INFORMATION
SYSTEMS IN
MANAGEMENT

Information Systems in Management (2016) Vol. 5 (2) 215–226

NEURAL MODELING OF THE ELECTRIC POWER STOCK MARKET IN USAGE OF MATLAB AND SIMULINK TOOLS FOR THE DAY AHEAD MARKET DATA

DARIUSZ RUCIŃSKI, JERZY TCHÓRZEWSKI

Department of Computer Science, Siedlce University of Natural Sciences and Humanities

The work contains selected results of the modelling of neural Electric Power Exchange (EPE) in Poland. For modelling EPE system, artificial neural network (ANN) was constructed. ANN was learned and tested using of the next day market data. Generated neural model was used for simulation tests and susceptibility tests. Suitable model was implemented in Simulink. As a result of simulation tests and susceptibility testing a lot of interesting research results were obtained.

Keywords: Artificial Neural Network, Neuronal Modelling, MATLAB and Simulink Environment, simulation research

1. Introduction

The main advantages of the design of ANN and learning the ANN model in the available computing environments is among others adaptability, self-organization, the possibility of parallel processing of information, low cost of building the system model, fault tolerance, and the like. Each ANN contains unique learned characteristics that predispose it to closely oriented categories of usage, such as to analyze the data with the ability to detect groups carrying discover approximation, prediction, optimization, pattern recognition, machine-based learning, cluster analysis etc. [1, 6-7, 10, 15] .

The basic dilemma for designers who use the SSN is a compromise between the size of the data set intended for learning and testing the network and the complexity of its architecture. On the one hand, ANN trained for too many pairs of trainees may be overtraining, which would deprive it of important properties, which is the generalization of knowledge. On the other hand, strongly developed architecture caused that the model created with great imaging of data set used in the estimation of parameters, does not work on a set of previously unused data, despite belonging to the same statistical population.

ANN unique properties allow you to get much better results in solving problems related to learning model of the system. ANN also exhibit to be a simple tool to help solve the problem of searching for regularities of development, especially in the field of data mining area. The greatest advantage is the speed of ANN due to the parallel process information and original way of functioning the very similar to the learning process occurring in living organisms.

2. Formulation of the problem

Modeled building is the Electric Power Exchange (EPE) for business processes on the Day-Ahead Market (DAM). As input data adopted 24 figures for the volume of electricity distributed and sold on EPE [kWh] in different hours of the day, and as the size of the output respectively obtained in different hours of the day, average prices for electricity [zł / kWh]. The figures refer to six months, i.e. 181 days (January 1, 2015 to June 30, 2015).

For many years the exchange of electricity in Poland is of interest not only for researchers in technical sciences, but also for those in the social and economic sciences. So far, however, it was not yet generally acceptable definition of electricity exchanges as a system. Market concept simple and understandable in practical consideration represents a major theoretical difficulties and interpretation, especially resulting from the need to strictly define a model for determining the average price of electricity only on the basis of delivered energy volume, assuming that the model implicitly included are thus different types external conditions (temperature, demand for power and electricity installed capacity, imports, exports, etc.), which led to specific practical relationship [2, 4, 5, 11, 13]. So in this way one strive to achieve model of the system as an equivalent circuit EPE in analogy to the diagrams replacement constructed using the theory of control systems. The defining of the electricity market, as DAM processes at EPE of the system is possible on the basis of control and systems theory, which can be found, among others, in the research work of W. Mielczarski [5], J. Malki [3], J. Tchórzewski [11-13] and many others.

There are also proposals for the creation of neural models, as SSN scientists using more input variables. Among other things at work *One-Hour-Ahead Load*

Forecasting Using Neural Network [8] proposed the model ANN using temperature as a factor determining energy consumption, which allow to include into the model short-term changes of demanding for the electricity energy. Another proposal was presented at work *Identification of nonparametric Dynamic Power System Equivalents with Artificial Neural Networks* [9], in which an attempt was made to find sensitive parameters, followed by reduction of neural model in order to adapt to market requirements. On the other hand, in the article entitled. *Next-Day Electricity Price Forecasting Based on Support Vector Machines and Data Mining Technology* [4] there was given a proposal for seeking the improvement of market ANN model by modifying the model using back-propagation method to harden him for atypical and boundary data.

