An Intelligent Approach to Educational Data: Performance Comparison of the Multilayer Perceptron and the Radial Basis Function Artificial Neural Networks

Article *in* Educational Sciences: Theory and Practice · November 2015

DOI: 10.12738/estp.2015.5.0238

Murat Kayri
Yuzuncu Yil University

ISSN 1303-0485 • eISSN 2148-7561
DOI 10.12738/estp.2015.5.0238

Educational Sciences: Theory & Practice • 2015 October • 16(1) • 1-10

Received | June 16, 2015
Accepted | September 17, 2015
OnlineFirst | October 2, 2015

An Intelligent Approach to Educational Data: Performance Comparison of the Multilayer Perceptron and the Radial Basis Function Artificial Neural Networks

Murat Kayri[a]
Muş Alparslan University

Abstract
The objective of this study is twofold: (1) to investigate the factors that affect the success of university students by employing two artificial neural network methods (i.e., multilayer perceptron [MLP] and radial basis function [RBF]); and (2) to compare the effects of these methods on educational data in terms of predictive ability. The participants' transcript scores were used as the target variables and the two methods were employed to test the predictors that affected these variables. The results show that the multilayer perceptron artificial neural network outperformed the radial basis artificial neural network in terms of predictive ability. Although the findings suggest that research in quantitative educational science should be conducted by using the former artificial neural network method, additional supporting evidence needs to be collected in related studies.

Keywords: Multilayer perceptron • Radial basis function • Data mining • Artificial neural network • Educational data

a **Correspondence**
Assoc. Prof. Murat Kayri (PhD), Department of Computer Engineering, Muş Alparslan University, Kampüs, Muş 49100 Turkey
Research areas: Statistical analysis; Data mining; Decision trees; Exploratory Factor Analysis; Confirmatory Factor Analysis; Artificial neural network
Email: muratkayri@gmail.com

An intelligence system basically consists of three layers (i.e., input, output, and hidden), which contain built-in neurons that connect one layer to another. In a neural network, basic functions are inferred from the data, which allow the network to understand complex interactions between predictor variables (Gonzalez-Camacho et al., 2012; Hastie, Tibshirani, & Friedman, 2009). Artificial neural networks (ANNs) are computational models based on parallel distributed processing, which can be used to model highly complex and non-linear stochastic problems such as the ability to learn, generalize, classify, and organize data (Gomes & Awruch, 2004; Sharaf, Noureldin, Osman, & El-Sheimy, 2005). ANNs can also be used to analyze complex data structures and large data sets, and these so-called intelligence systems are capable of generalizing the findings of scientific studies (Santos, Rupp, Bonzi, & Fileti, 2013). Inspired by the thinking patterns of the human brain, ANNs can "learn" the data structures and conduct numerous statistical processes such as parameter estimations, classifications, and optimizations. In other words, learning in ANNs is accomplished through algorithms that mimic the learning mechanisms of biological systems (Yılmaz & Özer, 2009). Therefore, the present study investigates the factors that affect the success of university students by employing two artificial neural network methods (i.e., multilayer perceptron and radial basis function), and compares the effects of these methods on educational data in terms of predictive ability.

Multilayer Perceptron Artificial Neural Network

The multilayer perceptron artificial neural network (MLPANN) and the radial basis function artificial neural network (RBFANN) are both widely used as supervised training methods. Although their structures are somewhat similar, the RBFANN is used to solve scientific problems, whereas the MLPANN is applied for pattern recognition or classification problems by using the error back propagation algorithm. The main purpose of this algorithm is to minimize estimation error by computing all of the weights in the network. In addition, this algorithm systematically updates these weights in order to achieve the best neural network configuration. Essentially, this algorithm consists of two steps: propagation and weight update. Basically, the propagation step involves forward propagations (producing output activations) and backward ones (computing the difference between input (X_i) and output (Y_i) by using output activations). In the weight update process, synaptic weight is multiplied by the delta (X_i-Y_i) to obtain the gradient weight. Then, a percentage of the gradient weight is subtracted to obtain the rate of pattern recognition. In this case, if the percentage is low, then the accuracy of the training is high. Moreover, if the percentage is high, then the training of the neurons is faster. During this process, the two steps (propagation and weight update) are repeated until the performance of the network architecture is satisfactory.

