

The Use of Artificial Neural Networks to Assess the Capacity of Transport Measures

Artur Duchaczek, Dariusz Skorupka

General Tadeusz Kościuszko Military Academy of Land Forces
Faculty of Management
e-mail: a.duchaczek@wso.wroc.pl, d.skorupka@wso.wroc.pl

Abstract

In the area of logistics management both managers and engineers rely primarily on proven computational algorithms, for this reason, it is often difficult to convince them to the use of artificial neural networks in solving decision problems. The paper presents the possibilities of using the FANN library in building of a computer application applied in the area of logistics. The possibilities of the component are presented on the example of applications of artificial neural networks to estimate the capacity of transport vehicles based on their dimensions. The example presented in the work was solved with the use of a multi-network Layered Perceptron. The example depicted not only the possibility of using artificial neural networks for solving poorly structured tasks but also practical application of the TFannNetwork component.

Key words: logistics, transport, artificial neural networks, computer programming

1 Introduction

The use of artificial neural networks is an alternative solution to the decision-making problems in many areas of activities taken by managers and engineers. The developed decision models with use of artificial neural networks can be multidimensional. This allows their creators to take into account a lot of input and output variables by mapping of very complex non-linear dependencies. Given not only the complexity of decision-making processes but also taking into account the need for mapping of complex mathematical functions the need to use modern tools, which are able to solve such problems, seems to be indispensable [1], [2].

The paper presents the possibility of using artificial neural networks to build a computer application used to solve one of the problems associated with the widely understood logistics management.

Logistics management includes management areas related to the movement of materials from the source to the user of finished products. It takes into account both the elements of planning, integrating and coordinating, as well as control activities in the field of logistics, so that markets are supplied in the most efficient, in terms of cost, ways [3]. It can therefore be assumed that logistics is an activity supporting all deliberate process, to the extent that the measures necessary for the implementation of the manufacturing and service tasks were available in sufficient quantity, quality and at the right time and place [4]. Transport is obviously an integral part of the logistics process, since it is the activity of the planned movement of goods and persons [5]. The concept associated with the transport is freight forwarder. It is the business organizing shipment of goods on behalf of the legal entity or natural person connected to the execution of additional activities that arise from the nature of the given order [6].

Civil engineering as a field of science lets us shape natural environment which surrounds us according to our needs. This science combines the skills of designing, constructing and maintaining buildings. Civil engineering is associated with a number of engineering specializations, including transport engineering. For this reason, transport issues are also indirectly connected with civil engineering.

Due to the wide range of activities in companies in the implementation of tasks related to transport and freight forwarding modern decision support systems may appear to be very helpful. They allow the decision-maker not only to use the data and models for recognition but also to understand and formulate the problem [7]. They also help to create solutions, enable both multi-forecasts and analysis of their effectiveness, necessary to evaluate variants and choice of the optimal variant. The purpose of decision support systems is, however, primarily the support of the reasoning process, not its elimination [7], [8]. In practice, in the case of unstructured or poorly structured tasks, it may be reasonable to use artificial neural networks in decision support systems [9].

One of the main advantages of artificial neural networks is the fact that they do not require programming. This means that the created network, by using appropriate algorithms, learns by itself automatically. The role of the application user is limited only to the design of such structure of the network that will be best to solve the given problem and then to the skilful guide of the network learning process. In a standard computer application the smallest mistake can lead to malfunction of the application, data loss and other problems. As far as the neural networks are concerned even in case of serious damage, they still work [10].

It seems that in the logistics understood as a broad field, it is possible to find a lot of problems that can be solved with the use of the so-called soft methods of analysis, including artificial neural networks and fuzzy logic [11], [12], [13].

The article presents the possibilities of using TFannNetwork component based on Fast Artificial Neural Network library (FANN) in version 2.0 for the construction of a computer application supporting decision-making processes within the activities related to the wider transport.

2 FANN Library Characteristics

Fast Artificial Neural Network library (FANN) is an open-source project that implements a multi-layer one-way neural network networks with support for both full and weakly connected networks [14]. FANN is easy to use, comprehensive, well documented and fast in acting [15], [16]. There are links to over 15 programming languages, including Delphi 7 program.

