2nd International IFS Conference, Mersin, Turkey, 14–18 Oct. 2015
Notes on Intuitionistic Fuzzy Sets
ISSN 1310–4926
Vol. 21, 2015, No. 4, 143–152

Opportunities for application of the intercriteria analysis method to neural network preprocessing procedures

Sotir Sotirov

Laboratory of Intelligent Systems, University "Prof. Dr. Assen Zlatarov"
1 "Prof. Yakimov" Blvd., Burgas 8010, Bulgaria
e-mail: ssotirov@btu.bg

Abstract: The artificial neural networks (ANN) are a tool that can be used for object recognition and identification. However, there are certain limits when we may use ANN, and the number of the neurons is one of the major parameters during the implementation of the ANN. On the other hand, the bigger number of neurons slows down the learning process. In our paper, we use a method for removing the number of the neurons without reducing the error between the target value and the real value obtained at the output of the ANN's output. The method uses the recently proposed approach of InterCriteria Analysis, based on index matrices and intuitionistic fuzzy sets, which aims to detect possible correlations between pairs of criteria. In this paper we use the data from 11 criteria of crude oil measurements.
Keywords: Intercriteria analysis, Intuitionistic fuzzy sets, Neural networks, Crude oil.
AMS Classification: 03E72.

1 Introduction

The use of neural networks is often hampered by the large number of resources which are needed. When using a large number of parameters as data at the input of the NN, the question of reducing part of the data arises. Upon their removal the large number of inputs of the NN is reduced as well as the number of weight coefficients in it. In a previous paper [18] we used the data of 11 criteria of crude oil measurements. In this case we use 11 parameters of biodiesel fuels.

For the reduction of the inputs of the NN we will use InterCriteria Analysis to find dependencies of all pairs of parameters applied to the input of the NN

2 Presentation of the InterCriteria Analysis

The presented method, titled InterCriteria Analysis (ICrA) [12] is based on two fundamental concepts: intuitionistic fuzzy sets and index matrices.

Intuitionistic fuzzy sets defined by Atanassov [3–6] represent an extension of the concept of fuzzy sets, as defined by Zadeh [15], exhibiting function $\mu_A(x)$ defining the membership of an element x to the set A, evaluated in the interval [0; 1]. The difference between fuzzy sets and intuitionistic fuzzy sets (IFSs) is in the presence of a second function $\nu_A(x)$ defining the non-membership of the element x to the set A, where $\mu_A(x) \in [0; 1]$, $\nu_A(x) \in [0; 1]$, under the condition of $(\mu_A(x) + \nu_A(x)) \in [0; 1]$. The IFS itself is formally denoted by:

$$A = \{\langle x, \mu_A(x), \nu_A(x)\rangle \mid x \in E\}.$$

Comparison between elements of any two IFSs, say A and B, involves pairwise comparisons between their respective elements' degrees of membership and non-membership to both sets.

The second concept on which the proposed method relies is the concept of index matrix, a matrix which features two index sets. The theory behind the index matrices is described in [1]. Here we will start with the index matrix M with index sets with m rows $\{O_1, \ldots, O_m\}$ and n columns $\{C_1, \ldots, C_n\}$, where for every p, q $(1 \le p \le m, 1 \le q \le n)$, O_p in an evaluated object, C_q is a evaluation criterion, and $e_{O_pC_q}$ is the evaluation of the p-th object against the q-th criterion, defined as a real number or another object that is comparable according to relation R with all the rest elements of the index matrix M.

$$M = \begin{array}{c|ccccccc}
 & C_1 & \cdots & C_k & \cdots & C_l & \cdots & C_n \\ \hline
O_1 & e_{O_1,C_1} & \cdots & e_{O_1,C_k} & \cdots & e_{O_1,C_l} & \cdots & e_{O_1,C_n} \\
\vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\
O_i & e_{O_i,C_1} & \cdots & e_{O_i,C_k} & \cdots & e_{O_i,C_l} & \cdots & e_{O_i,C_n} \\
\vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\
O_j & e_{O_j,C_1} & \cdots & e_{O_j,C_k} & \cdots & e_{O_j,C_l} & \cdots & e_{O_j,C_n} \\
\vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\
O_m & e_{O_m,C_1} & \cdots & e_{O_m,C_j} & \cdots & e_{O_m,C_l} & \cdots & e_{O_m,C_n}
\end{array},$$