Back propagation needs to compute the derivative of the squared error function by considering the weights in the network. Assuming one output neuron, the squared error function can be computed as follows:

$$\epsilon = \frac{1}{2}(t-y)^2 \quad (1)$$

where t is the target output, y is the actual output, and E is the squared error. Each output (O_j) that is matched with each neuron can be expressed as follows:

$$O_j = \varphi(net_j) = \varphi\left(\sum_{k=1}^{n} w_{kj} x_k\right) \quad (2)$$

The net_j (input) to a neuron is the weighted sum of output O_k. In addition, w_{kj} is the weight between neurons $_k$ and j. In general, the activation function of the hidden and output layers is non-linear and differentiable. A common activation function is shown as follows:

$$\varphi(z) = \frac{1}{1+e^{-z}} \quad (3)$$

The activation function in Equation 3 is a logistic form that is commonly used as the activation function. This process continues until the error is optimized. In other words, the back propagation algorithm attempts to find the derivative of the error.

In feed-forward neural networks such as the MLPANN, the input layer includes a linear activation function, but sigmoid tangent, logarithmic, and hyperbolic activation functions (which are non-linear functions) are used in the hidden and output layers. A typical structure of a MLPANN is shown in Figure 1.

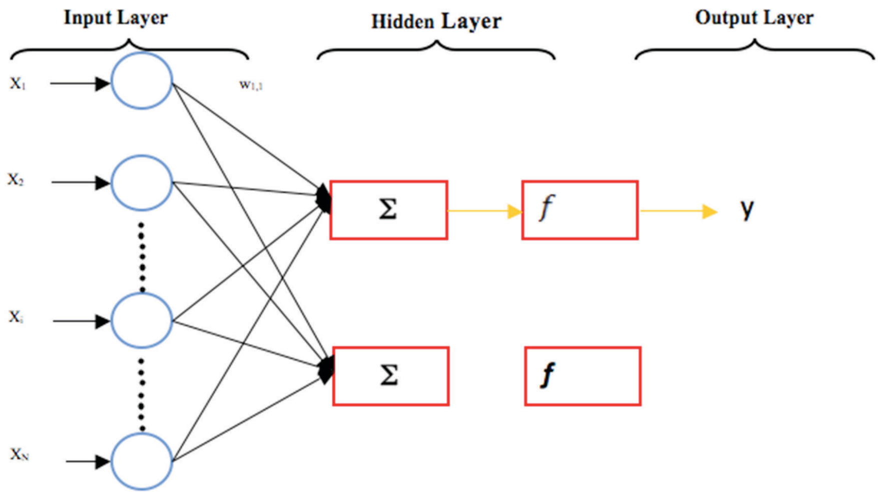

Figure 1: The basic structure of a multilayer perceptron artificial neural network.

In Figure 1, each input variable is calculated with the weight of *w*. The target variable (output), which is represented as *y*, can be expressed as follows:

$$y = f(w_i + b), \tag{4}$$

where *b* is the bias associated with the neurons. In a MLPANN, hidden layers often include sigmoid neurons or hyperbolic functions that reveal the non-linear relationships between inputs and outputs. The training process and the computation of the neurons can be calculated as follows (Yılmaz & Özer, 2009):

$$y_p^k = sgm_p^k\left[\sum_{i=1}^{N_{k-1}} w_{ip}^{k-1} \cdot y_i^{k-1} - \beta_i^k\right]$$
$$p = 1,2,\dots,N_k; k = 1,2,\dots,M, \tag{5}$$

where w_{ip}^{k-1} is the connection weight between the *i*th neuron in the $(k-1)$th layer and the *p*th neuron in the *k*th layer, y_p^k is the output of the *p*th neuron in the *k*th layer, is the sigmoid activation function of the *p*th neuron in the *k*th layer, sgm_p^k and β_i^k is the threshold of the *p*th neuron in the *k*th layer (Kasiri, Momeni, & Kasiri, 2012). However, the sigmoid activation function is given as:

$$sgm(x) = {}^{1}/_{1 + exp(-x)} \tag{6}$$

Radial Basis Function Artificial Neural Network

The radial basis function artificial neural network (RBFANN) contains one significant advantage in which it can reveal non-linear relationships between independent and target parameters (Mustafa, Rezaur, Rahardjo, & Isa, 2012). Basically, the RBFANN was offered as an alternative to the MLPANN for analyzing complex models (Luo & Unbehauen, 1999) since it was shown that the RBFANN can be implemented with increased input dimensions (Wilamowski & Jaeger, 1996). The RBFANN includes two additional advantages: its training process is faster than the conventional back propagation neural network; and it more robust to the complex problems associated with active (non-stationary) inputs (Chen, Zhao, Liang, & Mei, 2014).

