2011

Neural network application for fuzzy multi-criteria decision making problems

Davood Golmohammadi, PhD, *University of Massachusetts Boston*

Int. J. Production Economics 131 (2011) 490–504

Contents lists available at ScienceDirect

Int. J. Production Economics

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

Neural network application for fuzzy multi-criteria decision making problems

Davood Golmohammadi *

Department of Management Science and Information Systems, University of Massachusetts, Boston, USA

ARTICLE INFO

Article history:
Received 23 June 2010
Accepted 22 January 2011
Available online 1 February 2011

Keywords:
Neural networks
Decision making
Fuzzy sets

ABSTRACT

In this paper, a fuzzy multi-criteria decision making model is presented based on a feed forward artificial neural network. This model is used to capture and represent the decision makers' preferences. The topology of the neural network model is developed to train the model. The proposed model can use historical data and update the database information for alternatives over time for future decisions. Basically, multi-criteria decision making problems are formulated, and neural network is used to learn the relation among criteria and alternatives and rank the alternatives. We do not use any utility function for the modeling; however, a unique method is proposed for eliciting the information from decision makers. The proposed model is applicable for a wide variety of multi-attribute decision making problems and can be used for future ranking or selection without managers' judgment effort. Simulation of the managers' decisions is demonstrated in detail and the design and implementation of the model are illustrated by a case study.

1. Introduction

Decision making involves choosing some course of action among various alternatives. For a decision maker, fulfilling the conflicting goals while satisfying the constraints of the system is the main concern.

The ability to make the best possible decision (output) based on past and present information and future predictions (input) is a formidable task. A tool that can facilitate this task is of great help for decision makers (Malakooti and Zhou, 1994).

Increasingly, researchers are exploring artificial neural networks as a method for decision support systems. Neural network is a powerful data modeling tool that is able to capture and represent complex input/output relationships.

The most important applications of neural networks can be categorized in four groups: Classification such as medical diagnosis and target recognition, function approximation such as process modeling, time series prediction, and data mining (Golmohammadi et al., 2009).

The objective of this paper is primarily to design a decision model based on neural networks, rather than to focus on neural networks design challenges in depth. The proposed model is explained in general and is illustrated by a case on supplier evaluation and selection as one of the neural network applications. The main contributions of this research are:

- A decision model based on the relationship of a set of input and output is designed in a unique manner to predict the score of objects (suppliers or any objects).
- An innovative pairwise comparison technique for weight calculations and output values is applied for an NN model.
- Performance history of objects is considered in the evaluation process and modeling.

The model can use the input and evaluate objects and rank them as output. The proposed model is used for future ranking without decision makers' judgment effort. However, the traditional decision making methods follow the same process to make the decision. The difference between this model and other traditional models is shown in Fig. 1. We will discuss more about the neural networks models for multi-criteria decisions in the literature section.

This paper is organized in several sections. After the literature review and brief introduction of neural network structure, model design is illustrated in Section 4. In Section 5, the model design is implemented with more details using a case study. In Section 6, the design is improved for better results. In Section 7 a sensitivity analysis is performed for the case study model. The last section provides one more example of the model application and draw conclusion.

* Tel.: +1 617 287 7888.
E-mail address: davood.golmohammadi@umb.edu

0925-5273/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.ijpe.2011.01.015

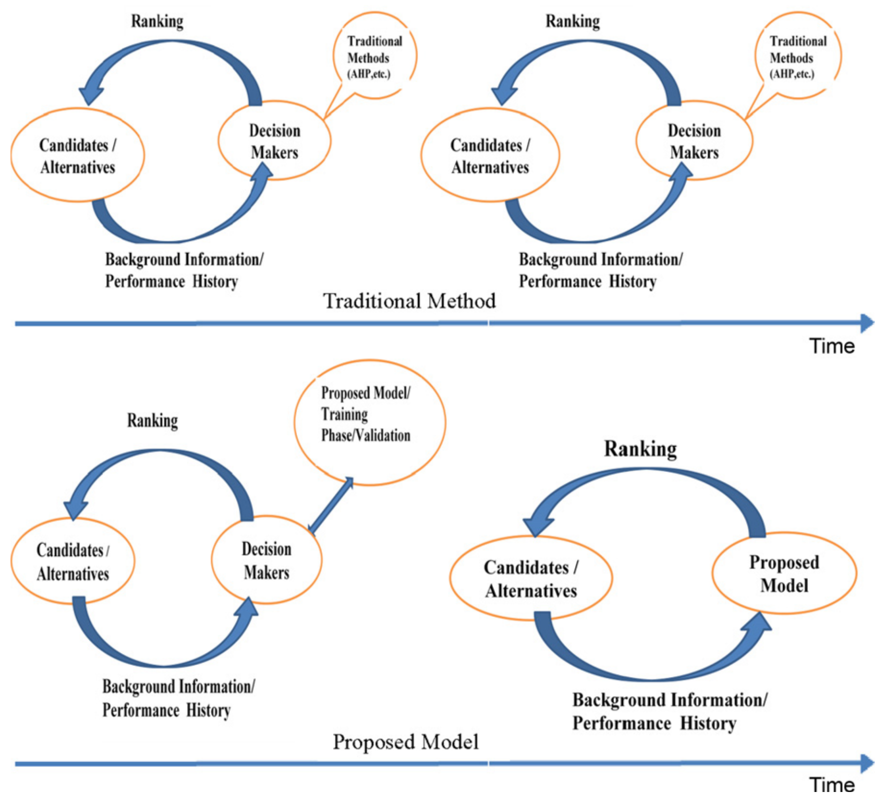

Fig. 1. The conceptual difference between the proposed model and traditional methods.

2. Literature review

Many models have been presented for decision making in the literature. They are often intuitively judged, applied in a complex manner, lack qualitative factors to show the imprecision of the performance data, and do not corporate to historical performance.

These methods are mostly based on multi-objective optimization problems (MOP) (Weber et al., 1998, 2000; Dahel, 2003; Chen et al., 2006; Awasthi et al., 2010), data envelopment analysis (DEA) (Liu et al., 2000; Weber, 1996), analytic hierarchical process (AHP) (Ghodsypour and O'Brien, 1998; Bhutta and Huq, 2002), simple multi-attribute rating technique (SMART) (Seydel, 2005), total cost approaches and multiple attribute utility theory (Ellram, 1990).

These models provide systematic approaches for decision makers to evaluate and score objects with multiple criteria. Nevertheless, these models are not easy to implement. Models based on multi-objective optimization require the decision makers to exogenously specify the exact values of weights of individual criteria. It is, however, difficult to obtain precise weight values. The weight determination is a challenging task for implementing the MOP approach. Decision makers choosing the SMART approach face a similar problem. DEA appears to be the easiest for practical implementation. Where DEA models are applied to supplier selection problems, decision makers cannot have any involvement or control of the importance of the criteria. To some extent, these DEA approaches are black box models for decision makers in real situations (Singh et al., 2007).

The total cost of ownership is a methodology and philosophy which looks beyond the price of a purchase to include many other purchase related costs. This approach has become increasingly important as organizations look for ways to better understand and manage their costs. However, the total cost model is expensive to implement due to its complexity, requires more time, and implies the ability to identify the more important elements (Ellram, 1995).

The multiple attribute utility theory (MAUT) method is appropriate in situations where a variety of uncontrollable and unpredictable factors affect the decision. It is capable of handling multiple conflicting attributes inherent in international supplier selection (Bard, 1992).

Another category of decision-making models is intelligent methods. Neural networks as one of these intelligent methods attempt to simulate the human brain by collecting and processing data for the purpose of "remembering" or "learning". Some researchers have developed neural networks models for multi-criteria decision making (Malakooti and Zhou, 1994; Sun et al., 1996; Chen and Lin, 2004). Sun et al. (2007) proposed a new interactive multiple objective programming procedure that combines the strengths of the interactive weighted Tchebyche procedure and the interactive feed forward artificial neural network procedure. Chen and Lin (2004) proposed a new approach for solving multiple criteria decision-making (MCDM) problems based on decision neural network (DNN). The DNN is used to capture and represent the decision maker's (DM's) preference. Then, with DNN, an optimization problem is solved to search for the most desirable solution. Shih et al. (2004) focused on utilizing the dynamic behavior of artificial neural networks (ANNs) to solve multi-objective programming (MOP) and multilevel programming (MLP) problems. Singh et al. (2007) proposed multi-criterion frameworks involving several subjective and quantitative factors that allow the complexity of group decision making (GDM) to get worse, especially for those problems which have strategic dimensions. Their improved decision neural network (IDNN) based methodology has been developed to solve the multi-criterion decision problem in GDM. Reductions in the training data set, exploitation of indirect methods like multiplicative preference relation during the training process, and reduced number of iterations to map the MAUF are the advantages of this novel methodology. Tan et al. (2006) proposed a hybrid intelligent system integrating case-based reasoning (CBR) and the fuzzy ARTMAP (FAM) neural network model to support managers in making timely and optimal manufacturing technology investment decisions. Wu (2009) presented a hybrid model using data envelopment analysis (DEA), decision trees (DT) and neural networks (NNs) to assess supplier performance. The model consists of two modules: Module 1 applies DEA and classifies suppliers into efficient and inefficient clusters based on the resulting efficiency scores. Module 2 utilizes firm performance-related data to train DT, NNs model and apply the trained decision tree model to new suppliers.