From the requirement for comparability above, it follows that for each i, j, k it holds the relation $R(a_{O_iC_k}, a_{O_jC_k})$. The relation R has dual relation $\overline{R}$, which is true in the cases when relation R is false, and vice versa. For the needs of our decision making method, pairwise comparisons between every two different criteria are made along all evaluated objects. During the comparison, it is maintained one counter of the number of times when the relation R holds, and another counter for the dual relation.

Let $S^{\mu}_{k,l}$ be the number of cases in which the relations $R(e_{O_iC_k}, e_{O_jC_k})$ and $R(e_{O_i\,C_l}, e_{O_jC_l})$ are simultaneously satisfied. Let also $S^{\nu}_{k,l}$ be the number of cases in which the relations $R(e_{O_iC_k}, e_{O_jC_k})$ and its dual $\overline{R}(e_{O_iC_l}, e_{O_jC_l})$ are simultaneously satisfied. As the total number of pairwise comparisons between the object is $m(m-1)/2$, it is seen that there hold the inequalities:

$$0 \le S^{\mu}_{k,l} + S^{\nu}_{k,l} \le \frac{n(n-1)}{2}.$$

For every k, l, such that $1 \le k \le l \le m$, and for $n \ge 2$ two numbers are defined:

$$\mu_{C_k,C_l} = 2\frac{S^{\mu}_{k,l}}{n(n-1)},\ \nu_{C_k,C_l} = 2\frac{S^{\nu}_{k,l}}{n(n-1)}.$$

The pair constructed from these two numbers plays the role of the intuitionistic fuzzy evaluation of the relations that can be established between any two criteria C_k and C_l. In this way the index matrix M that relates evaluated objects with evaluating criteria can be transformed to another index matrix M^* that gives the relations among the criteria:

$$M^* = \begin{array}{c|ccc} & C_1 & \cdots & C_m \\ \hline C_1 & \langle \mu_{C_1,C_1}, \nu_{C_1,C_1} \rangle & \cdots & \langle \mu_{C_1,C_m}, \nu_{C_1,C_m} \rangle \\ \vdots & \vdots & \ddots & \vdots \\ C_m & \langle \mu_{C_m,C_1}, \nu_{C_1,C_m} \rangle & \cdots & \langle \mu_{C_m,C_m}, \nu_{C_m,C_m} \rangle \end{array}.$$

From practical considerations, it has been more flexible to work with two index matrices M^{μ} and M^{ν}, rather than with the index matrix M^* of IF pairs.

The final step of the algorithm is to determine the degrees of correlation between the criteria, depending on the user's choice of μ and ν. We call these correlations between the criteria: 'positive consonance', 'negative consonance' or 'dissonance'. Let $\alpha, \beta \in [0; 1]$ be the threshold values, against which we compare the values of μ_{C_k,C_l} and ν_{C_k,C_l}. We call that criteria C_k and C_l are in (α, β)-positive consonance, if $\mu_{C_k,C_l} > \alpha$ and $\nu_{C_k,C_l} < \beta$; (α, β)-negative consonance, if $\mu_{C_k,C_l} < \beta$ and $\nu_{C_k,C_l} > \alpha$; (α, β)-dissonance, otherwise.

Obviously, the larger α and/or the smaller β, the less number of criteria may be simultaneously connected with the relation of (α, β)-positive consonance. For practical purposes, it carries the most information when either the positive or the negative consonance is as large as possible, while the cases of dissonance are less informative and are skipped.

3 Artificial neural networks

The artificial neural networks [9, 10] are one of the tools that can be used for object recognition and identification. In the first step it have to be learned and after that we can use for the recognitions and for predictions of the properties of the materials. Fig. 1 shows in abbreviated notation of a classic two-layered neural network.