Similar to the structure of the other architectures, the RBFANN also includes three layers: input, hidden, and output. The output layer includes a linear form while the hidden layer is supported with a non-linear RBF activation function. The structure of the RBFANN is shown in Figure 2.

The input variables are the combination of the input vector x = $[x_1, x_2, \dots\dots x_n]$. These vectors are matched with the radial basis functions in each hidden node. In addition, vector *y*, which is a linear combination of the final output, is yielded (Chen et al., 2014) after which the *y* output can be obtained as follows:

$$y = \sum_{i=1}^{M} \omega_i \, \emptyset_i(x), \tag{7}$$

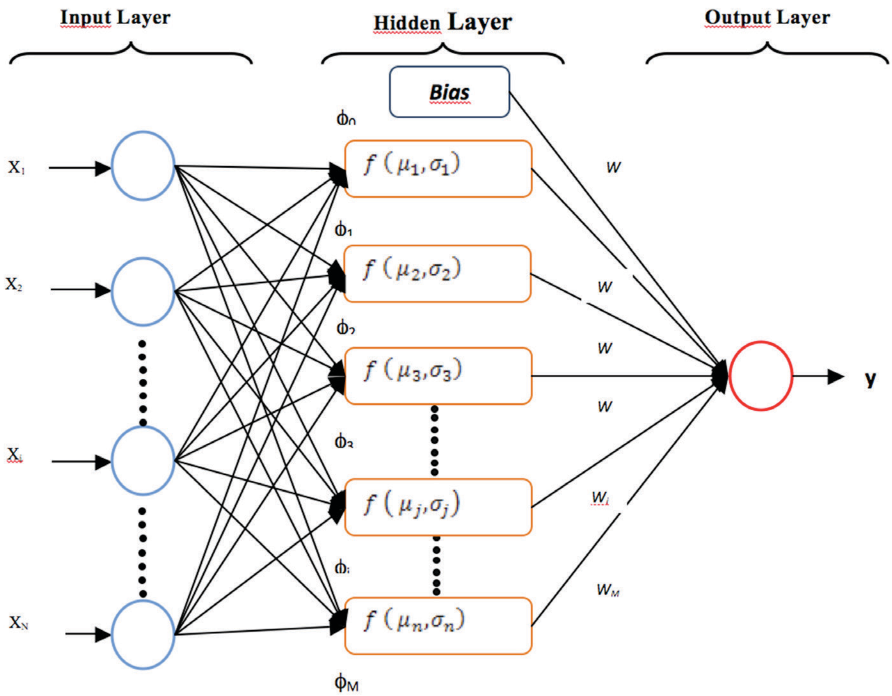

Figure 2: The basic structure of a radial basis function artificial neural network.

where w_i denotes connection weights between the hidden and output layers and w_0 is the bias. The notations x_1, x_2,......x_n represent the number of input nodes in the output layer, whereas $\emptyset_i(x)$ is the radial function in the hidden layer and *M* is the total number of nodes in this layer. Basically, a radial basis function is a multi-dimensional function that describes the distance between a given input vector and a pre-defined center vector (Chen et al., 2014).

In the related literature, various radial basis functions have been described and used depending on the data structure. The normalized Gaussian model is most commonly used (also in the present study) as the radial basis function (Mustafa et al., 2012; Narendra, Sivapullaiah, Suresh, & Omkar, 2006), which can be expressed as follows:

$$\emptyset_i(x) = exp\left(-\frac{\|x-\mu_i\|^2}{2\sigma_i^2}\right), \tag{8}$$

where μ_i and σ_i represent the center and spread width parameter of the basic function $\emptyset_i$. In addition, $||,||$ is the norm of Euclidean distance and using the Gaussian function provides some advantages such as radial symmetry and improved smoothness.