All these models used Neural Networks; however, the models are either deeply involved in developing neural network structures (especially weight computations among nodes) or cumbersome computations are needed to evaluate alternatives. The design of our proposed model can map and simulate the decision makers' preference for training and future predication easily and efficiently. Using the history data for future evaluation and ranking is another advantage of this model. The input and output design and application of pairwise comparisons are unique. Especially, regular procedure of pairwise comparison is improved to enhance the accuracy of training data.

3. Neural networks structure

A set of valid and adequate data combined with a properly designed neural network can lead to the correct decision. The general structure of the artificial neural networks is shown in Fig. 2. In this structure there are three parallel layers. The first layer (input) contains the independent variables, the second layer(s) are hidden layers containing processing units called hidden nodes, and the third (output) layer contains the dependent variables. The layers are connected by weighted links. These weights are estimated through a training procedure. Fig. 3 shows input and output of one of the hidden nodes.

The network performs a forward pass with the production of an error signal for each output neuron. The error signals are then transmitted backward from the output layer to the neurons in the intermediate layer that contribute directly to the output (Chen and Lin, 2004; Sun et al., 1996; Malakooti and Zhou, 1994). This is part of the training step which will be discussed later.

Selecting the best neural network architecture is crucial to the success of NN modeling (Hill et al., 1996). Several design factors,

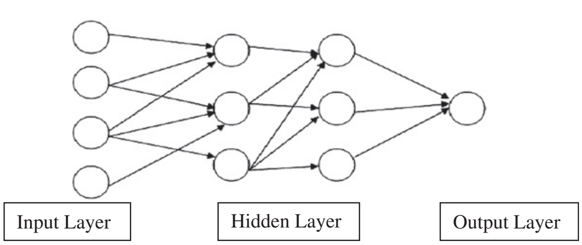

Fig. 2. The structure of the artificial neural networks.

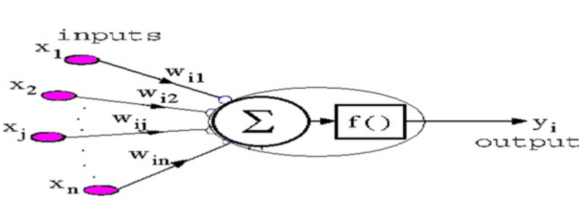

Fig. 3. The mapping of a hidden node.

including selection of input variables, architecture of the network and quantity of training data significantly impact the accuracy of neural networks forecast. The process involves the daunting task of constructing a large number of NN topologies with different structures and parameter values before arriving at an acceptable model (Denton and Hung, 1996; Dorsey and Mayer, 1994). The main effective factors in neural networks modeling are the number of input nodes, the number of hidden layers, the number of hidden nodes, weight initialization, transfer function, learning rule, learning rate, and stopping training.

The number of input nodes may affect the neural network performance. Zhang et al. (2001) studied the effect of input nodes from 1 to 5, hidden nodes, and training sample size. The result showed that the number of input nodes has significant importance when compared with hidden nodes. Hidden nodes can affect the nonlinearity of equations. Hidden layers act as layers of abstraction, pulling features from inputs. Adding hidden layers will increase both the time and the number of training exemplars necessary to train the network properly. However, too many hidden layers will cause memorizing instead of generalizing (Dow and Sietswa, 1991). One hidden layer is sufficient for most problems. Increasing the number of units in the hidden layer seems beneficial (Siying et al., 1997).

Weight initialization has an effect on the convergence time. The weights are most commonly initialized randomly. The transfer function describes how a neuron's firing rate varies with the input it receives.

Most of NNs applications report using backpropagation method. The goal of this method is to find values for all weights in the network that minimize the error through the gradient descent method.

4. Decision making models based on fuzzy data

Fuzzy data was used in this research. A brief review of fuzzy set theory is as follows:

A fuzzy set-based approach to supplier selection may represent a valid tool in supporting an organizational decision-making process. The concept of fuzzy set theory was introduced by Zadeh (1965). Fuzzy set theory is a very powerful tool that can be used to quantify imprecise data and deal with vague and incomplete data.

4.1. Fuzzy sets

A fuzzy set is a class of objects with a continuum of membership grades. A membership function, which assigns to each object a grade of membership, is associated with each fuzzy set. Usually, the membership grades range between 0 and 1. When the grade of membership for an object in a set is one, this object is absolutely in that set; when the grade of membership is 0, the object is not in that set. A brief review of some basic definitions of fuzzy set and its operations from mathematical aspects is presented. The main reference for this section is by Chen and Hwang (1991).

4.1.1. Definition of fuzzy set

Let U be a classical (or ordinary) set of objects, called the universe, whose generic elements are denoted by x. That is, $U=\{x\}$. A fuzzy set A in U is characterized by a membership function $\mu_A(X)$ which associates with each element in U a real number in the interval (0–1). The fuzzy set, A, is usually denoted by the set of pairs (Chen and Hwang, 1991).

$$A=\{(x,\mu_A(X)),x\in U\} \tag{1}$$

For an ordinary set, A

$$\mu_A(X)=\begin{cases}1, & x\in A\\ 0, & x\notin A\end{cases} \tag{2}$$

When U is a finite set $\{x_1,\ldots,x_n\}$, the fuzzy set on U may also be represented as

$$A=\sum_{i=1}^{n} x_i/\mu_A(x_i) \tag{3}$$

When U is an infinite set, the fuzzy set maybe represented as

$$A=\int (x/\mu_A(x))\,dx \tag{4}$$

4.1.2. Basic concepts of fuzzy set

The complement, support, α-cut, convexity, normality and cardinality of a fuzzy set are presented in the following sections.

- *Complement of a fuzzy set*: The definition of the complement of fuzzy set A is defined as

$$\mu_{\bar{A}}(X)=1-\mu_A(X),\quad x\in U \tag{5}$$

- *Support of a fuzzy set*: Those elements which have nonzero membership grades are considered as support of that fuzzy set

$$S(A)=\{x\in U\,|\,\mu_A(X)\geq 0\} \tag{6}$$

- *α-Cut of a fuzzy set*: α-Cut of a fuzzy set is an ordinary set whose elements belong to fuzzy set A, at least to the degree of α

$$A\alpha=\{x\in U\,|\,\mu_A(X)\geq \alpha\} \tag{7}$$

It is a more general case of the support of a fuzzy set. If $\alpha=0$ then $A\alpha=S(A)$

- *Convexity of a fuzzy set*: A fuzzy set is convex if

$$\mu_A(\lambda X_1+(1-\lambda)X_2)\geq \min(\mu_A(X_1),\quad \mu_A(X_2)) \tag{8}$$

X_1 and $X_2 \in U$ also $\lambda\in(0\text{–}1)$

- *Normality of a fuzzy set*: A fuzzy set A is normal only if there are one or more x' values such that $\mu_A(x')=1$

- *Cardinality of a fuzzy set*: The cardinality of fuzzy set A evaluates the proportion of elements of U having the property A. When U is finite, it is defined as

$$|A|=\sum \mu_A(x),\quad x\in U \tag{9}$$

For infinite U, the cardinality is defined as

$$|A|=\int_x \mu_A(x)\,dx \tag{10}$$

For more details, enormous materials can be found in the literature about fuzzy set theory.

5. Model design

To design a neural network model, three main phases should be considered. These phases – valid for any application – are illustrated in detail for suppliers selection. Fig. 4 shows the model design.

Multiple criteria need to be taken into account when selecting suppliers. To determine how suppliers have performed in the previous or current contracts, the decision maker must consider a set of criteria such as quality, technology, and price. In other words, the performance history, based on defined criteria, is an important input for the decision making process. The decision maker evaluates suppliers based on their input, and determines the ranks for suppliers as output. How the decision maker thinks and makes judgments about the suppliers' rank is a black box. If one can simulate this black box, then the input data could be used to estimate the suppliers' scores and rank them for future usage without the decision makers' judgment. The proposed model simulates this black box by Neural Networks.