Without compromising the objectives achieved by the construction of the above. Neuronal models we should note that they are constructed based on a macroeconomic approach, which means that there is no taken into consideration systemic nature of the electricity market and the resulting model can not be used to predict electricity prices. Meanwhile EPE, as the system has a number of mechanisms whose understanding requires a simulation tests and susceptibility testing in order to determine the nature of the system EPE. For obvious reasons, carry out any experiments on real system is too risky and harmful. For these reasons, it is worthy to build models of EPE as a substitute schemes and thus have a sufficient and necessary workshop to conduct all types of experimental research.

2.1. ANN to implement the system model EPE

To implement the EPE model of multi-layered ANN was chosen, it was trained by back propagation of error algorithm in MATLAB and Simulink environment by using Neural Network Toolbox (NNT). The structure and writing conventions is as in Fig. 1.

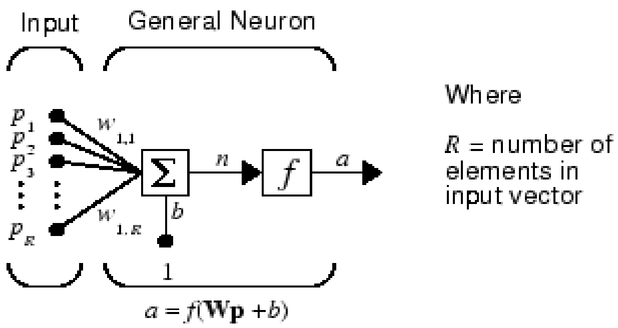

Figure 1. The neuron model in the convention used in MATLAB and Simulink. Symbols: p_R – $_R$- entrance to the ANN, R - number of input signals, , w_{ij} - the weight vector component j is connected to neuron i, b - bias, n - total signal value of the neuron (ie. net), f() - function of neuron activation, a - value of the output neuron signal (usually denoted as y). *Source*: [16]

There are many different types of artificial neural networks depending on the adopted architecture, methods of learning or activation function [4, 6-10, 15-16]. In the example the functioning of the EPE for DAM such a network-oriented learning system model can be a one-way single-layer ANN, trained by back-propagation of logsig activation function () (Fig. 2).

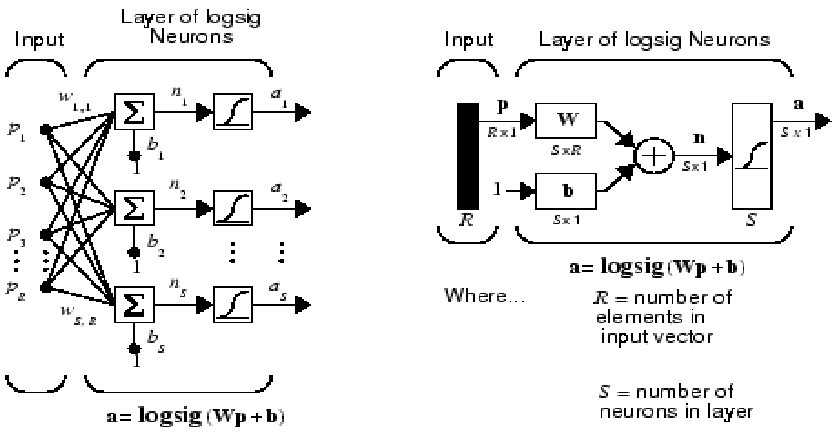

Figure 2. Architecture of single-layer artificial neural network, learning by backward propagation, the activation function logsig (on the left side of architecture ANN trained neural model, EPE, right - flowchart ANN). Indications as to Figure 1. *Source*: [16]

Multi-layered one-way artificial neural networks have one or more hidden layers. If the ANN has a non-linear neural activation function then it can be learned non-linear relation between the input vector and the output. Sometimes these types of issues it is convenient to apply the last layer of ANN linear function activation to achieve the model fit to the actual system.

ANN used to implement the EPE system model is based on the activation functions tansing (in the hidden layer) and purelin (the output layer) as shown in Fig. 3.