In regard to using the Gaussian function, the output of the radial basis network yields the following equation:

$$y = \sum_{i=1}^{M} \omega_i exp\left(-\|x-\mu_i\|^2/2\sigma_i^2\right) \tag{9}$$

Basically, the training process of the RBFANN consists of three steps: 1) calculate the width σ_i; 2) adjust the center μ_i; and 3) adjust the weight w_i. The two-step clustering algorithm is used to find the RBF center and width. In regard to fixing the width according to the spread of the centers, the radial basis function is:

$$\emptyset_i = e^{\left(-\frac{h}{d^2}\|x-\mu_i\|^2\right)},\ i = 1, 2, \ldots h, \tag{10}$$

where *h* is the number of centers and d is the maximum distance between the chosen centers. As a result

$$\sigma = \frac{d}{\sqrt{2h}} \tag{11}$$

Furthermore, the base function depends on the smaller value of *d* to obtain a smaller width in the RBFANN (Chen et al., 2014).

Performance Criteria to Determine the Best Artificial Neural Network

In general, the purpose of indicators in an artificial neural network, such as the error function, is to examine the performance of its architecture. In RBF and MLP networks, the sum-of-squares error (SSE) is used as the error function, which can be expressed as follows:

$$\varepsilon = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{M}\left[y_j(x_i) - t_{ij}\right]^2, \quad (12)$$

where $y_j(x_i)$ is the network output and t_{ij} is the target. In other words, training in RBF and MLP networks is achieved by minimizing the error function e. In addition, the error function is the squared difference between the predicted and the real (observed) data.

Not only is the SSE used as a network performance measurement but also the mean squared error (MSE), the root-mean-square error (RMSE), the mean absolute error (MAE), the coefficient of efficiency (CE), and the coefficient of correlation measurements are also used to determine the optimal architecture of a neural network. These measurements are defined as follows:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(P_i - O_i)^2 \quad (13)$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(P_i - O_i)^2} \quad (14)$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|P_i - O_i| \quad (15)$$

$$CE = 1 - \frac{\sum_{i=1}^{N}(P_i - O_i)^2}{\sum_{i=1}^{N}(P_i - P_M)^2} \quad (16)$$

where, P_i and Q_i are the predicted and the real data of the output variable, respectively. Moreover, P_M is the mean of predicted output, whereas N is the number of real data. The error can be calculated by using these values. Regarded as ideal, the values of MSE, RMSE, and MAE should be obtained as zero, whereas the CE should be computed as one (Mustafa et al., 2012). However, in practice, it is impossible to obtain the values of zero and one. The optimal architecture of a neural network depends on how much the MSE, the RMSE, and the MAE performance measurements are closer to zero or one for the CE.

Purpose

The purpose of this study is twofold: *1*) to investigate the factors that affect the success of university students by employing two artificial neural network methods (i.e., multilayer perceptron and radial basis function); and *2*) to compare the effects of these methods on educational data in terms of predictive ability. To date, such a comparison has not been previously conducted, which makes the findings of the present study even more appealing. The overall goal of this study is to foster researchers to employ these advanced methods in future quantitative educational studies.

Methods

Material (Data Set)

In this study, the data was based on the transcript scores of 1,271 university students (858 female [67.5%] and 413 male [32.5%]). The mean age of the students was 20.89 (with a standard deviation of 2.05 years) and the mean transcript scores, which was output (*y*) in this study, was 2.69 (with a standard deviation of 0.43). The overall range of

Table 1
The Descriptive Statistics of the Predictors

Inputs	Categories	Frequency	%
Satisfaction with the department	1- Yes 2- No	999 272	78.6 21.4
Self-assessment of course attendance	1. I attend courses as much as possible 2. I use my periodic absence rights until the end of the course 3. I am never absent from courses that I like	894 195 182	70.3 15.3 14.4
Preferred study times	1- Daytime 2- Nighttime 3- It does not matter	225 569 477	17.7 44.8 37.5
Efficient use of time	1- Yes 2- No	439 832	34.5 65.5
Planning before any tasks	1- Yes 2- No	1004 267	79 21
The contributions of friends on success	1- Positive contributions 2- Negative contributions 3- No effect	582 150 539	45.8 11.8 42.4

the transcript scores was between 0 and 4. There were nine variables of which eight were the inputs of the architecture and one was the output of the model. The data was collected from prepared questionnaires in which some of the input variables were nominal (dichotomous or multinomial) and some were ordinal or continuous. The following were used as inputs of the neural network: gender; age; satisfaction with the department; self-assessment of course attendance; preferred study times; efficient use of time; planning before any tasks; and the contributions of friends on success. It was found that the inputs did have an effect on the students' scores after using the MLPANN and the RBFANN. The descriptive statistics of the inputs are presented in Table1.