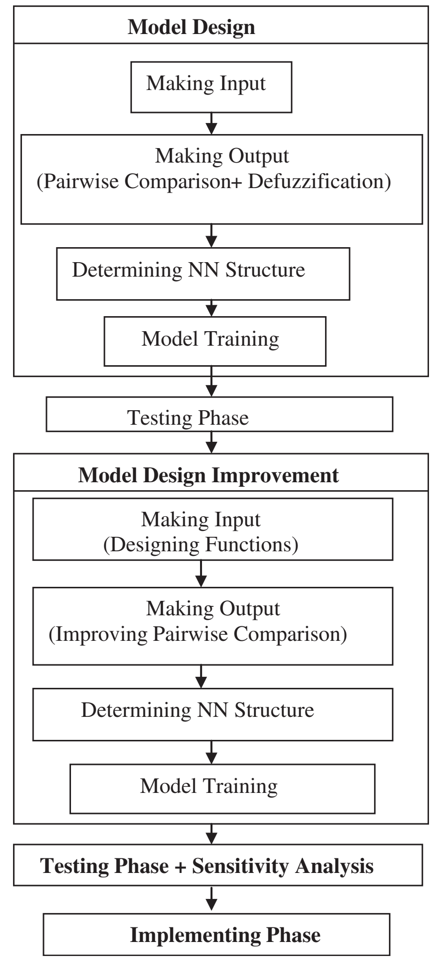

Fig. 4. Model design steps.

5.1. Design elements

Input and output layers are designed based on the application. Preparing the input and output data must be completed prior to the hidden layers development. Input is the data or information needed to make the decision. The results of the decision are the object rankings or scores as output.

5.1.1. Input

In supplier selection, the input data is prepared from suppliers' performance. Decision makers consider several criteria, such as quality, delivery, price, technology, etc., to evaluate suppliers' performance. Historical and recent suppliers' data are needed for each individual criterion. If they have a comprehensive database of the suppliers' performance, a set of functions can be defined to transform the raw data for each criterion to the input data for the neural network model. For example, quality might be measured by the ratio of the number of defective parts to the number of delivered parts in each contract for each supplier. It is emphasized that the database, which has all data history of suppliers performance such as delays in deliveries, amount of defective parts, quality issues, etc., should be complete and accurate; so that the functions can be defined appropriately.

5.1.2. Output

Output is the rank or score of suppliers based on the decision makers' judgment. In a simple case, decision makers can consider the rank intuitively, or follow their own scoring system. In this study an application of the pairwise comparison method based on fuzzy data for output estimation in neural network is proposed to make the decision.

5.1.3. Neural networks architecture

Selecting the rest of neural networks architecture is the last step to make the entire structure of the model ready for the training step. Several design factors should be taken into account, such as number of hidden nodes, hidden layers, transfer functions, learning rules, etc. There is no optimal formula for determining the appropriate architecture; however, trial and error approach for the key factors, and following the rule of thumb, can facilitate finding a good solution. Using existing off-the-shelf software such as Neurosolution, apart from application, can help the user design a proper structure. Discussing this level of detail is not the purpose of this paper, and it can be easily obtained via literature in neural network structure (Sexton et al., 1998; Malakooti and Zhou, 1994; Hill et al., 1996).

5.2. Training

Once a network has been structured for a particular application, it is ready to be trained. The network is trained with a training set of data representing the decision-makers' decisions and preferences in several situations to learn the decision-maker's behavior (Albino and Garavelli, 1998). Training is a process in which the network is presented with a desired output value, also called a knowledgeable teacher, for each pattern that is presented as input. This type of training is called supervised training. In supervised training, both the inputs and the outputs are provided. The network then processes the inputs and compares its resulting outputs against the desired outputs. Errors are then propagated back through the system, causing the system to adjust the weights, which control the network. This process occurs over and over as the weights are continually tweaked. The set of data, which enables the training, is called the training set. During the training of a network, the same set of data is processed many times as the connection weights are ever refined (Neural Computing—A Technology Handbook, 1993).

Some networks never learn for a number of reasons such as situations where the input data do not contain the specific information from which the desired output is derived. Networks also cannot converge if there is not enough data to enable complete learning. Ideally, there should be enough data so that part of the data can be held back as a test. If a network simply cannot solve the problem, with adequate and valid data, the designer must review the network structure.

Another part of the designer's creativity governs the rules of training. There are many laws (algorithms) used to implement the adaptive feedback required to adjust the weights during training. The most common technique is backward-error propagation, more commonly known as backpropagation. The backpropagation neural network is the most popular neural network paradigm used (Freeman and Skapura, 1992). When the system has been correctly trained, and no further learning is needed, the weights can be frozen.

To train the model accurately, the extreme (minimum and maximum) values in the population should be used in the training set. The size of this set is about 80% of the entire data set as a rule of thumb.

5.3. Testing

Many layered networks with multiple nodes are capable of memorizing data. To determine if the system is simply memorizing its data in some non-significant way, supervised training must hold back a set of data to test the system after it has undergone its training.

In the testing phase, the networks performance is checked by analyzing the results of some examples not used in the training set and comparing it to actual decisions (or scores) of the decision maker. The size of this set is about 20% of the entire data as a rule of thumb.

5.4. Implementing

Once implemented, the network then generates a score or rank for each of the potential suppliers in future.

6. Case study

In the following, 31 suppliers for 8 products of a company in the automotive industry are studied. This company is in charge of parts preparation for a car manufacturer, with all responsibilities such as dealing with suppliers, scheduling, quality, etc. The list of suppliers is shown in Table 1.

First a neural network model is designed based on managers' judgment for suppliers' performance evaluation. Second the drawbacks of this model are discussed and input and output of model are redesigned to improve the accuracy of the prediction. The improvement is validated and a sensitivity analysis is performed for the improved model as shown in Fig. 4.

6.1. Input

The criteria are quality (Q), delivery (D), technology (T), price (P) and location (L). Since historical and recent suppliers' data are needed for each individual criterion, a well-defined function for each criterion should be determined to convert the data into input data for the model. This function which covers performance history of suppliers can be very effective for the final result. However, it needs a database which includes the data for suppliers' performance. For example for delivery we need to know how many parts suppliers are behind schedule, or how much delay is based on their schedule. Suppliers might have several contracts. These should be considered in the function to show their performance as input.

To determine the role of performance data, first a model is designed. The decision makers of the system provided a simple

Table 1
Candidate suppliers for each product.

Products	Suppliers				
Product A	8	3	5	10	
Product B	5	6	13	8	
Product C	11	12	16	1	4
Product D	5	15	9		
Product E	14	7	17	18	
Product F	21	20	26	29	
Product G	23	25	24	28	30
Product H	19	22	27	31	

Table 2
The scale of criteria ratings.

Very Poor (VP)	$1 \leq W < 2$
Poor (P)	$2 \leq W < 3$
Poor Medium (PM)	$3 \leq W < 4$
Medium (M)	$4 \leq W < 6$
Medium Good (MG)	$6 \leq W < 7$
Good (G)	$7 \leq W < 8$
Very Good (VG)	$8 \leq W < 9$

table for the suppliers' performance as shown in Table 3. The scale of criteria ratings is shown in Table 2. Decision makers assigned a score to the suppliers from 1 to 9 based on criteria. The weight is the average score of two decision makers considered for each criterion as input. After completing Table 3, *W* columns as input data are ready to be used for training the neural network model.

6.2. Output

Decision makers assign evaluation scores to suppliers based on their performance and rank them. The score is the output of the decision making process. The goal is to have a model which produces a score (model output) for each supplier. The score for some suppliers is required for training purposes by the NN. To obtain these scores, a unique way of simulating the decision makers' judgment based on pairwise comparisons matrix is developed.

6.2.1. Pairwise comparison

One would like to be able to rank the alternatives (or criteria) in order of importance, and to assign to the alternatives (or criteria) some relative ranking indicating the degree of importance of each alternative (criterion) with respect to the other alternative. In other words, decision maker's judgment based on his or her preference for the pairwise comparisons of two alternatives is required. However, due to its use of unbalanced scale of judgments and its inability to adequately handle the inherent uncertainty and imprecision in the pairwise comparison process, fuzzy judgments were developed as a solution to tackle these criticisms (Deng, 1999).

Two methods of eliciting fuzzy judgment are described in the literature. The first method is eliciting information directly from fuzzy numbers (Tseng and Klein, 1989). When giving numerical values of strength of preferences, the decision makers are limited from 1 to 9. However, they are not restricted to integer numbers and have freedom to select any real number in the range of 1–9. The highest and the lowest preferences are assigned by 9 and 1, respectively.

Let $C=\{C_j|j=1, 2, \ldots, n\}$ be the set of criteria. Matrix A $(n \times n)$ shows the pairwise comparison on n criteria. The preference value of the comparison is shown as a_{ij} $(i,j=1, 2, \ldots, n)$.

$$A=\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{11} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}, \quad a_{ii}=1, \quad a_{ji}=1/a_{ij}, \quad a_{ij} \neq 0 \qquad (11)$$

In the second method, one uses linguistic variables which are a set whose elements are linguistic values that are names of fuzzy sets on the universe of discourse (Dubois and Prade, 1980). Direct method of obtaining the judgment data has been criticized as less effective than using linguistic variables (Zimmirmann, 1987). Here, we use the linguistic variables method to take advantage of its effectiveness.