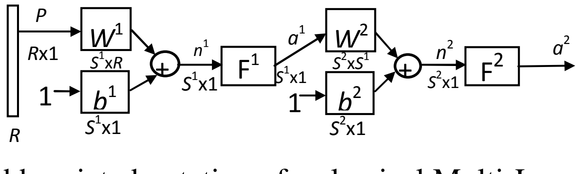

Figure 1. Abbreviated notation of a classical Multi-Layer Perceptron

In the two-layered neural networks, one layer's outputs become inputs for the next one. The equations describing this operation are: $a^2 = f^2(w^2 f^1(w^1 p + b^1) + b^2)$, where:

- a^m is the output of the m-th layer of the neural network for $m = 1, 2$;
- w^m is a matrix of the weight coefficients of the each of the inputs of the m-th layer;
- b is neuron's input bias;

- f^1 is the transfer function of the 1-st layer;
- f^2 is the transfer function of the 2-nd layer.

The neuron in the first layer receives outside inputs *p*. The neurons' outputs from the last layer determine the neural network's outputs *a*. Since it belongs to the learning with teacher methods, to the algorithm are submitted training set (an input value and a target – on the network's output) $\{p_1, t_1\}$, $\{p_2, t_2\}$, ..., $\{p_Q, t_Q\}$, where $Q \in (1, ..., n)$, *n* – numbers of learning couple, where p_Q is the input value (on the network input), and t_Q is the output's value corresponding to the aim. Every network's input is preliminary established and constant, and the output have to corresponding to the aim. The difference between the input values and the aim is the error $e = t - a$. The "back propagation" algorithm [13] uses mean-quarter error: $\hat{F} = (t-a)^2 = e^2$.

In learning the neural network, the algorithm recalculates network's parameters (*W* and *b*) so to achieve mean-square error.

The "back propagation" algorithm for *i*-neuron, for $k + 1$ iteration use equations:

$$w_i^m(k+1) = w_i^m(k) - \alpha \frac{\partial \hat{F}}{\partial w_i^m}; \quad b_i^m(k+1) = b_i^m(k) - \alpha \frac{\partial \hat{F}}{\partial b_i^m},$$

where

- α - learning rate for neural network;
- $\frac{\partial \hat{F}}{\partial w_i^m}$ - relation between changes of square error and changes of the weights;
- $\frac{\partial \hat{F}}{\partial b_i^m}$ - relation between changes of square error and changes of the biases.

The overfitting [8] appears in different situations, which effect over trained parameters and make worsen output results as shown in Fig. 2.

There are different methods that can reduce the overfitting – "Early Stopping" and "Regularization". Here we will use Early Stopping [8].

When the multilayer neural network is trained, usually the available data has to be divided into three subsets. The first subset is named "Training set", is used for computing the gradient and updating the network weighs and biases. The second subset is named "Validation set". The validation error normally decreases during the initial phase of training, as does the training set error. Sometimes, when the network begins to overfit the data, the error of the validation set typically begins to rise. When the validation error increases for a specified number of iterations, the training stops, and the weights and biases at the minimum of the validation error are returned [10]. The last subset is named "test set". The sum of these three sets has to be 100% of the learning couples.

When the validation error e_V increases (the amendment de_V have positive value) the neural network learning stops when $de_V > 0$.

The classic condition for the learned network is when $e^2 < E_{\max}$, where $E_{\max}$ is the maximum square error.

For the preparing we use MATLAB and neural network structure 11:25:1 (11 inputs, 25 neurons in hidden layer and one output (Fig. 3). The numbers of the weight coefficients are $11 \times 25 = 225$. The method is focused to removing part of the number of neurons (and weigh coefficients) and dues not reduce the average deviation of the samples, used for the learning testing and validating the neural network.