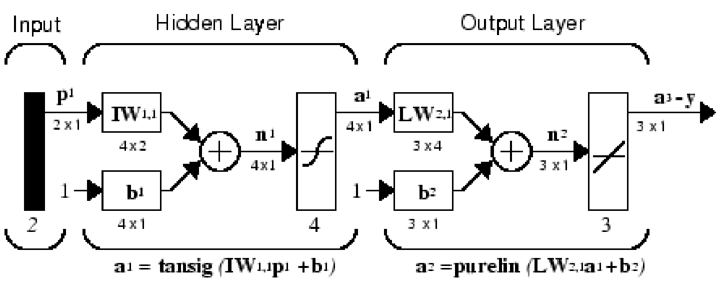

Figure 3. Refer to Fig. 1. *Source*: MATLAB and Simulink [16]

2.2. ANN for implementation of system models

The process of preparing and training the ANN model of the system involves the need to pre-prepare the training data, which is carried out increasingly different pre-treatments, such as normalization of data. This process preceding the design of ANN called cluster analysis requires knowledge of the issues enough to be able to determine the scope of modeling. The aim is to achieve high-quality neural model, which depends not only on the quality of the input and output, but also on properly conducted analysis of data to be stored in a file learners and the training file.

The selection of inputs, and the architectural elements of the model is the subject of many studies that include various proposals and example. In [1] proposed adding to the size of the model inputs of neural temperature for the day, because it is one of the factors determining the energy consumption that is directly Translating short-term changes in demand for power and electricity.

Another proposal was presented in the article [8], which proposed to improve the way search parameters in order to reduce the ANN model in the case when, for distributed electricity market, there is a lack of reliable data. Improving neural model parameters is possible using different methods and instance. In [9] emphasized the possibility of improving the system model by modifying the method of back propagation and replacing it by a more resistant to the critical values. It is important, therefore, that the scope of data used for training the ANN model of the system EPE covered all the essential characteristics of the model.

So before start working with the data they should be adequately prepare for example by using the cluster analysis. At present time, the most common data preparation before learning is normalizing, which is to align the data within the scope processed by ANN. There are many methods of standardization of data used for training the ANN model of the system. The cluster analysis are sufficiently describes, inter alia in the monograph by M. Kłopotek and S. Wierzchonia entitled *Custer Analysis* [14].

2.3. ANN design for learning the EPE model

After appropriate preparation of the test and learner data file we should proceed to its design. In MATLAB and Simulink library using the NNT first step is declaration type of SSN, eg. Using functions feedforwardnet (hiddenSizes, trainFcn). The function has two optional arguments: hiddenSizes (number of neurons in the hidden layers) and trainFcn (learning function). An important stage is the selection of the activation function of the individual layers, which in the present case were adopted respectively for the first and second layers as "tansing" and "purelin". Further configuration property SSN is possible using the configure function, in which can be specified as an argument input vector, output vector, learnersvector. For example, Net = configure (net 'outputs', t, i). The "configure" also initializes the weight and displacement called "bias".

It is also possible to reinitialize the weights by using net = init (net). And finally the last command used to initialize learning ANN function is train as a [net tr] = train (net, X, T, X, A, and, EW), in which the individual arguments are respectively as follows: net is the name of the network, and X is input matrix network, this is the desired network (teacher), Xi is ANN initial delay input, Ai is the initial delay layer, ΔW is severity of the errors. As a result of the above operation function, object is returned as net SSN, including matrices and vectors weight values biases.

MATLAB and Simulink environment enables implementation of various types of networks, single and multi directional, single and multi layer. For example, Cascadeforwardnet which forms the unidirectional cascade, the net-like feedforwardnet. The difference is input to the combined weight of each layer of the network and the weight of each subsequent layer to layers of the following. This architecture sometimes improves the learning speed network. Another useful function was a patternnet function which is similar to feedforwardnet except that the last network layer performs the function of transfer 'tansing' instead of purelin'.

2.4. ANN learning EPE model

ANN can be learned EPE model, which is the issue of determining the type of function approximating (linear regression). In this paper the learning process of ANN with a teacher who needs two sets of data, input data and reference data set (so called file learner of the trainees' pairs).

ANN learning process depends on such adjustment weights to the file of the learner to input data in a way that is closest to the exit coincided with reference data.