TFannNetwork is a component of Delphi (created by Pereira Maia) that connects the application with FANN library. Undoubtedly it is not necessary to install *TFannNetwork* to use FANN in Delphi but the component makes the library more friendly for Delphi environment [14].

TFannNetwork component has the ability to set (from the application level) nine basic properties thanks to which the user can rapidly create a simple application that uses artificial intelligence. These properties can be set directly on a permanent basis in *ObjectInspector* or changed at run-time of application software using the appropriate procedures (functions). These properties include, among other, the following issues [16]: selection of the activation function of the hidden layer (*ActivationFunctionHidden* option) and output layer (*ActivationFunctionHidden* option); determination of the value of the coefficient that controls the connections number within the network (*ConnectionRate* option); construction of the network structure, i.e. determining the number of hidden layers and the number of neurons in each of them (*Layers* option); determining the value of inertia ratio (*LearningMomentum* option); determining the value of the learning rate (*LearningRate* option); the choice of learning algorithm (*TrainingAlgorithm* option).

TFannNetwork component however, does not allow its users to set the value of the steepness of the activation function, hence the value of the parameter *s* (*steepness*). During the creation of a new network, its value is automatically set to 0.5. Of course, getting or setting of this parameter for the hidden layers and for the output layer can be implemented with the use of functions and procedures available from the FANN library level such as *fann_get_activation_steepness_hidden* function or *fann_set_activation_steepness_hidden* procedure [14].

Undoubtedly, an important advantage of the library discussed is the fact that the product belongs to an open-source type and yet both the library itself and the discussed component have quite professional and concise documentation [14]. According to the authors of the *TFannNetwork* component is ideal not only for testing or comparing the results obtained with other programs (eg. Statistica or Matlab) but also for teaching applications presented, for instance, in this work.

Unfortunately the FANN library and thus the discussed component currently enable only one-way network implementations [14]. In fact, from the authors previous experience results the fact that, in the majority of cases occurring in engineering practice, the above implementation is sufficient [17], [18].

3 Example of Evaluation of Wheeled Vehicles Capacity with the Use of Artificial Neural Networks

The paper presents the possibility of using artificial neural networks to evaluate the capacity of transport vehicles based on their dimensions. Organization of cargo depends on the owned fleet, especially on its capacity. In time of war or other emergency situations it may happen that the post holder (manager, etc.), responsible for organizing the evacuation of people and property, will have to quickly estimate the capacity of not only owned, but most of all obtained from the national economy, means of transport. It seems that prepared in advance decision support systems using artificial neural networks may appear to be of great help in the above mentioned emergencies.

In this work only six hypothetical wheeled vehicles, that did not have trailers or semi-trailers, were analyzed (Table 1) [19]. In the analyzed regression task the aim was to estimate the value of continuous output variable, when the input variables values are known. Regression problems are represented by sets of data, in which the output variable is not a nominal variable but is of numeric character [20].

In the analyzed case the learner vector V consisted of four arguments $V = \{L, W, H, C\}$. The first three arguments are independent (input) data, such as the length L , width W and height H of the vehicle. The fourth argument was vehicle capacity C as the dependent (output) data [18].

Both the independent and the dependent variables had to be scaled, as in case of the logistic function the neuron output value is always in the interval $(0, 1)$, whereas the range of input values, for which the function is sensitive, is slightly larger than the interval $(-1, +1)$ [20]. These values were scaled linearly according to the equation (1) [18] and are shown in Table 1, using them for learning and validation of the neural network.

$$SV = \frac{OV - Min}{Max - Min}, \quad (1)$$

where SV is the scaled value and OV is the original value of the considered dimension. In contrast, Min is the smallest value and Max is the largest value among the analyzed data set. In this task they reached respectively 0.9 and 24 value.