Several methods have been illustrated in the literature to analyze the linguistic variables. The majority of approaches require cumbersome computations (Chen and Hwang, 1991), and they are not suitable for solving problems with more than 10 alternatives associated with more than 10 attributes. Also some of the methods require that the elements in the decision matrix be presented in a fuzzy format, even though they are crisp in nature. Chen and Hwang (1991) proposed a method that tackles these issues. This method is applied and illustrated for our case. This approach converts the fuzzy linguistic terms to crisp data. This transformation has two steps:

(1) *Converting linguistic terms to fuzzy numbers*: A numerical approximation system is used to systematically convert linguistic terms into their corresponding fuzzy numbers, which contain eight conversion scales. The principle of this system is simply to select a figure that contains all the verbal terms given by the decision maker and use the fuzzy numbers in that figure to represent the meaning of the verbal terms.
(2) *Converting the fuzzy number to crisp number*: Details of the process and calculations are shown in Appendix A. We consider the linguistic terms as very low, low, medium, high and very high to compare the suppliers. These terms are matched with scale 4 of the Chen and Hwang method, as shown in Fig. 5. Suppliers of product A are compared based on decision makers' preference as shown in Table 4.

The results of total score ($\mu_{T(M)}$) computations are summarized in Table 5.

We follow Saaty's procedure for pairwise comparisons in AHP (Saaty, 1980). As mentioned earlier, the scale of preference for decision makers is limited from 1 to 9. So values of $\mu_{T(M)}$ have been multiplied by 10 to fit into the scale. The new values of $\mu_{T(M)}$ are replaced in Table 4 for suppliers of product A, and the results are shown in Table 6.

The mathematical process begins by normalizing and finding the relative weights for each matrix. The relative weights are given by the right eigenvector (W) corresponding to the largest eigenvalue ($\lambda_{\max}$), as

$$A_w=\lambda_{max}W \qquad (12)$$

If the pairwise comparisons are completely consistent, the matrix A has rank 1, and $\lambda_{max}=n$. In this case, weights can be obtained by normalizing any of the rows or columns of A. Since each decision maker makes judgments individually, the consistency check should be considered separately for each decision maker. Decision maker(s) should revise their judgment about any comparisons that are inconsistent. The consistency index is computed in the following manner:

$$\mathrm{CI}=\frac{(\lambda_{max}-n)}{(n-1)} \qquad (13)$$

The consistency ratio is CR=CI/RI . Random index (RI) is the consistency index of a randomly generated reciprocal matrix from the scale 1–9, with forced reciprocally. Saaty (1980) suggested that if a consistency ratio is 0.1 or less, the consistency of the judgment is acceptable for each matrix.

Saaty (2005) provides arguments as to why pairwise comparisons are fundamentally a new paradigm because it creates relative scales from measurements based on an absolute scale of real numbers for making the comparisons. Paired comparisons imply dependence in the measurement of alternatives on the quality and number of other alternatives with which they are compared. Thus due to the dependence of the measurement of each alternative on the other alternatives, the ranking of alternatives could change if new alternatives are added or old ones are deleted. Contrast this with the traditional process of rating of alternatives one at a time, which presupposes that rank should

Table 3
Suppliers' performance (input).

Products	Suppliers	Decision makers	Criteria									
			Quality	*W* (average)	Delivery	*W* (average)	Technology	W (average)	Price	*W* (average)	Location	*W* (average)
A	8	DM1	6	5	7	6	5	6	5	5.5	3	2.5
		DM2	4		5		7		6		2	
	3	DM1	8	7.5	7	7.5	9	9	4	3	5	4.5
		DM2	7		8		9		2		4	
	5	DM1	4	5.5	7	7.5	6	5.5	3	3.5	7	7.5
		DM2	7		8		5		4		8	
	10	DM1	2	3	7	6	3	3.5	3	4	2	1.5
		DM2	4		5		4		5		1	
B	5	DM1	8	8.5	7	8	8	7.5	2	2.5	7	7.5
		DM2	9		9		7		3		8	
	6	DM1	8	7	9	8	8	7	3	4.5	1	2.5
		DM2	6		7		6		6		4	
	13	DM1	1	2	5	6.5	3	2	7	8	3	4
		DM2	3		8		1		9		5	
	8	DM1	6	7	5	6	4	4.5	7	6	2	3
		DM2	8		7		5		5		4	
C	11	DM1	8	7	8	6.5	8	7	6	5.5	7	8
		DM2	6		5		6		4		9	
	12	DM1	7	6	8	6.5	9	8	3	4.5	4	5
		DM2	5		5		7		6		6	
	16	DM1	7	5.5	8	6.5	7	5.5	8	7	5	5.5
			4		5		4		6		6	
	1	DM1	7	5.5	4	4.5	5	6	5	4.5	6	7
		DM2	4		5		7		4		8	
	4	DM1	2	3.5	4	5	7	6	8	6.5	4	3.5
		DM2	5		6		5		5		3	
D	5	DM1	8	6.5	8	7.5	5	6	8	6.5	2	3
		DM2	5		7		7		5		4	
	15	DM1	5	4	3	4	8	7	8	6.5	7	6
		DM2	3		5		6		5		5	
	9	DM1	4	5	5	6.5	2	3.5	3	3.5	7	8
		DM2	6		8		5		4		9	
E	14	DM1	8	8.5	6	7	8	6.5	3	4	2	3
		DM2	9		8		5		5		4	
	7	DM1	4	5	6	7	9	9	6	4.5	6	5
		DM2	6		8		9		3		4	
	17	DM1	8	6.5	5	6.5	7	6.5	5	6.5	7	5.5
		DM2	5		8		6		8		4	
	18	DM1	4	3.5	2	3.5	8	6	5	6	1	1.5
		DM2	3		5		4		7		2	
F	21	DM1	8	7.5	7	6	8	7	6	5.5	9	8
		DM2	7		5		6		5		7	
	20	DM1	4	5	6	6.5	7	8	6	5	5	6
		DM2	6		7		9		4		7	
	26	DM1	6	5	6	7	5	5.5	6	7	6	5.5
		DM2	4		8		6		8		5	
	29	DM1	9	8	9	8	9	8.5	5	6	3	4
		DM2	7		7		8		7		5	
G	23	DM1	3	4	5	6	8	7	6	5	9	9
		DM2	5		7		6		4		9	
	25	DM1	9	8	9	9	9	8	3	4	6	5
		DM2	7		9		7		5		4	
	24	DM1	7	6	8	7	9	9	8	7	9	8
		DM2	5		6		9		6		7	
	28	DM1	5	4	6	5.5	8	7	9	8	4	4
		DM2	3		5		6		7		4	
H	30	DM1	8	7	9	8	8	7	5	6	2	3
		DM2	6		7		6		7		4	
	19	DM1	7	8	9	9	9	8	6	5	6	4
		DM2	9		9		7		4		5	
	22	DM1	5	5	5	4	6	5.5	5	4	6	5
		DM2	5		3		5		3		4	
	27	DM1	7	6	8	7	6	6	6	5	7	6
		DM2	5		6		6		4		5	
	31	DM1	4	3	5	4	4	5	4	4	8	7
		DM2	2		3		6		4		6	

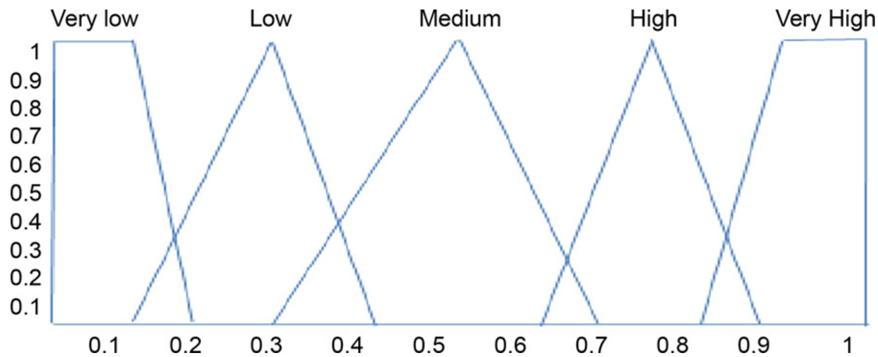

Fig. 5. Scale 4 of Chen and Hwang approach.

Table 4
Pairwise comparison for product A (DM1).

A	3	5	8	10
3	–	Medium	High	Very High
5	–	–	Low	Medium
8	–	–	–	Low
10	–	–	–	–

Table 5
Values of $\mu_{T(M)}$.

Very Low	0.09
Low	0.26
Medium	0.5
High	0.72
Very High	0.9

Table 6
Pairwise comparison for product A (DM1).

A	3	5	8	10
3	1	5	7.2	9
5	–	1	2.6	5
8	–	–	1	2.6
10	–	–	–	1

not be influenced unless new criteria are added or judgments changed. Saaty has illustrated and applied the pairwise comparison for the analytic hierarchy process (AHP) approach (Saaty, 2008).

After calculating weights for all suppliers, the input and output data are complete for use in the training step as shown in Table 7.