As shown in Table 1, 78% of the students were satisfied with their departments and 70.3% were willing to attend courses as much possible. Regarding study times, 44.8% of the students preferred studying during the day while 37.5% did not have a preference. The majority (65.5%) felt that they did not use their time efficiently, but 79% did have a plan or schedule before beginning their tasks. Finally, 45.8% of the students believed that their friends positively contributed to their overall success. Thus, along with the other variables, this variable was added as a predictor in the neural network models.

Data Analysis

The feed-forward neural networks (the MLPANN and the RBFANN used herein) and the input vector of independent variable x_i was related to the target variable (y_i, transcript score), based on the framework in Figures 1 and 2. The architecture of the network was such that $\mathbf{P}_\mathbf{i}^\mathbf{'} = (p_{i1}, p_{i2} \dots, p_{i8})$ contained values for eight input (independent) variables from individual *i*. Following Mackay (2008), for two layers (the hidden and output layers) of supervised learning in a feed-forward network, the mapping includes the following formulas for the relationship between output and the independent variables:

$$\textit{Hiddenlayer}\ n_k^{(1)} = \Sigma_{j=1}^{R} w_{kj}^{(1)} p_j + b_k^{(1)}; \qquad a_k^1 = f_{level-one}\left(n_k^{(1)}\right)$$

$$\textit{Outputlayer}\ n_k^{(2)} = \Sigma_{j=1} w_k^{(2,1)} a_k^1 + b_1^{(2)}; \quad \hat{t}_i = a_k^{(2)} = f_{level-two}\left(n_k^{(2)}\right)$$

In the activation function, the biases were computed and then the activation function was reapplied to the data in order to move the transformed function to the output layer. In other words, the transformed activation function yields the estimated target variable (the transcript score) (Okut, Gianola, Rosa, & Weigel, 2011):

$$\hat{t}_i = g\left\{\Sigma_{k=1}^{N} w_k^{'} f\left(\Sigma_{j=1}^{R} w_{kj} p_j + b_k^{(1)}\right) + b^{(2)}\right\}; \quad j = 1,2\dots,R \ \ k = 1,2\dots,N \qquad (17)$$

In this study, the combination activation function (*f*) was used as follows:

1) $f_{hiddenlayer}(.) = linear(.)$ *and* $f_{outputlayer}(.) = linear(.)$
2) $f_{hiddenlayer}(.) = hyperbolictangent(.)$ *and* $f_{outputlayer}(.) = linear(.)$

Furthermore, the MSE, the RMSE, the MAE, and the CE were used to compare the predictive ability of the MLPANN and the RBFANN. Before testing the neural network architecture, a multi-collinearity test was conducted, which examined high intercorrelations or interassociations among the input variables. In addition, the variance inflation factor (VIF) was used to determine multi-collinearity. If the value of the VIF is greater than 10 or the tolerance value is less than 0.1, then there is a serious multi-collinearity problem among the predictors (Keller, El-Sheikh, Granger, & Buckhalt, 2012). In the present study, the VIF values were between 1.026 and 1.059 and the tolerance values were between 0.944 and 0.990. These indicators show that there was no multi-collinearity problem among the predictors.

In this study, 70% of the data was used for training and the remainder of the data (30%) was used for testing. Before analyzing the algorithms of the neural network, Gaussian normalization was performed using the data set. Both the MLPANN and the RBFANN were tested in the following order:

i) Maximum steps without a decrease in error is 1;
ii) Maximum training time is 15 minutes;
iii) Maximum training epochs is automatically computed;
iv) Minimum relative change in training error is 0.0001 and minimum relative change in training error ratio is 0.001.