In an extreme situation, where we have a very small number of cases (patterns), the test group can be omitted [21]. It was therefore decided to divide the data only into learning and validation sets in a ratio of 2:1. Parameters, that described the quality of the network, were errors of prediction for cases of learners and validation groups. It was important that validation errors were significantly larger than the learning errors, since this symptom indicates weak generalization ability of the network. If the network begins overfitting, which means that it is too well adapted to the learning data, there occurs growth of validation errors while further reducing the learning error. [21]

The division of the available patterns (6 cases) into the training and validation sets took place at the beginning of the process of network learning, based on the so-called bootstrap sampling type, the procedure described in the instructions of STATISTICA *Neural Networks* program

[20]. Firstly, two cases for the validation set were drawn randomly without replacing. Then, the remaining four cases were drawn randomly with bootstrapping (i.e. with replacement). It was assumed that the learner subset had twelve cases because there was a double number of all patterns adopted. This was possible since the draw was randomized with replacement, so the subset could be arbitrarily large. In the standard procedure [20] a test subset was created from cases that remained after the "bootstrap" procedure. However, since in our case, the set was not created, these cases were automatically placed on the training set, in place of the last drawn.

Table 1: Vehicle parameters [19] used for testing and variables used in the learning and validation process

Vehicle number	Vehicle parameters				Values of used variables			
	length (m)	width (m)	height (m)	capacity (T)	L	W	H	C
1	3.6	1.8	1.7	1.5	0.117	0.039	0.035	0.026
2	3.6	1.7	1.7	0.9	0.117	0.035	0.035	0.000
3	5.2	2.3	2.3	3.0	0.186	0.061	0.061	0.091
4	6.0	2.4	2.6	2.5	0.221	0.065	0.074	0.069
5	7.8	2.4	2.4	8.0	0.299	0.065	0.065	0.307
6	13.6	2.4	2.7	24.0	0.550	0.065	0.078	1.000

The calculations presented in this paper were performed with the use of proprietary software created by the authors using in Delphi 7 TFannNetwork component (Fig. 1).

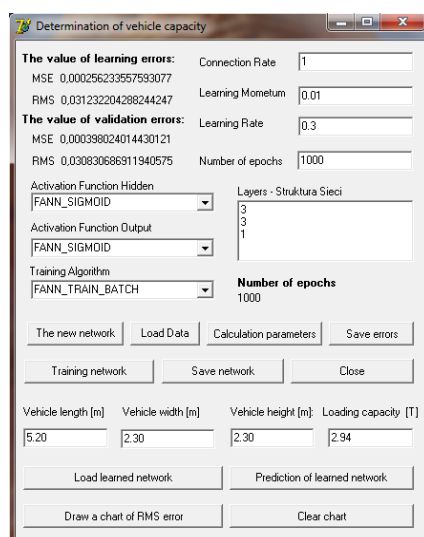


Figure 1: Window of proprietary computer application implementing network learning process and recording changes of RMS error value during learning and validation [18]

As a result of analysis, it was decided to adopt a Multi Layered Perceptron (MLP) network of the structure 3:3–3–3:1 (Fig. 2). The number of neurons in the hidden layer is limited to three in order to avoid the so-called overfitting. This phenomenon occurs during training time and relies in excessive matching of network to learning points (memorizing), which causes incorrect operation of the network for other data.