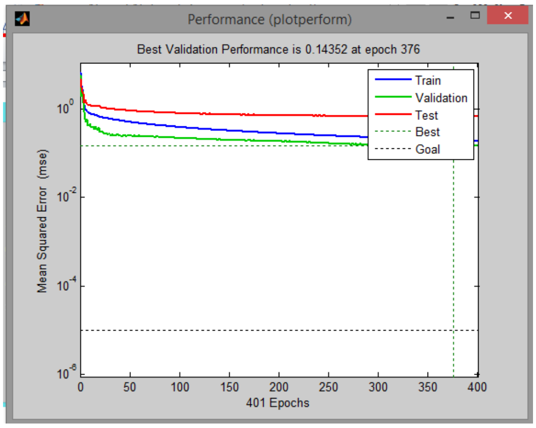

Figure 2: The learning process

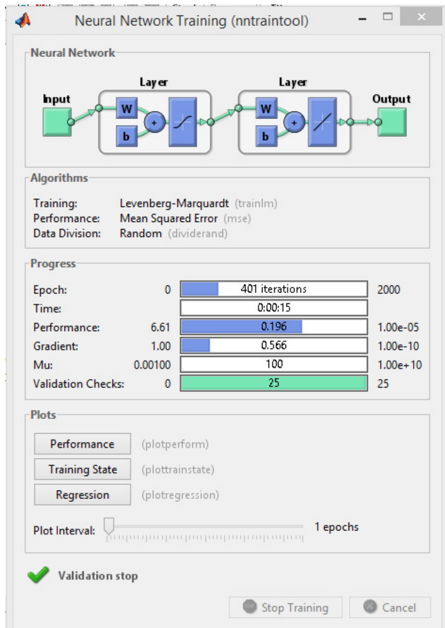

Figure 3: The neural network structure

4 Testing

At the input of the neural network we put the experimental data for obtaining type number, based on certain correlations with the rest of the criteria of biodiesel fuels measurements. We work with data from 91 biodiesel fuels, measured against 11 criteria (Fatty acid methyl esters): – palmitate (C16:0), stearate (C18:0), oleate (C18:1), linoleate (C18:2), linolenate (C18:3)– from Sigma-Aldrich. Reagents of recognized analytical grade were used.

We use the same data as input data for the InterCriteria Analysis method, applied to the whole 91 × 11 table, and a software application that implements the ICrA algorithm returns the results in the form of two index matrices in Tables 1 and 2, containing the membership and the non-membership parts of the IF correlations detected between each pair of criteria (121 pairs).

ρ	C16:0	C18:0	C18:1	C18:2	C18:3	SAT	MUNS	PUNS	FDU	CN	IN
C16:0	1.000	0.614	0.352	0.504	0.374	0.879	0.353	0.503	0.506	0.606	0.505
C18:0	0.614	1.000	0.314	0.608	0.354	0.656	0.311	0.625	0.623	0.424	0.624
C18:1	0.352	0.314	1.000	0.204	0.593	0.368	0.988	0.187	0.208	0.717	0.208
C18:2	0.504	0.608	0.204	1.000	0.287	0.470	0.201	0.938	0.854	0.206	0.853
C18:3	0.374	0.354	0.593	0.287	1.000	0.379	0.596	0.345	0.420	0.602	0.427
SAT	0.879	0.656	0.368	0.470	0.379	1.000	0.371	0.479	0.475	0.609	0.477
MUNS	0.353	0.311	0.988	0.201	0.596	0.371	1.000	0.184	0.206	0.719	0.207
PUNS	0.503	0.625	0.187	0.938	0.345	0.479	0.184	1.000	0.905	0.203	0.905
FDU	0.506	0.623	0.208	0.854	0.420	0.475	0.206	0.905	1.000	0.244	0.990
CN	0.606	0.424	0.717	0.206	0.602	0.609	0.719	0.203	0.244	1.000	0.245
IN	0.505	0.624	0.208	0.853	0.427	0.477	0.207	0.905	0.990	0.245	1.000