Results

First, the MLPANN was applied to the data set. The input layer consists of eight predictors, the number of hidden layers is one, and the optimal number of units in the hidden layer (bias) is 10. The activation function of the hidden layer is hyperbolic tangent and the activation function of the output layer is identity. The identity function takes real-valued arguments and returns them unchanged. The error

Table 2
The Performance of the MLPANN and the RBFANN

Neural Network Architecture	Relative Error	SSE	Correlation	MSE	RMSE	MAE	CE
MLP	0.839	145.534	0.421**	0.173	0.416	0.310	−2.568
RBF	0.884	191.845	0.349**	0.164	0.406	0.312	−6.654

**Correlation between the observed and predicted data is significant at the 0.01 level.

function of the output layer is the sum-of-squares error (SSE) and error computations were based on the testing sample. The training time for the MLPANN was 94 seconds.

In regard to applying the RBFANN to the data set, similar to the MLPANN, the number of hidden layers is one and the optimal number of units in the hidden layer (bias) is 10. The activation function of the hidden layer is softmax and the activation function of the output layer is identity. The softmax activation function in the hidden layer takes the vector of real-valued arguments and transforms it into a vector whose elements fall in the range (0, 1) and sum to 1 (Singh, Mittal, & Kahlon, 2013). The training time for the RBFANN was 2 minutes and 21 seconds. The results show that the process of the MLPANN was obviously better than the RBFANN in terms of training time. The model, obtained with the MLPANN and the RBFANN, is summarized in Table 2.

Table 2 shows that the correlation between the observed and predicted data in the MLPANN is higher than that of the RBFANN. The other performance indicators (MSE, RMSE, and MAE) should be close to zero and the CE should be close to one. According to the SSE criteria, the MLPANN obtained better results than the RBFANN. The values of MSE, RMSE, and MAE are acceptable since they are close to zero and there is no meaningful difference between the MLPANN and the RBFANN. It is known that the value of the CE should be one (as ideal) and theoretically, the value of CE should be between $-\infty$ and 1. In this case, compared to the RBFANN, the CE value of the MLPANN is closer to 1. In other words, the predicted value obtained by the MLPANN is more reliable than that of the RBFANN. Thus, the performance of the MLPANN is more robust than the RBFANN in terms of correlation and CE. The findings of the performance criteria are shown in Figure 3.

According to the performance criteria, the architecture of the MLPANN should be taken into consideration. The reason being that the predictors that affect the target variables in the MLPANN architecture become less unbiased and more robust. The importance of the independent variables in the MLPANN architecture is shown in Table 3.

Table 3
Independent Variable Importance in the MLPANN Architecture

Predictors	Importance	Normalized Importance (%)
Age	0.218	100
Gender	0.200	92
Preferred study times	0.154	70.6
The contributions of friends on success	0.139	64
Efficient use of time	0.098	45.1
Satisfaction with the department	0.087	39.7
Self-assessment of course attendance	0.064	29.5
Planning before any tasks	0.040	18.2

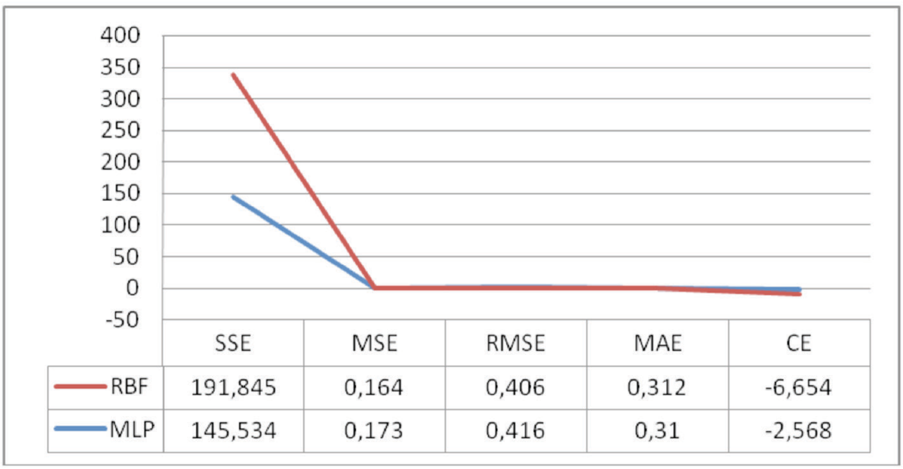

Figure 3: The Performance of the MLPANN and the RBFANN.