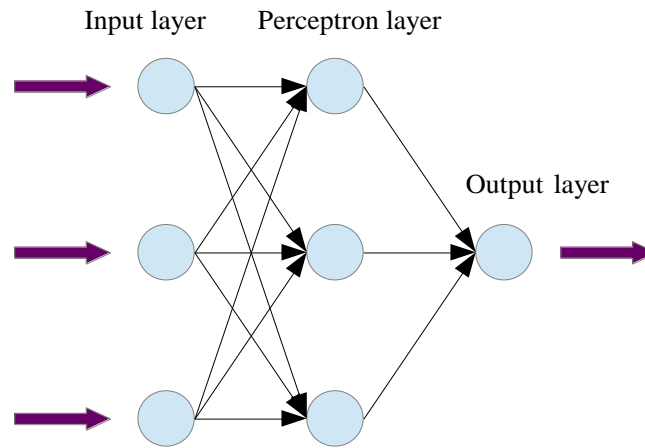


Figure 2: MLP network structure adopted for the analysis

In the calculations the maximum number of epochs equal to 1000 was assumed. At the same time to the analysis sigmoid activation function of hidden layer and output layer (*fann_sigmoid* function) and *fann_train_batch* learning algorithm were adopted. This is a standard back propagation algorithm, which updates the weights of neurons after calculating the mean square error (MSE) for all training sets (in one iteration) which means that the weights are updated only once every epoch [14]. It follows that the mean square error is calculated here with greater precision than in the standard method (*fann_train_incremental* procedure) [14]. The value of inertia ratio (*LearningMomentum* option) was 0.01, whereas the value of learning rate (*LearningRate* option) equalled 0.3. The value of the rate controlling the number of connections within the network (*ConnectionRate* option) was assumed at the level of 1.

The adopted in the modeling process network structure (architecture, activation function of hidden neurons and learning algorithm) was characterized by the lowest error value *Root Square Error* (RMS) [14]. It is calculated by summing of squared errors of the individual components, dividing that sum by the number of included n elements and the designation of the square root of the ratio calculated as:

$$RMS = \sqrt{\frac{1}{n} \sum_{i=1}^n (a_i - b_i)^2}, \quad (2)$$

where: a is the value of a learning variable, whereas b is the answer of MLP network.

Figure 3 presents the error values changes of *Root Square Error* (RMS) during learning and validation of MLP network, in successive epochs of learning with the back propagation algorithm (*fann_train_batch*). RMS error value with the progress of learning decreases and the observed convergence of error values for a validation and learning set certainly proves the correct course of the learning process.

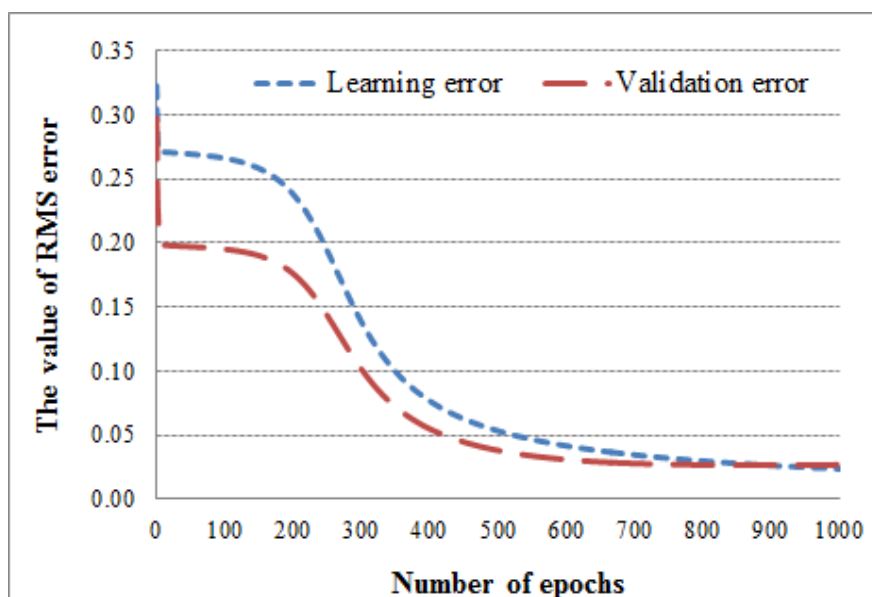


Figure 3: The course of changes in the RMS error during training and validation of MLP network in successive epochs with the learning error back-propagation algorithm

While assessing the network model, particular attention should be paid to two factors, i.e. the deviation ratio and correlation. The quotient of deviations is a measure which always takes non-negative values. In a very good model the measure reaches the value from 0 to 0.1. If the measure is greater than one, then the use of the constructed model is not justified, because a more accurate estimate of the dependent variable value is the arithmetic mean determined on the basis of the training set. The correlation is a measure that takes values between 0 and 1, wherein the higher the value the better the quality of the model [22].

The results of parameters calculations of the output variable regression in the model based on MLP network learnt with back-propagation algorithm were as follows: the average value – 6.558, the standard deviation – 7.478, the mean error – 0.092, the standard error deviation – 0.768, the deviation ratio – 0.103 and the correlation – 0.998.