Table 1. Membership part of the IF pairs, giving the InterCriteria correlations

σ	C16:0	C18:0	C18:1	C18:2	C18:3	SAT	MUNS	PUNS	FDU	CN	IN
C16:0	0.000	0.333	0.621	0.470	0.577	0.093	0.619	0.471	0.469	0.372	0.473
C18:0	0.333	0.000	0.646	0.353	0.586	0.301	0.648	0.335	0.339	0.540	0.340
C18:1	0.621	0.646	0.000	0.787	0.373	0.619	0.003	0.804	0.783	0.277	0.786
C18:2	0.470	0.353	0.787	0.000	0.680	0.518	0.789	0.055	0.139	0.790	0.144
C18:3	0.577	0.586	0.373	0.680	0.000	0.585	0.370	0.622	0.548	0.369	0.544
SAT	0.093	0.301	0.619	0.518	0.585	0.000	0.616	0.509	0.514	0.383	0.515
MUNS	0.619	0.648	0.003	0.789	0.370	0.616	0.000	0.807	0.785	0.275	0.788
PUNS	0.471	0.335	0.804	0.055	0.622	0.509	0.807	0.000	0.087	0.793	0.091
FDU	0.469	0.339	0.783	0.139	0.548	0.514	0.785	0.087	0.000	0.753	0.007
CN	0.372	0.540	0.277	0.790	0.369	0.383	0.275	0.793	0.753	0.000	0.755
IN	0.473	0.340	0.786	0.144	0.544	0.515	0.788	0.091	0.007	0.755	0.000

Table 2. Non-membership part of the IF pairs, giving the InterCriteria correlations

The correlations with highest μ are shown on the section *A* on Figure 4 and listed below:

- C16:0 – SAT: ⟨0,879386; 0,0925439⟩
- C18:1 – MUNS: ⟨0,988158; 0,00263158⟩
- C18:2 – PUNS: ⟨0,937939; 0,0546053⟩
- C18:2 – FDU: ⟨0,854386; 0,138816⟩
- C18:2 – IN: ⟨0,852632; 0,14364⟩
- PUNS – FDU: ⟨0,905482; 0,0872807⟩
- PUNS – IN: ⟨0,905263; 0,0905702⟩
- FDU – IN: ⟨0,990351; 0,00657895⟩

The correlations with the highest ν are shown in section *B* in Figure 4 and are listed below:

- C18:1 – C18:2: ⟨0,204386; 0,786623⟩
- C18:1 – PUNS: ⟨0,187281; 0,803728⟩
- C18:1 –FDU: ⟨0,207895; 0,783333⟩
- C18:1 – IN: ⟨0,208114; 0,786184⟩
- C18:2 – MUNS: ⟨0,201316; 0,789254⟩
- C18:2 – CN: ⟨0,20614; 0,790132⟩
- MUNS – PUNS: ⟨0,18443; 0,806579⟩
- MUNS – FDU: ⟨0,20636; 0,784868⟩
- MUNS – IN: ⟨0,206579; 0,787719⟩
- PUNS – CN: ⟨0,203289; 0,792544⟩
- FDU – CN: ⟨0,24364; 0,753289⟩
- CN – IN: ⟨0,245395; 0,754605⟩

The other correlations are shown in section C.

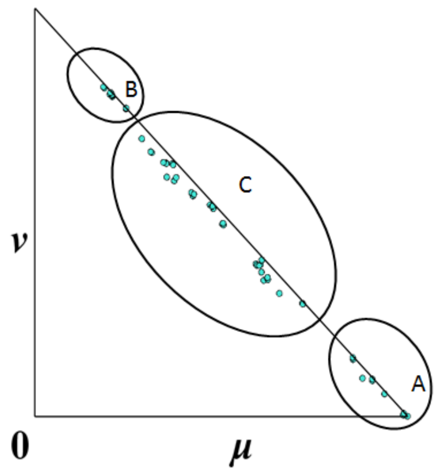

Figure 4. Intuitionistic fuzzy triangle

The objective of the preparation of the two matrices is to remove one or more columns of parameters which are repetitive (with the corresponding index of the positive and negative consonance). Testing is done as in the first step all the measurements of the 91 biodiesel fuels probes against the 11 criteria are analyzed in order to make a comparison of the obtained results thereafter. For this comparison to be possible, the predefined weight coefficients and offsets that are normally random values between –1 and 1, are now established and are the same in all studies of the various attempts.