The next step is to train the model. Neurosolution software, version 5, was used to train the model. This leading edge neural network development software combines a modular, icon-based network design interface with an implementation of advanced learning procedures, such as conjugate gradients and backpropagation through time. NeuroSolutions adheres to the so-called local additive model. Under this model, each component can activate and learn using only its own weights and activations, and the activations of its neighbors (Neurosolution software). In order to use the software, first input and output should be defined. Second, the data should be classified into training and test sets. Third, the network's structure should be defined. Fourth, the software trains the model and shows the performance of the training set. Last, the test data set is used to evaluate the performance of the design network. This software is very comprehensive and user friendly in comparison with other software.

Several scenarios were tested, and the structure was designed as shown in Tables 8 and 9. In order to evaluate and validate the results of the model, the mean square error (MSE) was used based

Table 7
Input and output used in training.

Suppliers	Input					Output
	Q	*D*	*T*	*P*	*L*	Supplier score
8	5	6	6	5.5	2.5	0.12
3	7.5	7.5	9	3	4.5	0.59
5	5.5	7.5	5.5	3.5	7.5	0.23
10	3	6	3.5	4	1.5	0.05
5	8.5	8	7.5	2.5	7.5	0.46
6	7	8	7	4.5	2.5	0.34
13	2	6.5	2	8	4	0.22
8	7	6	4.5	6	3	0.10
11	7	6.5	7	5	8	0.47
12	6	6.5	8	4.5	5	0.27
16	5.5	6.5	5.5	7	5.5	0.19
1	5.5	4.5	6	4.5	7	0.15
4	3.5	5	6	6.5	3.5	0.13
5	6.5	7.5	6	6.5	3	0.36
15	4	4	7	6.5	6	0.25
9	5	6.5	3.5	3.5	8	0.16
14	8.5	7	6.5	4	7	0.61
7	5	7	9	4.5	5	0.23
17	6.5	6.5	6.5	6.5	5.5	0.11
18	3.5	3.5	6	6	1.5	0.05
21	7.5	6	7	5.5	8	0.49
20	5	6.5	8	5	6	0.30
26	5	7	5.5	7	5.5	0.19
29	8	8	8.5	6	4	0.64
23	4	6	7	5	9	0.19
25	8	9	8	4	5	0.59
24	6	7	9	7	8	0.42
28	4	5.5	7	8	4	0.28
30	7	8	7	6	3	0.44
19	8	9	8	5	4	0.61
22	5	4	5.5	4	5	0.11
27	6	7	6	5	6	0.48
31	3	4	5	4	7	0.07

Table 8
MSE after training the model with 1 hidden layer.

Learning rate	0.1			0.5			1		
Momentum	0.2	0.6	0.9	0.2	0.6	0.9	0.2	0.6	0.9
Hidden nodes	2	2	2	4	4	4	6	6	6
tanh	0.06	0.08	0.06	0.05	0.09	0.07	0.09	0.06	0.05
Sigmoid	0.04	0.09	0.08	0.01	0.07	0.05	0.06	0.07	0.08

on the following function:

$$\mathrm{MSE}=\frac{\sum_{j=0}^{p}\sum_{i=0}^{n}(d_{ij}-y_{ij})^2}{np} \tag{14}$$

where p is the number of output processing elements, n is the number of exemplars in the data set, y_{ij} is the network output for exemplar i (sample) at processing element j, and d_{ij} is the desired output for exemplar i at processing element j.

The performance of the model based on one hidden layer with the sigmoid transfer function showed the best results. The learning rate and momentum values were 0.5 and 0.2, respectively, with four hidden nodes. As earlier mentioned, there is no formula to find an optimal solution for the best structure. Testing all possible combinations of factors is a cumbersome task; however, trial and error and rules of thumb could facilitate to design the structure.

Fig. 6 shows the expected declining trend of mean square error (MSE) during training. After training, eight suppliers are considered as test samples to be sure the model can create valid results. The outputs of the model (scores) are compared with the scores of suppliers from the pairwise comparison matrices, as shown in Fig. 6. Fig. 7 depicts the test results.

If the test results prove that the model can predict the score of suppliers within an acceptable range, the model can be used for suppliers' evaluation. In future, whenever a comparison among the suppliers is needed, the model uses the input and ranks the suppliers without decision makers' judgment.

7. Improving the designed input and output

We designed a unique manner of input and output for the neural network; however, there are several hidden issues in the input and the output values. These issues can stimulate noise and error in the training step and test results.

7.1. New input

To mitigate the impact of bias in the judgment of managers about the suppliers' performance evaluation, a meticulous analysis is needed. The following solution is proposed to quantify the performance of suppliers.

For each criterion, a function is defined to quantify suppliers' performance. Historical and recent suppliers' data are needed for each individual criterion, and a function for each criterion is defined to convert the raw data into input data for the model. Over time, the vendor can update suppliers' performance based on their contracts. The suppliers' performance is evaluated based on the following criteria:

7.1.1. Quality

To measure the supplier's performance, we defined the quality history of supplier, which is the ratio of the defective parts to the total number of parts supplied. The input function applied for quality is

$$Q=\frac{\sum_{j}^{m}(d_j)}{tn} \tag{15}$$

where Q is the input function for quality, j is the delivery number (batch) for a contract, $j=1,\ldots,m$, m is the number of deliveries or batches for a contract, d_j is the number of defective parts in delivery number j, and tn is the total number of delivered products.

7.1.2. Delivery

Quantity and delay in delivery are the two parameters of suppliers' performance that are selected. Therefore, the delivered quantity and punctuality can indicate the supplier performance. The input function applied for delivery is

$$dq=\sum_{j=1}^{m}(|q_{jp}-q_{ja}|/q_{jp})/m \tag{16}$$

$$dt=\sum_{j=1}^{m}(|t_{jp}-t_{ja}|/t_{jp})/m \tag{17}$$

$$D=(dq+dt) \tag{18}$$

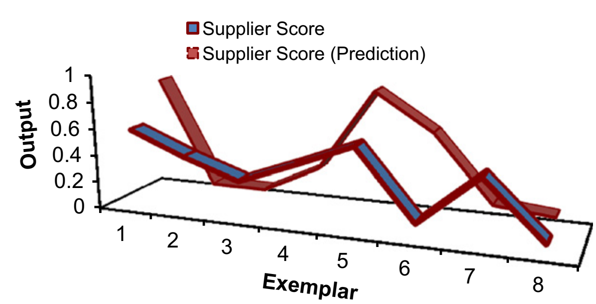

Fig. 7. Desired output and actual network output.

Table 9
MSE after training the model with 2 hidden layers.

Learning rate	0.1			0.5			1		
Momentum	0.2	0.6	0.9	0.2	0.6	0.9	0.2	0.6	0.9
Hidden Nodes	2	2	2	4	4	4	6	6	6
tanh	0.07	0.06	0.04	0.07	0.09	0.08	0.09	0.06	0.05
Sigmoid	0.04	0.05	0.06	0.04	0.06	0.07	0.05	0.04	0.04

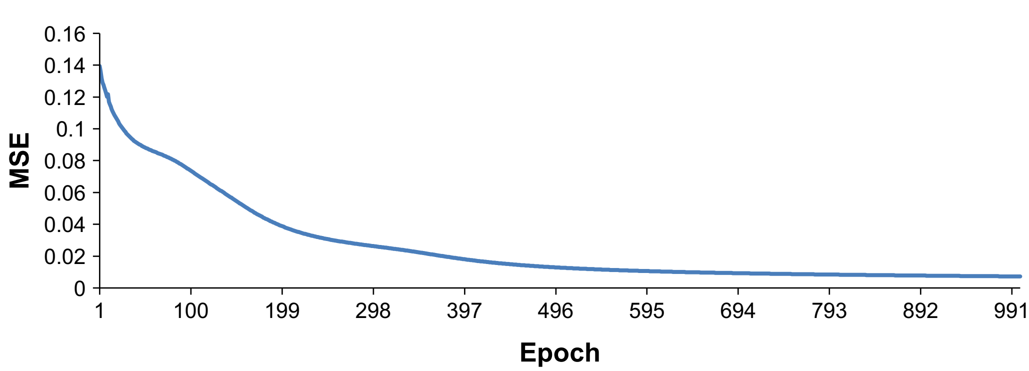

Fig. 6. MSE versus epoch.

where dq is the quantity evaluation function, q_{jp} is the planning quantity based on contract, q_{ja} is the actual delivered quantity, t_{jp} is the planning time of delivery based on contract, t_{ja} is the actual delivery time, dt is the on time evaluation function, D is the input function for delivery, j is the delivery number (batch) for a contract $j=1,\ldots,m$, and m is the number of deliveries or batches for a contract

7.1.3. Price

This criterion plays a critical role in decision making. In this research, discount issues and types of payment are converted together into a final offer price from supplier to buyer. Since price is a tangible criterion for comparison, the price offered from a supplier is considered for comparison.

7.1.4. Location

Since transportation cost (TC) is a tangible criterion for comparison, the TCs of suppliers are considered for comparison.