The presented regression characteristics were designated for all sets at the same time (i.e. for the learner and validation set). On the basis of the variable value, there were regression characteristic specified, out of which the most important was the standard deviation ratio [23]. It is determined by the quotient of the standard deviation of the prediction error and standard deviation of the output variable [20]. This quotient is the main indicator of the quality of the constructed model, since it indicates the degree of prediction accuracy.

It seems that particularly the quotient value of deviation and correlation confirm the quality of

the created MLP model. The quotient of the standard deviations (errors and data) for the created set (the learning and validation subsets) reached the value of 0.103. The standard deviation of the data (appearing in the denominator of the ratio) is defined for the data to be analyzed both in the learning and validation process. The relatively low value of the quotient indicates a small value of standard deviation of errors (located in the numerator of the quotient) in relation to the deviation of actual data [23]. For this reason, this value can be considered as satisfactory ratio. The correlation coefficients between the actual values and those determined according to the MLP network for the created set also reached a very good value, i.e. 0.998.

4 Conclusion

Currently, artificial neural networks are a tool not widely used by engineers. Using artificial neural networks in order to evaluate the capacity of means of transport is definitely an innovative solution. Especially important for scientific reasons was the method of modifying data and choice of neural network architecture. Presented in the work FANN library was only an alternative to the use of commercial tools based on artificial neural networks. The analysis carried out on the described case enables not only presenting of the advantages of using artificial neural networks for solving poorly structured tasks but also presenting the possibility of using *TFannNetwork* component itself. The implementation of the developed and learned in the above way network, in author's decision support systems, is very simple and should not pose any major problems for those who deal with programming in Delphi only in an 'amateurish' way. Undoubtedly, that implementation in network applications created in professional programs, such as "Statistica Neural Networks" is also possible [20] but it requires from the programmer much more skills, as well as an additional license, e.g. for a product of StatSoft Poland Ltd. It seems, therefore, that the possibility of *TFannNetwork* component introduction was entirely valid. Due to its simplicity it seems that it should be widely used by engineers and managers to solve problems requiring the support of artificial intelligence and which cannot be easily solved in a logarithmic space. All tasks related to the wider civil engineering [24], [25], [26] or logistics [27], [28] undoubtedly belong to such class of problems. Knowledge of the FANN library itself can greatly facilitate the use of neural networks, but of course this library obviously still needs to be refined in many details [14]. It seems, however, that in the future may it become a fully professional product, as well as other *open-source* software products.

References

- [1] Yegnanarayana B. (2009). *Artificial neural networks*. PHI Learning Pvt. Ltd.
- [2] Patterson D. W. (1998). *Artificial neural networks: theory and applications*. Prentice Hall PTR
- [3] Pastuszak Z. (26-02-2015). *Zarządzanie logistyczne. Podstawowe definicje* [in Polish], <http://umcs.net.pl/index.php?act=Attach&type=post&id=2800>
- [4] Chaberek M. (2006). Główne problemy badawcze w zakresie rozwoju logistyki na obszarze Unii Europejskiej, *Zeszyty Naukowe Uniwersytetu Szczecińskiego, seria Ekonomiczne Problemy Usług* nr 3, vol. 435, pp. 17-22 [in Polish]