For the learning process, we set the following parameters: Performance (MSE) = 0.00001; Validation check = 25. The input vector is divided into three different parts: Training (75/100); Validation (10/100) and Testing (15/100). For tagret we use the type number biodiesel fuels. The number of the biodiesel fuels is 6 (sunflower, rapeseed oil, palm and there mixture),

At the first step of the testing process, we use all the 11 criteria listed above, in order to train the neural network. After the training process all input values are simulated by the neural network. The average deviation of all 91 samples is 0.0488.

At the second step of the testing process, we make a fork and try independently to remove one of the columns, and experiment with data from the remaining ten columns. We compare the results in the next section 'Discussion'. First, we make a reduction of column 7 (with maximal intercriteria IF pair (0,988158; 0,00263158)) and put the data at the input of the neural network. After the training process all input values are simulated. The average deviation of all 91 samples is 0.05586.

At the third step, we alternatively experiment with the reduction of a different column, column 3 (with maximal intercriteria IF pair (0,988158; 0,00263158)), and put the data at the input of the neural network. After the training process all input values are simulated. The average deviation of all samples is 0.02578.

At the next step, we alternatively experiment with the reduction of a different column, column 4 (with maximal intercriteria IF pair (0,937939; 0,0546053)), and put the data at the input of the neural network. After the training process all input values are simulated. The average deviation of all samples is 0.03952.

Now, at this step, we proceed with providing the neural network with 9 inputs, reducing both columns 4 and 7 simultaneously; their maximal intercriteria IF pair given below. The average deviation of all 91 samples is 0.0477. At the next step, we reduce the number of inputs with one more, i.e. we put at the input of the neural network experimental data from 8 inputs, with removed columns 3, 4, and 7; their maximal intercriteria IF pair is given below. The average deviation of all 91 samples is 0.07408.

One interesting situation is when we use neural network experimental data from 8 inputs, with removed columns 3, 8, and 10, which maximal intercriteria IF pair is given below. The average deviation of all 91 samples is 0.02978.

Finally, at the this step, we experiment with providing the neural network with 7 inputs only. The reduced columns are 3, 4, 7 and 10; their maximal intercriteria IF pair are given below. The average deviation of all 91 samples is 0.04236.

5 Discussion

As we stated above, reducing the number of the input parameters of a classical neural network leads to a reduction of the weight matrices, resulting in implementation of the neural network in limited hardware and saving time and resources in training. For this purpose, we use the intuitionistic fuzzy sets-based approach of ICrA, which gives dependencies between the criteria, and thus helps us reduce the number of the input parameters, yet keeping high enough level of precision.

Table 3 below summarizes the most significant parameters of the process of testing the neural network with different numbers of inputs, gradually reducing the number in order to discover optimal results. These process parameters are the NN-specific parameters ‘Average deviation’ and ‘Number of the weight coefficients’, and the ICrA-specific parameters: maximal value for μ per column and respective value for ν, [7].