Parameter model is the net, which consists of a weighted summation process input variables defined in Matlab and Simulink using net.performFcn () - Table 1. The default function of executing the process in the adopted artificial neural network (like feedforwardnet) is Mean Square Error (MSE).

Table 1. "*perf*" *function paramers = performance(net,t,y,ew.*)

net	Network name
t	The values of the desired network
y	The output values of the network
ew	weight of the errors (optional)

Source: [16]

This function returns a value of errors. It has two optional parameters influencing the process of training:

- Regularization, which may have values between 0 and 1, where a higher value indicates that more weight is taken into account when calculating the value perf in relation to the error, and the default value of 0 indicates no impact on the learning process.

- Normalization, which can take the value of none (default), the standard (normalization error of [−2, 2], and percent (normalization error of [−1, 1]).

There are two ways to implement the learning process of ANN: incremental method and a batch. Incremental method calculates the gradient of weight and updates its value after processing of each learning pair. In the batch method entire files are processed and after processing the weights are updated. It is believed that this first method is more accurate, and the other for faster learning ANN model of the system.

For the purpose of training the ANN model, EPE trainlm function is selected, which is an implementation of the algorithm Levenberg-Marquardt due to, among others, its effectiveness, ie. Long convergence point optimization, and the fact that given the input vectors to approx. 1000 pieces shown a moderate demand for memory operations.

Characteristics of the algorithm is approximate character of calculations, i.e. it is designed to be able to estimate the direction and value of change into the matrix weights on the basis of the information contained in the gradient error objective function (understood as the difference between the reference value and the value of output) and arbitrarily to modular regulatory factor.

3. The research experiments and obtained results

ANN learning system model has been used EPE, EPE figures for the DAM of the period from January 1, 2015 to June 30, 2015. Adopted 24 input variables representing the volume of electricity for each hour of the day and 24 representing the size of the output electricity prices in the different hours of the day.

As a method for network learning they use the "trainlm", which is an implementation of the algorithm Levenberg-Marquardt, the first layer activation function is the function sigmodalna "tansing" and the other a linear function "PURELINE". ANN architecture was built in MATLAB and Simulink using the NNT (Fig. 4).

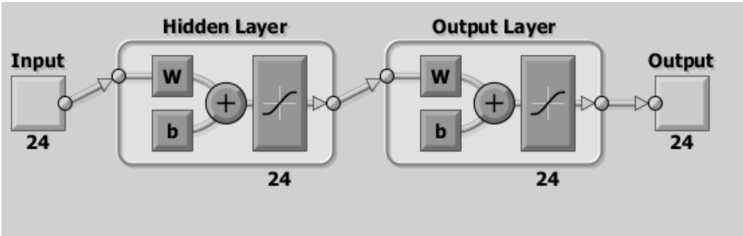

Figure 4. ANN network model for DAM. Symbols: Input is the vector of the input (volume of electric power and energy [kWh]), Hidden Layer is the hidden layer, Output layer is a layer Output, Output is the vector output value (price)
(MATLAB and Simulink calculations)

3.1. Learning function ANN of EPE model

As a result of learning obtained PPS system model, EPE, which in the fourth epoch received optimal network performance, and therefore can be considered as "sufficiently trained", i.e. the next era of learning does not improve its quality. In the generated model, EPE has also been examined, among others, the impact of the number of hidden layer neurons on the quality and speed of network learning (Fig. 6).

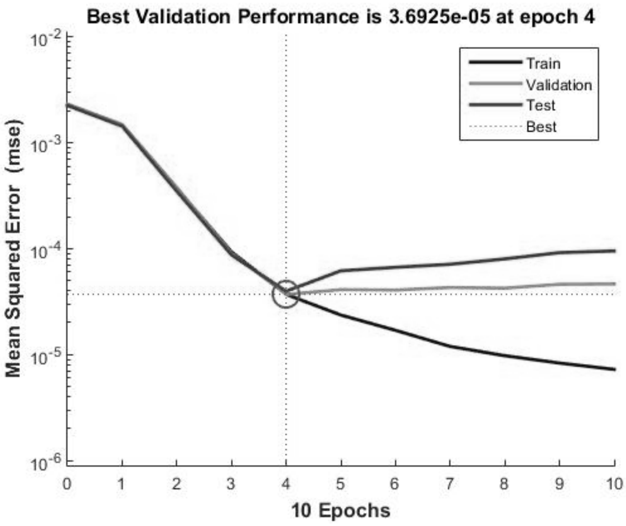

Figure 5. The course of training the ANN model of the system of EPE
(MATLAB and Simulink calculations)