7.1.5. Technology

The most common way to quantify the technology level of a supplier is to evaluate the system based on several factors (design, materials, machinery, process, etc.). In our case study, the expert had a procedure to evaluate suppliers, and a supplier is rated on a scale of 10 (low) to 100 (high).

7.2. New output

Since the concept of pairwise comparison is to compare suppliers of a product with each other (and not how well a suppliers' performance is in comparison with an ideal supplier) the values in the pairwise comparison are assigned to that set of suppliers from 1 to 9. Therefore we face two issues:

(1) A group of candidate suppliers competing for a product are being compared with each other. If the performance of the best supplier in this group is not similar to the best suppliers in the other groups for other products in the training set, the output score for this group will not be very accurate for the training purpose.
(2) If suppliers of a product have similar performance and they are among the highly qualified suppliers from the pool of all the suppliers' products, in the pairwise comparison matrix, low preference values are used (e.g. 3 out of 9) which result in low values as output. However, we expect to have high output values for high performing suppliers. In the following example, the issues are clarified.

Consider eight suppliers for two products, W and Z (these are not real products as a part of our case study). We created the following pairwise comparison matrices, which show that supplier 'a' is extremely better than supplier 'd' for product W and supplier 'e' is extremely better than supplier 'h' for product Z, as shown in Tables 10 and 11. After following the weight calculation procedure, the output results are shown in Table 12.

Table 10
Pairwise comparison for product W.

Product W	a	b	c	d
A	1	7	8	9
B	–	1	2	4
C	–	–	1	2
D	–	–	–	1

Table 11
Pairwise comparison for product Z.

Product Z	e	f	g	h
E	1	7	8	9
F	–	1	2	4
G	–	–	1	2
H	–	–	–	1

Table 12
Sample training set based on products W & Z.

Suppliers	**Input**					**Output**
	Quality	Delivery	Technology	Price	Location	Supplier score
a	9	9	9	9	9	0.69
e	6	6	6	6	6	0.69

Table 13
Scale for supplier comparison with an ideal supplier.

The difference of the best supplier with the ideal supplier	Score
The same	1
Low	1.2
Low to medium	1.4
Medium	1.6
Medium to high	1.8
High	2

Table 14
Comparison of supplier **a** with the ideal supplier.

	Supplier a
Ideal supplier	1.6

Table 15
Revised pairwise comparison matrix for Product W.

Product W	**a**	**b**	**c**	**d**
a	1	7∗1/1.6	8∗1/1.6	9∗1/1.6
b	–	1	2∗1/1.6	4∗1/1.6
c	–	–	1	2∗1/1.6
d	–	–	–	1

The performance of suppliers 'a' and 'e' (input) based on the defined criteria is shown in Table 12. Although both suppliers have the same output scores, they have different input values for NN model, which can create errors in the training pattern. This is one of the issues that we face. In pairwise comparison, suppliers are being compared based on each other. Being the best supplier among a group of suppliers which are competing for a product does not mean that it is absolutely the best or at the same level of other high qualified suppliers in the training data set resulted from several products. Therefore, creating an output set from different products needs more attention.

To mitigate the impact of this issue on the training data, a method is proposed as follows:

(1) *Define a scale for the comparison of suppliers*: In order to increase the accuracy of output, a standard table for supplier evaluation is created as shown in Table 13.

(2) Convert the comparison values based on the defined standard.
(3) Recalculate the outputs.

In each pairwise comparison matrix (for each product), the supplier which has the highest score is selected and compared with an ideal supplier. The scale is from 1 to 2 as shown in Table 13, which means if performance of the selected supplier is at the same level of the ideal supplier, the score is 1 and the pairwise comparison is not changed, but if the selected supplier in comparison with the ideal supplier is assigned any number more than 1, the original pairwise comparison of suppliers should be revised. The rationale behind the new scale is as follows.

First, converting the values based on the ratio should not create a fraction value, for instance 1/3, which is not part of pairwise comparison scale (1–9). In general, the lowest grade that shows preference of one supplier to the other in pairwise comparison is 2 out of 9 (1 out of 9 means no preference). Therefore, having a ratio of 1/2 (it happens in an extreme case; in comparison with the ideal supplier), makes the converted number equal to 1 which is in the range of 1–9. Second, in a competitive market, considering the performance of an ideal supplier more than two times better than the best supplier for a product might not be realistic. The following example illustrates the above steps.

The best supplier for product W, Table 10, is supplier 'a'. In general, it can be recognized by following the weight calculation procedure. Comparing supplier 'a' with an ideal supplier, as shown in Table 14, leads to the next step. Since the score is 1.6 as shown in Table 14, numbers in the original Table 10 should be replaced, based on the ratio of 1/1.6 as shown in Table 15.

Now, the new outputs are calculated based on Table 15. These changes overcome the aforementioned issue, and the new output should improve the quality of the training data. To validate this improvement, inputs and outputs are modeled based on the redesigned procedure. The new input and output values are shown in Table 16.

After training the model and testing it for eight samples, MSE values for training and test results depict a reduction after redesigning input and output, as shown in Table 17. More details are shown in Figs. 8 and 9 and Tables 18 and 19.

Table 16
New input and output values.

Suppliers	**Input**					**Output**
	Q	*D*	*T*	*P*	*L*	Supplier score
8	0.009	0.12	70	0.25	0.11	0.10
3	0.007	0.06	90	0.40	0.22	0.52
5	0.014	0.07	65	0.35	0.13	0.19
10	0.031	0.14	50	0.29	0.25	0.04
5	0.005	0.05	80	0.44	0.13	0.41
6	0.012	0.03	75	0.31	0.11	0.26
13	0.056	0.11	70	0.22	0.17	0.19
8	0.012	0.18	60	0.27	0.18	0.08
11	0.01	0.09	75	0.25	0.09	0.42
12	0.014	0.17	85	0.29	0.16	0.23
16	0.019	0.22	55	0.19	0.15	0.14
1	0.021	0.19	60	0.27	0.11	0.11
4	0.039	0.21	60	0.23	0.19	0.08
5	0.012	0.08	60	0.25	0.18	0.34
15	0.024	0.13	70	0.22	0.13	0.21
9	0.011	0.01	55	0.34	0.09	0.12
14	0.003	0.04	90	0.29	0.11	0.53
7	0.014	0.15	90	0.31	0.16	0.18
17	0.021	0.19	65	0.24	0.15	0.1
18	0.078	0.28	60	0.23	0.25	0.03
21	0.01	0.09	80	0.27	0.09	0.46
20	0.013	0.11	90	0.26	0.13	0.26
26	0.023	0.06	60	0.22	0.15	0.13
29	0.005	0.03	90	0.24	0.17	0.63
23	0.009	0.12	80	0.28	0.08	0.14
25	0.004	0.01	85	0.35	0.16	0.54
24	0.009	0.08	70	0.23	0.09	0.34
28	0.018	0.19	70	0.2	0.17	0.21
30	0.009	0.05	80	0.26	0.18	0.32
19	0.003	0.01	85	0.29	0.17	0.56
22	0.039	0.13	55	0.37	0.16	0.06
27	0.013	0.08	70	0.31	0.13	0.40
31	0.038	0.13	55	0.33	0.11	0.04

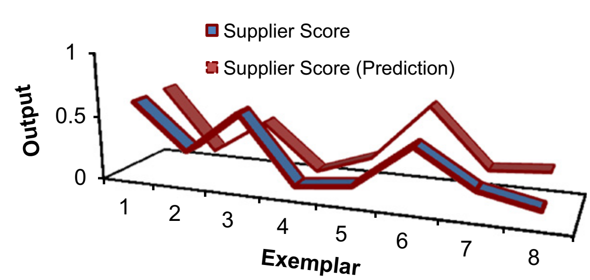

Fig. 9. Desired output and actual network output.

Table 17
MSE comparison for both designs.

	Training	Test
First design	0.01	0.09
New design	0.002	0.01

Table 18
MSE of training set.

Best network	**Training**
Epoch #	1000
Minimum MSE	0.002

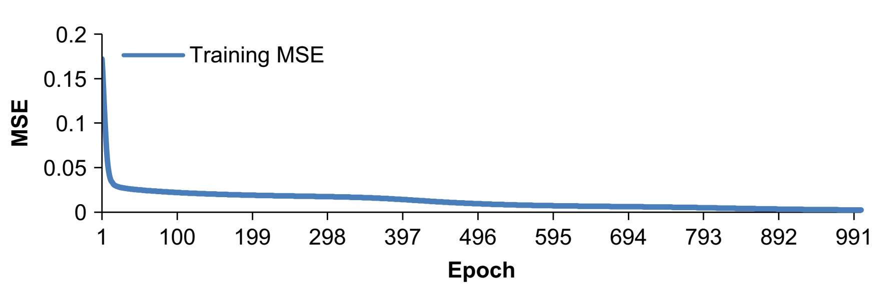

Fig. 8. MSE versus epoch.