N:	Number of inputs	μ and ν	Average deviation	Number of the weight coefficients
1	11 inputs	-	0,0488	275
2	10 inputs without column 7	C18:1 - MUNS: ⟨0,988158; 0,00263158⟩	0,05586	250
3	10 inputs without column 3	C18:1 - MUNS: ⟨0,988158; 0,00263158⟩	0,02578	250
4	10 inputs without column 4	C18:2 - PUNS: ⟨0,937939; 0,0546053⟩	0,03952	250
5	10 inputs without column 8	C18:2 - PUNS: ⟨0,937939; 0,0546053⟩	0,04728	250
6	10 inputs without column 7	MUNS - PUNS: ⟨0,18443; 0,806579⟩	0,03198	250
7	10 inputs without column 10	C18:2 - CN: ⟨0,20614; 0,790132⟩	0,04248	250
8	9 inputs without column 4 and 7	C18:2 - PUNS: ⟨0,937939; 0,0546053⟩, C18:1 - MUNS: ⟨0,988158; 0,0026315⟩>	0,0477	225
9	9 inputs without column 3 and 4	C18:2 - PUNS: ⟨0,937939; 0,0546053⟩, C18:1 - MUNS: ⟨0,988158; 0,00263158⟩	0,07666	225
10	9 inputs without column 7 and 8	C18:2 - PUNS: ⟨0,937939; 0,0546053⟩, C18:1 - MUNS: ⟨0,988158; 0,00263158⟩	0,07588	225
11	9 inputs without column 3 and 8	C18:2 - PUNS: ⟨0,937939; 0,0546053⟩ C18:1 - MUNS: ⟨0,988158; 0,00263158⟩	0,05348	225
12	9 inputs without column 3 and 10	C18:2 - PUNS: ⟨0,937939; 0,0546053⟩, C18:1 - MUNS: ⟨0,988158; 0,00263158⟩ C18:2 - CN: ⟨0,20614; 0,790132⟩	0,0478	225
13	8 inputs without column 4, 7 and 10	C18:2 - PUNS: ⟨0,937939; 0,0546053⟩, C18:1 - MUNS: ⟨0,988158; 0,00263158⟩ C18:2 - CN: ⟨0,20614; 0,790132⟩	0,08106	200
14	8 inputs without column 3, 4 and 7	C18:2 - PUNS: ⟨0,937939; 0,0546053⟩, C18:1 - MUNS: ⟨0,988158; 0,00263158⟩ C18:2 - CN: ⟨0,20614; 0,790132⟩	0,07408	200
15	8 inputs without column 3, 8 and 10	C18:2 - PUNS: ⟨0,937939; 0,0546053⟩, C18:1 - MUNS: ⟨0,988158; 0,00263158⟩ C18:2 - CN: ⟨0,20614; 0,790132⟩	0,02978	200
16	7 inputs without column 3, 4, 7 and 10	C18:2 - PUNS: ⟨0,937939; 0,0546053⟩, C18:1 - MUNS: ⟨0,988158; 0,00263158⟩ C18:2 - CN: ⟨0,20614; 0,790132⟩	0,04236	175

		PUNS - CN: ⟨0,203289; 0,792544⟩		
17	7 inputs without column 3, 4, 8 and 10	C18:2 - PUNS: ⟨0,937939; 0,0546053⟩, C18:1 - MUNS: ⟨0,988158; 0,00263158⟩ C18:2 - CN: ⟨0,20614; 0,790132⟩ PUNS - CN: ⟨0,203289; 0,792544⟩	0,05632	175

Table 3. Table of comparison

The average deviation in using 11 input vectors is 0.0488 with number of weight coefficients 225. By reducing the number of the inputs, the number of weight coefficients is also decreased which theoretically is supposed to reduce the matching coefficient. In this case the removal of column 7 (and therefore one input is removed) causes further increase of average deviation to 0.05586 With maximal membership of the intercriteria IF pair (0,988158; 0,00263158) for column 3 the additional information used for training the neural network is very little, and the total Average deviation is smaller. The use of 7 columns (excluding column 8) leads to a result which is better than the previous one - 98.1609%. This shows that, while maintaining the number of weight coefficients and reducing the maximal membership in the intercriteria IF pair (0,988158; 0,00263158), the neural network receives an additional small amount of information which it uses for further learning.

Best results (average deviation = 0.02978) are obtained by removing the 3 columns with the greatest membership components of the respective IF pairs.

In this case, the effect of reducing the number of weight coefficients from 275 to 175 and the corresponding MSE is greater than the effect of the 3 columns.

The worst results (average deviation = 0.0997) are obtained in the lowest number of columns – 6 of investigation. Although the number of weight coefficients here is the smallest, the information that is used for training the neural network is less informative.

6 Conclusion

The number of the neurons is one of the major parameters during the realization of the ANN. Here we use the integration of intuitionistic fuzzy InterCriteria Analysis method for reducing the number of input parameters of the classical neural network. This leads to a reduction of the weight matrices, and thus allows implementation of the neural network in limited hardware and saving time and resources in training.