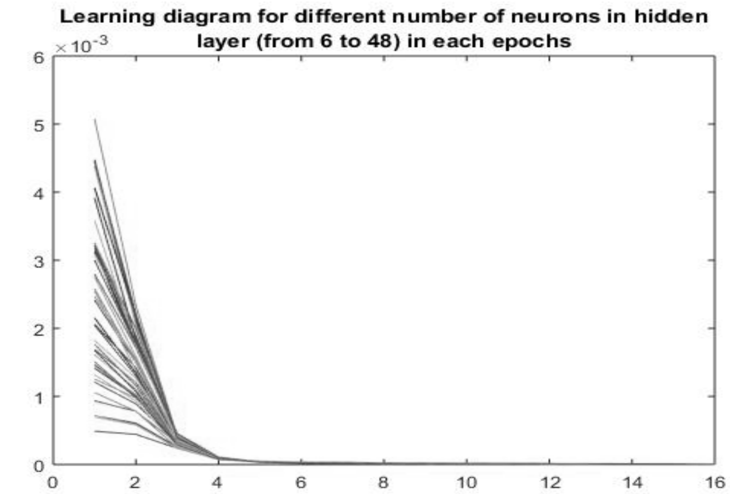

Figure 6. Trajectory of error MSE depending on the number of neurons in the hidden layer (MATLAB and Simulink calculations)

As can be seen training the ANN model of the system EPE proceeded relatively quickly regardless of the number of neurons in the hidden layer. In any case, since the fourth epoch weight changes were small and the number of neurons in the hidden layer have small effect on the rate and quality of learning.

3.2. The sensitivity test of the neutral model EPE depending of the number of neurons in the hidden layer

EPE neuronal model was also examined from the point of view of the number of hidden layer neurons to quality "fit" model to the training data (ie. Regression). The model reproduces well studied problem, if the value of regression is close to one - Fig. 7.

According to Fig. 7 it shows for example, that the test pattern has a high stability in the tested range of 6 to 48 neurons in the hidden layer, and the regression is formed in the range of 0.75 to 0.85. You may also notice there's a single "peak" for a specific number of neurons in the hidden layer. It was therefore further attempt to investigate the above. Therefore there was another attempt to train the network with the same input condition and examine the regression.

The course of the regression coefficient (R) depending on the number of hidden layer neurons (Fig. 8) shows among other things that the average value of the regression as well as for the case of Fig. 7 is formed in the range between 0.75 and 0.85, but "peaks" appear for a different number of neurons in the hidden layer than in the case of Fig. 7. The research experiment shows that it is caused by, among others, random weight initialization method in the learning process of ANN model of the system of EPE.

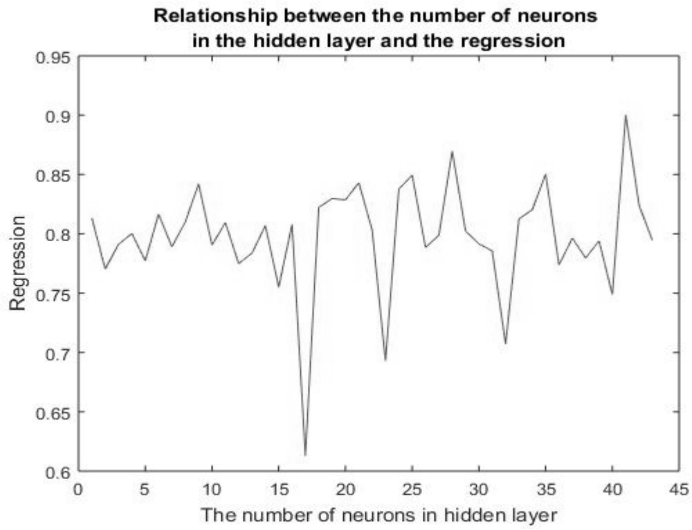

Figure 7. Trajectory of the regression, depending on the number of neurons in the hidden layer (MATLAB and Simulink calculations)

