. A Comprehensive Foundation, 2nd ed., Prentice Hall, 1999.
[37] J.N. Crook, D.B. Edelman, L.C. Thomas, Recent developments in consumer credit risk assessment, Eur. J. Oper. Res. 183 (2007) 1447–1465.
[38] A. Asuncion, D.J. Newman, UCI Machine Learning Repository Irvine, School of Information and Computer Science, University of California, CA, 2007, http://www.ics.uci.edu/~mlearn/MLRepository.html.
[39] E. Boros, P.L. Hammer, T. Ibaraki, A. Kogan, E. Mayoraz, I. Muchnik, An implementation of logical analysis of data, IEEE Trans. Knowl. Data Eng. 12 (2) (2000) 292–306.
[40] B. Baesens, T.V. Gestel, S. Viaene, M. Stepanova, J. Suykens, J. Vanthienen, Benchmarking state-of-the-art classification algorithms for credit scoring, J. Oper. Res. 54 (2003) 627–635.
[41] Y.S. Kim, S.Y. Sohn, Managing loan customers using misclassification patterns of credit scoring model, Expert Syst. Appl. 26 (2004) 567–573.
[42] A.F. Atiya, Bankruptcy prediction for credit risk using neural networks: A survey and new results, IEEE Trans. Neural Netw. 12 (4) (2001) 929–935.
[43] S. Sakprasat, M.C. Sinclair, Classification rule mining for automatic credit approval using genetic programming, in: Proceedings of the IEEE Congress on Evolutionary Computation, Singapore, September 2007, 2007, pp. 548–555.
[44] A. Khashman, IBCIS: intelligent blood cell identification system, Prog. Nat. Sci. 18 (10) (2008) 1309–1314.
[45] D. Rumelhart, G.E. Hinton, R.J. Williams, in: D. Rumelhart, J. McClelland (Eds.), Learning Internal Representations by Error Propagation. Parallel Distributed Processing, vol. 1, MIT Press, 1986.
[46] A. Khashman, A modified back propagation learning algorithm with added emotional coefficients, IEEE Trans. Neural Netw. 19 (11) (2008) 1896–1909.
[47] A. Khashman, Application of an emotional neural network to facial recognition, Neural Comput. Appl. 18 (4) (2009) 309–320.
[48] A. Khashman, Blood cell identification using emotional neural networks, J. Inf. Sci. Eng. 25 (6) (2009) 1–16.
[49] B. Hamers, J.A.K. Suykens, B. De Moor, Coupled transductive ensemble learning of kernel models, J. Mach. Learn. Res. 1 (2003) 1–48.

**Adnan Khashman** received the B.Eng. degree in electronic and communication engineering from Birmingham University, England, UK, in 1991, and the M.S. and Ph.D. degrees in electronic engineering from Nottingham University, England, UK, in 1992 and 1997. During 1998–2001 he was an Assistant Professor and Chairman of the Computer Engineering Department, Near East University, Lefkosa, Turkey. During 2001–2009 he was an Associate Professor and Chairman of the Electrical and Electronic Engineering Department at the same university. From 2007 till 2008 he was also the Vice-Dean of the Engineering Faculty. Since 2009 he is a full Professor and the Head of the Intelligent Systems Research Group (ISRG) which he founded in 2001 at the same university. Current research interests include emotion modeling in neural networks and their applications. Contact him at: ISRG, Faculty of Engineering, Near East University, Lefkosa, Mersin 10, Turkey; khashman@ieee.org.