A very important aspect of the testing of the neural network after reducing some of the data (respectively the number of inputs) is to obtain an acceptable correlation between the input and output values, as well as the average deviation (or match) of the result. Here we use the data from the [17] for the biodiesel fuels probes against the 11 criteria.

Acknowledgments

The author is thankful for the support provided by the Bulgarian National Science Fund under Grant Ref. No. DFNI-I-02-5 *"InterCriteria Analysis: A New Approach to Decision Making"*.

References

[1] Atanassov K. (1991) *Generalized Nets*. World Scientific, Singapore.

[2] Atanassov K., D. Mavrov & V. Atanassova (2014) InterCriteria decision making. A new approach for multicriteria decision making, based on index matrices and intuitionistic fuzzy sets. *Issues in IFS and GN*, 11, 1–7.

[3] Atanassov K. (1983) Intuitionistic fuzzy sets, *Proc. of VII ITKR's Session*, Sofia, June (in Bulgarian).

[4] Atanassov K. (1986) Intuitionistic fuzzy sets. *Fuzzy Sets and Systems*. 20(1) 87–96.

[5] Atanassov K. (1999) *Intuitionistic Fuzzy Sets*. Springer, Heidelberg.

[6] Atanassov, K. (2012) *On Intuitionistic Fuzzy Sets Theory*. Springer, Berlin.

[7] Atanassova, V., D. Mavrov, L. Doukovska & K. Atanassov (2014) Discussion on the threshold values in the InterCriteria Decision Making approach. *Notes on Intuitionistic Fuzzy Sets*, 20(2), 94–99.

[8] Bellis, S., K. M. Razeeb, C. Saha, K. Delaney, C. O'Mathuna, A. Pounds-Cornish, G. de Souza, M. Colley, H. Hagras, G. Clarke, V. Callaghan, C. Argyropoulos, C. Karistianos, & G. Nikiforidis (2004) FPGA Implementation of Spiking Neural Networks – An Initial Step towards Building Tangible Collaborative Autonomous Agents, *Proc. of FPT'04, Int. Conf. on Field-Programmable Technology*, Brisbane, Australia, 449–452.

[9] Hagan, M. H. Demuth & M. Beale (1996) *Neural Network Design*, Boston, MA: PWS Publ.

[10] Haykin, S. (1994) *Neural Networks: A Comprehensive Foundation* NY: Macmillan.

[11] Himavathi, S., D. Anitha & A. Muthuramalingam (2007) Feedforward Neural Network Implementation in FPGA Using Layer Multiplexing for Effective Resource Utilization, *IEEE Transactions on Neural Networks*, 18(3), 880–888.

[12] Karantonis, D. M., M. R. Narayanan, M. Mathie, N. H. Lovell & B. G. Celler (2006), Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring, *IEEE Trans. Inform. Technol. Biomed.*, 10(1), 156–167.

[13] Meissner, M., M. Schmuker, & G. Schneider (2006) Optimized Particle Swarm Optimization (OPSO) and its application to artificial neural network training. *BMC Bioinformatics* 7(1), 125.

[14] Rumelhart, D., G. Hinton, & R. Williams (1986) Training representation by back-propagation errors, *Nature*, 323, 533–536.

[15] Zadeh, L. A. (1965) Fuzzy Sets. *Information and Control*, 8, 333–353.

[16] Zwe-Lee Gaing (2004) Wavelet-based neural network for power disturbance recognition and classification, *IEEE Transactions on Power Delivery*, 19(4), 1560–1568.

[17] Mustafa, Z., Surchev, S., Milina, R., & Sotirov, S. (2015). A contribution to the recognition of biodiesel fuels according to their fatty acid methyl esters profiles by the artificial neural networks. Petroleum & Coal, 57(1), 40–47.

[18] Sotirov S., V. Atanassova, E. Sotirova, V. Bureva & D. Mavrov (2015) Application of the intuitionistic fuzzy intercriteria analysis method to a neural network preprocessing procedure, 16th Congress of IFSA, 9th Conf. of EUSFLAT, Atlantis Press, 1559–1564.