Table 3 shows that the age predictor was the most effective variable with 100% normalized importance. The gender predictor was the second most effective variable with 92% normalized importance, while the predictor of preferred study times was the third highest at 70.6%. The contributions of friends on success variable were found to be moderately significant with 64% normalized importance. The predictors of efficient use of time, satisfaction with the department, and self-assessment of course attendance had lower significance with 45.1%, 39.7%, and 29.5% normalized importance, respectively. Finally, the least effective predictor was the planning before any tasks variable with 18.2% normalized importance.

Overall, the results of the predictors in the MLPANN were more reliable than those of the RBFANN. In addition, the effects of the predictors between the two models differed, and the results of the RBFANN were more biased and less robust than those of the MLPANN. The importance of the independent variables in the RBFANN is shown in Table 4.

Table 4
Independent Variable Importance in the RBFANN Architecture

Predictors	Importance	Normalized Importance (%)
Gender	0.270	100
Satisfaction with the department	0.207	76.5
Planning before any tasks	0.114	42.2
Self-assessment of course attendance	0.111	41.0
Preferred study times	0.103	38.2
Efficient use of time	0.083	30.8
The contributions of friends on success	0.082	30.4
Age	0.029	10.8

It was found that the age predictor was the least effective variable in the RBFANN, whereas it was the most effective variable in the MLPANN. In the RBFANN, the most effective variable was the gender predictor with 100% normalized importance. The second most effective predictor was the satisfaction with the department variable with 76% normalized importance. Although the third most effective variable was planning before any tasks, it was the least effective variable in the MLPANN. Finally, the self-assessment of course attendance, the efficient use of time, and the contributions of the friends predictors were computed with 41%, 38.2%, 30.8%, and 30.4% normalized importance, respectively. The results show that the MLPANN was more reliable on educational data due to non-linear relationships.

Conclusion and Discussion

The overall purpose of this study was to demonstrate the predictive abilities of the MLPANN and the RBFANN on educational data. The transcript scores of the sample of university students were used with two different feed-forward-based algorithms. Previous studies have shown that the predictive ability of the RBFANN was more effective than that of the MLPANN. Yılmaz and Özer (2009) proposed an artificial neural network-based pitch angle controller for wind turbines and found that the RBFANN outperformed the MLPANN. Bonanno, Capizzi, Graditi, Napoli, and Tina (2012) studied the electrical characteristics estimation of a photovoltaic module by using the RBFANN and the MLPANN comparatively. Their results showed that the RBFANN-based models achieved superior performance compared to the MLPANN. However, other studies have highlighted the advantages of using the RBFANN (Pontes, Paiva, Balestrassi, & Ferreira, 2012; Sideratos & Hatziargyriou, 2012; Wu & Liu, 2012; Yu, Xie, Paszczynski, & Wilamowski, 2011; Zhou, Ma, Li, & Li, 2012).

Although these aforementioned studies suggested that the RBFANN should be utilized in engineering research, some studies have shown that the performance of the MLPANN was better in engineering science. For example, in their study related to chemical engineering, Santos et al. (2013) tested the performance of the MLPANN and the RBFANN comparatively. According to their findings, the MLPANN outperformed the RBFANN. Nevertheless, the RBFANN is still suggested for engineering studies.

The present paper has shown that the MLPANN outperformed the RBFANN in terms of predictive ability. In addition, if the data is gathered from individuals via questionnaires or other instruments, then the predictive ability of the MLPANN is more robust and less biased than the RBFANN due to non-linear relationships. Therefore, it recommended that studies in educational science should be carried out using the MLPANN (instead of the RBFANN) since higher correlations and fewer errors can occur. Furthermore, independent variables are more reliable and more robust in the MLPANN architecture, and its training time is generally shorter than the RBFANN architecture. Finally, although the findings suggest that research in quantitative educational science should be conducted by using the MLPANN, additional supporting evidence needs to be collected in related studies.

References

Bonanno, F., Capizzi, G., Graditi, G., Napoli, C., T., & Tina, G. M. (2012). A radial basis function neural network based approach for the electrical characteristics estimation of a photovoltaic module. *Applied Energy*, *97*, 956–961.

Chen, J., Zhao, P., Liang, H., & Mei, T. (2014). Motion planning for autonomous vehicle based on radial basis function neural network in unstructured environment. *Sensors*, *14*, 17548–17566.