8. Sensitivity analysis

A sensitivity analysis was performed to measure the relative importance among the inputs of the neural model and illustrate how the model output varies in response to variation of the input. The network learning is disabled during this operation so that the network weights are not affected. The first input is varied between its mean and 10 standard deviations while all other inputs are fixed at their respective means. The network output is computed for 100 numbers of steps above and below the mean, the number of discrete values used to calculate the output. This process is repeated for each input individually. The model is not trained for any data either over the maximum values or below minimum values as shown in Table 20.

Plots showing the network output over the range of varied input are shown in Figs. 10–14.

The maximum value for quality in the training data set is 0.078. Fig. 6 shows that by increasing the quality values for suppliers, the model creates lower output values in quality axis; the model can create an acceptable result for any future data outside the training data range.

The maximum value for delivery in the training data set is 0.28. Fig. 11 shows the trend of varied delivery values. By increasing the delivery values for suppliers, this graph shows an expected drop in output values, but the graph shows that the slope of this trend is almost flat after point 0.3. Thus, if the delivery input values go beyond point 0.3, it may result in more conservative decisions.

Fig. 12 shows that by increasing the technology values for suppliers, the model results in a higher output value, and the model can create acceptable results for data outside the training data range.

Fig. 13 depicts that by increasing the price, the model results in lower output values and the model can create an acceptable result for any data outside the trained data range.

The relation of suppliers' location and the output is shown in Fig. 14. The trend shows that the greater the distance, more qualified suppliers will be available.

Table 19
Test performance.

Performance	Supplier score
MSE	0.0193
NMSE	0.4485
MAE	0.1070
Min. abs. error	0.0140
Max. abs. error	0.2779
r	0.7733

Table 20
Max. and min. values of training data set.

	Quality	Delivery	Technology	Price	Location
Max.	0.078	0.28	90	0.44	0.25
Min.	0.003	0.01	50	0.19	0.08

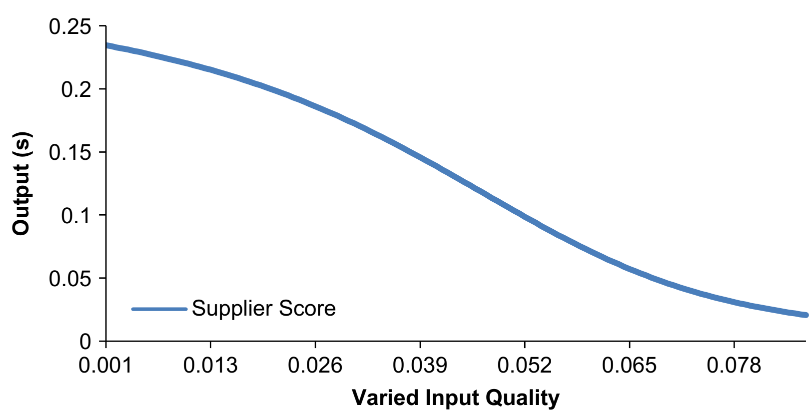

Fig. 10. Network output(s) for varied input quality.

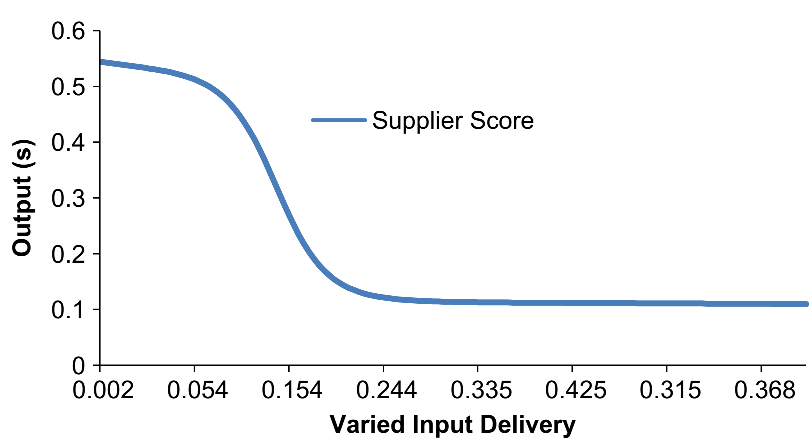

Fig. 11. Network output(s) for varied input delivery.

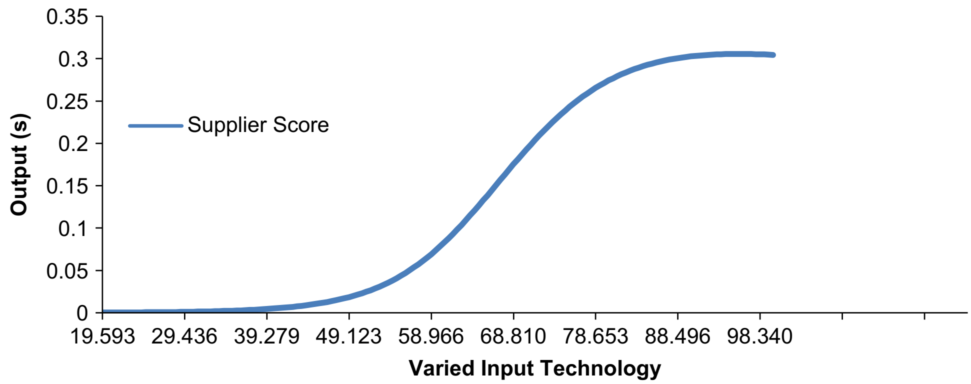

Fig. 12. Network output(s) for varied input technology.

Fig. 13. Network output(s) for varied input price.

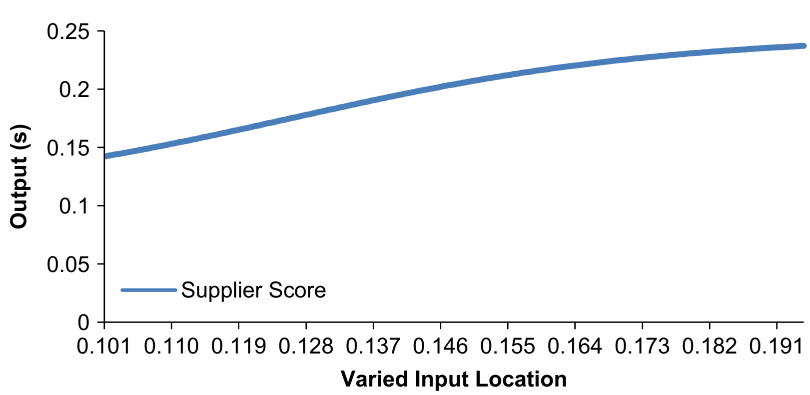

Fig. 14. Network output(s) for varied input location.

Table 21
A sample of input and output used in training.

Car model	Input						Output
	Price	Horsepower	Cylinder	Maintenance cost (annually)	MPG	Size	Score
A	12,000	110	4	1000	34	Economy	12.4
B	14,400	125	4	1500	30	Medium	14
C	13,000	115	4	1300	28	Economy	13
D	19,900	176	6	1800	25	Full	13.6

9. Another application

Several rental companies every year make decisions pertaining to purchasing new cars or selling some of their used cars. To make a purchase, several options are available. Based on several criteria such as price, horse power, reliability, maintenance cost, gas consumption rate, and size, decision makers make their decision. The values for these criteria are given as input and the scores or ranking of these cars are the output. Table 21 shows a sample of created input and output of the training data set.

After considering a structure for the neural network model, training and testing, the model can provide a ranking for the future use. Next year, the input data from manufacturers is given as input and the model assigns a score or rank to these options.

10. Conclusion

The steps to make a decision making model based on the neural networks were discussed with a couple of applications. These steps are valid for any application. The model design (input and output), performance history consideration, and prediction capability make the proposed model unique in comparison with other neural networks methods for solving multiple criteria decision-making problems. Analyzing and improving the pairwise comparison technique for mapping the managers' preference in the training section made our method exclusive. The vendor can update the suppliers' database information over time for future decisions without managers' judgment about the suppliers' evaluation.

The model is trained and it learns to simulate the way the decision makers' judge and make decisions.

The condition of having valuable results of the model is to have adequate valid data and well-designed structure. If the training data set is a good sample of the population, the model will provide useful results.

Appendix A. Fuzzy calculation

The following expressions show how the crisp number of a fuzzy number M is obtained. Given a maximizing set and a minimizing set as

$$\mu_{max}(x) = \begin{cases} x, & 0 \le x \le 1 \\ 0, & \text{otherwise} \end{cases} \tag{A1}$$

$$\mu_{min}(x) = \begin{cases} 1-x, & 0 \le x \le 1 \\ 0, & \text{otherwise} \end{cases} \tag{A2}$$

The right score of M is determined as

$$\mu_{R(M)} = \sup[\mu M(x) \wedge \max(x)] \tag{A3}$$

The right score is the intersection of the line $y=x$ and the right side of the number M. Also, support of a fuzzy set (sup) means elements with nonzero membership grades.