-
- [5] Marszałek S. (2001). *Ekonomika, organizacja i zarządzanie w transporcie*, Wydawnictwo Śląskiej Wyższej Szkoły Zarządzania, Katowice [in Polish]
- [6] Rydzikowski W., Wojewódzka-Król K. (2002). *Transport*, Wydawnictwo Naukowe PWN, Warszawa [in Polish]
- [7] Woźniak K. (26-02-2015). *System Wspomagania Decyzji* [in Polish], http://mfiles.pl/pl/index.php/System_wspomagania_decyzji
- [8] Aronson J., Liang T., Turban E. (2005). *Decision support systems and intelligent systems*. Yoyakarta, Andi
- [9] Delen D., Sharda, R. (2008). *Artificial neural networks in decision support systems*. In *Handbook on Decision Support Systems 1* (pp. 557-580). Springer Berlin Heidelberg.
- [10] Bartkowski B. (26-02-2015). *Sieci jednokierunkowe* [in Polish], <http://sknbo.ue.poznan.pl/neuro/ssn/pliki/czesc1.html>
- [11] Lin S. M. (2013). Analysis of service satisfaction in web auction logistics service using a combination of fruit fly optimization algorithm and general regression neural network. *Neural Computing and Applications*, vol. 22(3-4), pp. 783-791
- [12] He H. D., Lu W. Z., Xue Y. (2014). Prediction of particulate matter at street level using artificial neural networks coupling with chaotic particle swarm optimization algorithm. *Building and Environment*, vol. 78, pp. 111-117
- [13] Luo J. F., Ma T. S. (2013). An Integrated Predicting Model of New-Built Regional Logistics Center's Demand Based on the Artificial Neural Network. *Applied Mechanics and Materials*, vol. 253, pp. 1512-1517
- [14] Frankowski Ł. (01-02-2011). Fast Artificial Neural Network. Opis biblioteki FANN. Bydgoszcz 2005 [in Polish], http://www.l0co.we-bpark.pl/edu/fann/fann_ref.pdf
- [15] Nissen S. (2012). *Implementation of a fast artificial neural network library (FANN)*, 2003. Department of Computer Science University of Copenhagen (DIKU)
- [16] Nissen S. et al. (26-02-2015). *FANN - Fast Artificial Neural Network Library*. <http://leenissen.dk/fann/wp>
- [17] Skorupka D., Duchaczek A., Kamyk Z. (2011). Use of Artificial Neural Networks to Determine Class of Vehicle Load. *Journal of Science of the Gen. Tadeusz Kosciuszko Military Academy of Land Forces*, vol. 2, pp. 237-246, [In Polish]
- [18] Duchaczek A., Skorupka D. (2011). Application of Artificial Neural Networks in Decision Support Systems in Logistics Management. *Journal of Science of the Gen. Tadeusz Kosciuszko Military Academy of Land Forces*, vol. 4, pp. 270-277, [In Polish].
- [19] Hinet. (23.05.2011). *HINET - Transport i Spedycja* [in Polish]. <http://www.hinet.com.pl/rodzaje.html>
- [20] Support files of the program STATISTICA Sieci Neuronowe - version 6.0
- [21] Szaleniec M. (2008). Sieci Neuronowe i regresja wieloraka - czyli jak określić złożoność w badaniach naukowych, In: *Zastosowania statystyki i data mining w badaniach naukowych* (J. Wątroba, ed.), StatSoft, Kraków, pp.69-86 [in Polish]
- [22] Frydrychowicz W., Szymańska K. (2008). Applications of Artificial Neural Networks for Dynamical Behavior of Non-standard Market Processes. *Scientific Bulletin of Chelm Section of*

- Mathematics and Computer Science*, vol. 1, pp. 61-70, [in Polish]
- [23] Grzeszczyk T. A., (2003). *Zintegrowana metoda prognozowania w zarządzaniu przedsiębiorstwem*, StatSoft, Kraków [in Polish]
 - [24] Kasprowicz, T. (2000). Cost-time scheduling of construction works execution, *J. Infrastructure Planning and Management*, 660(IV-49).
 - [25] Kapliński, O., Janusz, L. (2006). Three phases of multifactor modelling of construction processes, *Journal of Civil Engineering and Management*, 12(2), 127-134
 - [26] Połoński, M. (2006). The analysis of the reliability of realization cost and investments, time-limits in Warsaw. *Electronic Journal of Polish Agricultural Universities Topic Civil Engineering*, 9(4), #10.
 - [27] Sobotka, A., Czarnigowska, A. (2005). Analysis of supply system models for planning construction project logistics. *Journal of Civil Engineering and Management*, 11(1), 73-82.
 - [28] Plebankiewicz E., Leśniak A. (2013). Overhead costs and profit calculation by Polish contractors. *Technological and Economic Development of Economy*, vol. 19.1, pp. 141-161.