Gomes, H. M., & Awruch, A. M. (2004). Comparison of response surface and neural network with other methods for structural reliability analysis. *Structural Safety*, *26,* 49–67.

Gonzalez-Camacho, J. M., Campos, G., Pe´rez, P., Gianola, D., Caims, J. E., Mahuku, G. R. B., & Crossa, J. (2012). Genome-enabled prediction of genetic values using radial basis function neural networks. *Theoretical and Applied Genetics, 125*, 759–771.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning*. New York, NY: Springer.

Kasiri, H., Momeni, H. R., & Kasiri, A. (2012). Optimal intelligent control for wind turbulence rejection in WECS using ANNs and genetic fuzzy approach. *International Journal of Soft Computing and Software Engineering (JSCSE), 2*(9), 16–34.

Keller, P. S., El-Sheikh, M., Granger, D. A., & Buckhalt, J. A. (2012). Interactions between salivary cortisol and alpha-amylase as predictors of children's cognitive functioning and academic performance. *Physiology & Behavior, 105*, 987–995.

Luo, F. L., & Unbehauen, R. (1999). *Applied neural networks for signal processing*. UK: Cambridge University Press.

Mackay, J. C. D. (2008). *Information theory, inference and learning algorithms*. UK: Cambridge University Press.

Mustafa, M. R., Rezaur, R. B., Rahardjo, H., & Isa, M. H. (2012). Prediction of pore-water pressure using radial basis function neural network. *Engineering Geology, 135*(136), 40–47.

Narendra B. S., Sivapullaiah, P. V., Suresh, S. N., & Omkar, S. N. (2006). Prediction of unconfined compressive strength of soft grounds using computational intelligence techniques: A comparative study. *Computers and Geotechnics, 33*, 196–208.

Okut, H., Gianola, D., Rosa, G. J. M., & Weigel, K. A. (2011). Prediction of body mass index in mice using dense molecular markers and a regularized neural network. *Genetics Research, 93,* 189–201.

Pontes, F. J., Paiva, A. P., Balestrassi, P. P., & Ferreira, J. B. (2012). Optimization of radial basis function neural network employed for prediction of surface roughness in hard turning process using Taguchi's orthogonal arrays. *Expert Systems with Applications, 39,* 7776–7787.

Sharaf, R., Noureldin, A., Osman, A., & El-Sheimy, N. (2005). Online INS/GPS integration with a radial basis function neural network. *IEEE A&E Systems Magazine, 3,* 8–14.

Santos, R. B., Rupp, M., Bonzi, S. J., & Fileti, A. M. F. (2013). Comparison between multilayer feedforward neural networks and a radial basis function network to detect and locate leaks in pipelines transporting gas. *Chemical Engineering Transactions*, *32,* 1375–1380.

Sideratos, G., & Hatziargyriou, N. D. (2012). Probabilistic wind power forecasting using radial basis function neural networks. *IEEE Transactions on Power Systems, 27*(4), 1788–1796.

Singh, S., Mittal, P., & Kahlon, K. S. (2013). Empirical model for predicting high, medium and low severity faults using object oriented metrics in Mozilla Firefox. *International Journal of Computer Applications in Technology (IJCAT), 47*(2/3), 110–124.

Wilamowski, B. M., & Jaeger, R. C. (1996). Implementation of RBF type networks by MLP networks. *Proceedings., IEEE International Conference on · Neural Networks* (pp. 1670–1675). Washington, DC: IEEE.

Wu, J. D., & Liu, J. C. (2012). A forecasting system for car fuel consumption using a radial basis function neural network. *Expert Systems with Applications, 39*, 1883–1888.

Yılmaz, A. S., & Özer, Z. (2009). Pitch angle control in wind turbines above the rated wind speed by multi-layer perceptron and radial basis function neural networks. *Expert Systems with Applications*, *36*, 9767–9775.

Yu, H., Xie, T., Paszczynski, S., & Wilamowski, B.M. (2011). Advantages of radial basis function networks for dynamic system design. *IEEE Transactions on Industrial Electronics, 58*(12), 5438–5450.

Zhou, R., Ma, X., Li, S., & Li, J. (2012). The green supplier selection method for chemical industry with analytic network process and radial basis function neural network. *Advances in information Sciences and Service Sciences (AISS), 4*(4), 147–158.