The left score of M is determined as

$$\mu_{L(M)} = \sup[\mu M(x) \wedge \mu_{min}(x)] \tag{A4}$$

Basically, the left score is the intersection of the line $y=1-x$ and the left side of the fuzzy number M.

The total score of M was computed as

$$\mu_{T(M)} = \left[\mu_{R(M)} + 1 - \mu_{L(M)}\right]/2 \tag{A5}$$

Now converting the fuzzy number to crisp number is performed as follows:

$$\mu_{very\ low} = \begin{cases} 1, & x < 0.1, \\ \frac{0.2-x}{0.1}, & x \ge 0.1 \end{cases} \tag{A6}$$

$$\mu_{low} = \begin{cases} \frac{x-0.1}{0.15}, & 0.1 \le x < 0.25, \\ \frac{0.4-x}{0.15}, & 0.25 \le x < 0.4 \end{cases} \tag{A7}$$

$$\mu_{medium} = \begin{cases} \frac{x-0.3}{0.2}, & 0.3 \le x < 0.5, \\ \frac{0.7-x}{0.2}, & 0.5 \le x < 0.7 \end{cases} \tag{A8}$$

$$\mu_{high} = \begin{cases} \frac{x-0.6}{0.15}, & 0.1 \le x < 0.75, \\ \frac{0.9-x}{0.15}, & 0.75 \le x < 0.9 \end{cases} \tag{A9}$$

$$\mu_{very\ high} = \begin{cases} \frac{x-0.8}{0.1}, & 0.8 \le x < 0.9, \\ 1, & x < 0.9 \end{cases} \tag{A10}$$

In order to determine $\mu_{T(M)}$, $\mu_{R(M)}$ and $\mu_{L(M)}$ should be computed. For instance, $\mu_{T(M)}$, for very low is

μ_R(Very Low)$=0.2-X/0.1=X$. So $X=0.18$ and μ_R(low)$=0.18$, μ_L(low)$=1$, $\mu_{T(M)}=[0.18+1-1]/2=0.09$.

References

Awasthi, A., Chauhan, Satyaveer S., Goyal, S.K., 2010. A fuzzy multi criteria approach for evaluating environmental performance of suppliers. International Journal of Production Economics 126, 370–378.

Albino, V., Garavelli, A., 1998. A neural network application to subcontractor rating in construction firms. International Journal of Project Management 16 (1), 9–14.

Bard, J.F., 1992. A comparison of the AHP process with Multi-attribute Utility Theory: a case study. IIE Transaction 24 (5), 111–121.

Bhutta, K.S., Huq, F., 2002. Supplier selection problem: a comparison of the total cost of ownership and analytic hierarchy process approaches. Supply Chain Management 7 (3), 126–135.

Chen, J., Lin, S., 2004. A neural network approach—decision neural network (DNN) for preference assessment. IEEE Transactions on Systems, Man and Cybernetics—Part C: Applications and Reviews 34 (2).

Chen, S.J., Hwang, L., 1991. Fuzzy Multiple Attribute Decision Making. Springer Verlag Publication.

Chen, C.T., Lin, Ching-Torng, Huang, Sue-Fn, 2006. A fuzzy approach for supplier evaluation and selection in supply chain management. International Journal of Production Economics 102, 179–190.

Dahel, N.E., 2003. Vendor selection and order quantity allocation in volume discount environments. Supply Chain Management 8 (4), 335–342.

Deng, H., 1999. Multi criteria analysis with fuzzy pair-wise comparison. International Journal of Approximate Reasoning 21.

Denton, J.W., Hung, M.S., 1996. A comparison of nonlinear optimization methods for supervised learning in multiplayer feed forward neural networks. European Journal of Operational Research 93 (2).

Dorsey, R.E., Mayer, W.J., 1994. Optimization using genetic algorithms, advances, artificial intelligence in economics. Finance, and Management 1, 69–91.

Dow, R., Sietswa, J., 1991. Creating artificial neural networks that generalize. IEEE Neural Networks 4 (1).

Dubois, D., Prade, H., 1980. Fuzzy Sets and Systems: Theory and Applications. Academic Press, New York.

Ellram, L.M., 1990. The supplier selection decision in strategic partnerships. Journal of Purchasing and Materials Management 26 (4), 8–14.

Ellram, L.M., 1995. Total cost of ownership: an analysis approach for purchasing. International Journal of Physical Distribution & Logistics Management 25 (8), 4–23.

Freeman, J.A., Skapura, D.M.,1992. Neural Network Algorithms—Application and Programming Techniques Addison-Wesley Publishing Company).

Golmohammadi, D., Creese, R., Valian, H., 2009. Neural network application for supplier selection. International Journal of Product Development 8 (3).

Ghodsypour, S.H., O'Brien, C., 1998. A decision support system for supplier selection using an integrated analytic hierarchy process and linear programming. International Journal of Production Economics 56–57, 199–212.

Hill, T., O'Connor, M., Remus, W., 1996. Neural network models for time series forecasting. Management Science 42 (7).

Liu, F., Ding, F.Y., Lall, V., 2000. Using data envelopment analysis to compare suppliers for supplier selection and performance improvement. Supply Chain Management 5 (3), 143–150.

Malakooti, B., Zhou, Y.Q., 1994. Feed forward artificial neural networks for solving discrete multiple criteria decision making problems. Management Science 40 (11), 1542–1561.

Neural Computing—A Technology Handbook for Neural Works Explorer, Neural Ware Inc., Technical Publications Group, 1993, pp. 5–239.

Neurosolution software. 〈http://www.neurosolutions.com/〉.

Saaty, T.L., 1980. The Analytic Hierarchy Process. McGraw-Hill, New York.

Saaty, T.L., 2005. Theory and Applications of the Analytic Network Process. RWS Publications, Pittsburgh, PA.

Saaty, T.L., 2008. Relative measurement and its generalization in decision making why pairwise comparisons are central in mathematics for the measurement of intangible factors the analytic hierarchy/network process. Revista de la Real Academia de Ciencias Series A: Materials 102 (2), 251–318.

Sun, M., Stam, A., Steuer, R.E., 1996. Solving multiple objective programming problems using feed-forward artificial neural networks: the interactive FFANN procedure. Management Science 42 (6).

Seydel, J., 2005. Supporting the paradigm shift in vendor selection: multi-criteria methods for sole sourcing. Managerial Finance 31 (3), 49–64.

Sexton, R.S., Dorsey, R.E., Johnson, J.D., 1998. Toward global optimization of neural networks: a comparison of the genetic algorithm and backpropagation. Decision Support Systems 22, 171–186.

Shih, H., Wen, U., Lee, Minglan, Chyihsiao, E.K., 2004. A Neural Network Approach to Multi Objective and Multilevel Programming Problems. Computers and Mathematics with Applications vol.48, 95–108.

Siying, W., Jinlong, Z., Zhicheng , L.,1997. A supplier selecting system using a neural network. In: IEEE International Conference on Intelligent Processing Systems.

Singh, R.K., Choudhury, A.K., Tiwari, M.K., Shankar, R., 2007. Improved Decision Neural Network (IDNN) based consensus method to solve a multi-objective group decision making problem. Advanced Engineering Informatics 21, 335–348.

Szolovits, P., Pauker, S.G., 1978. Categorial and possibilistic reasoning in medical diagnosis. Artificial Intelligence 11.

Sun, M., Stam, A., Steuer, R., 2007. Interactive multiple objective programming using Tchebyche! programs and artificial neural networks. Computers & Operations Research 27, 601–620.

Tseng, T.Y., Kein, 1989. New algorithm for the ranking procedure in fuzzy decision making. IEEE Transaction on Systems, Man and Cybernetics 19 (5).

Tan, K.H., Lim, Chee Peng, Platts, Ken, Hooi, Shen Koay, 2006. An intelligent decision support system for manufacturing technology investments. International Journal of Production Economics 104, 179–190.

Weber, C.A., Current, J.R., Desai, A., 1998. Non-cooperative negotiation strategies for vendor selection. European Journal of Operational Research 108, 208–223.

Weber, C.A., Current, J.R., Desai, A., 2000. An optimization approach to determining the number of vendors to employ. Supply Chain Management 5 (2), 90–98.

Weber, C.A., 1996. A data envelopment analysis approach to measuring vendor performance. Supply Chain Management 1 (1), 28–39.

Wu, D., 2009. Supplier selection: a hybrid model using DEA, decision tree and neural network. Expert Systems with Applications 36, 9105–9112.

Zadeh, L.A., 1965. Fuzzy sets. Information and Control 8, 338–353.

Zhang, G.P., Patuwo, B.E., Hu, M.Y., 2001. A simulation study of artificial neural networks for nonlinear time series forecasting. Computer and Operations Research 28.

Zimmirmann, H.J., 1987. Fuzzy Sets Decision Making and Expert Systems. Kluwer Academic Publishers, Boston, MA.