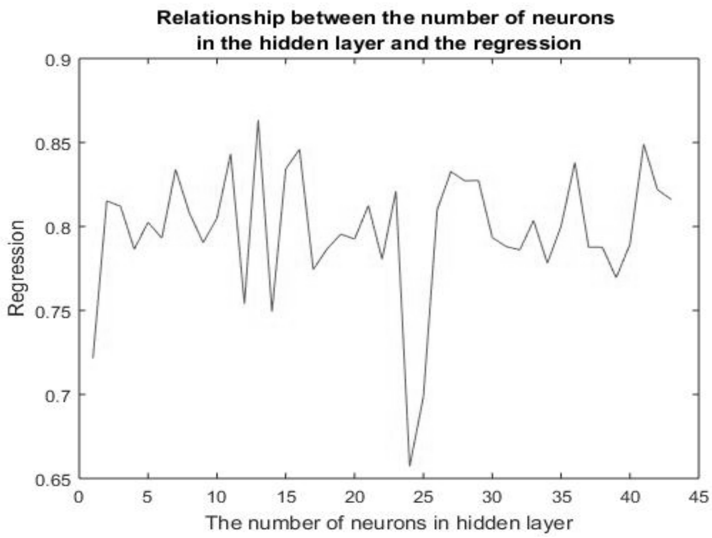

Figure 8. Trajectory of the regression, depending on the number of neurons in the hidden layer (MATLAB and Simulink calculations)

4. Conclusions and directions for further research

There is a need and possibilities for system modeling Electric Power Exchange in order to get the system model, EPE for DAM quotations. One of the method of modeling leading to a model of the system is EPE neuronal model. In order to obtain a model of neural system EPE figures collected for each hour of the day posted for trading DAM, subjected to normalization and used in the learning process of ANN model of the system of EPE.

The resulting model of neuronal system, EPE is possible to use in further studies, especially in the conduct simulation studies on possible directions of development and susceptibility testing EPE, EPE development for unusual situations that may arise in crisis situations. The method of modeling can also be used to construct a model of neural safe control of the development of the electricity market in Poland.

REFERENCES

[1] Fijorek K., Mróz K., Niedziela K., Fijorek D. (2010), *Prognozowanie cen energii elektrycznej na Rynku Dnia Następnego metodami data minig*. Rynek Energii nr 12.

[2] Mynarski S. (1993), *Analiza rynku. Systemy i mechanizmy.* Wyd. AE. Kraków.

[3] Malko J. (2003), *Black-out, czyli zdarzenia katastrofalne w krytycznych systemach infrastrukturalnych,* Wokół Energetyki vol.6 tom 5.

[4] Li J., Li J. (2008) *Next-Day Electricity Price Forecasting Based on Support Vector Machines and Data Mining Technology.* Proceedings of the 27th Chinese Control Conference, Kunming, Yunnan, China.

[5] Mielczarski W. (2000), *Rynki energii elektrycznej. Wybrane aspekty techniczne i ekonomiczne,* ARE, Warszawa.

[6] Osowski S. (2000), *Sieci neuronowe do przetwarzania informacji,* PW, Warszawa.

[7] Rutkowska D., Piliński M., Rutkowski L. (1997), *Sieci neuronowe, algorytmy genetyczne i systemy rozmyte,* PWN, Warszawa.

[8] Senjyu T., Takara H., Uezato K., and Funabashi T. (2002) *One-Hour-Ahead Load Forecasting Using Neural Network,* IEEE Trans. on Power Systems, vol. 17, no. 1, pp. 113-118.

[9] Stankovic A. M., Saric A. T., and Milosevic M., (2003) *Identification of Nonparametric Dynamic Power System Equivalents with Artificial Neural Networks,* IEEE Transactions on Power Systems, vol. 18, no. 4, pp. 1478-1486.

[10] Tadeusiewicz R. (1993), *Sieci neuronowe.* AOW RM, Warszawa.

[11] Tchórzewski J. (2013) *Rozwój systemu elektroenergetycznego w ujęciu teorii sterowania i systemów.* OW PWr. Wrocław.

[12] Tchórzewski J. (2009), *Identification of the Electrical Energy Stock Exchange and creating knowledge maps using MATLAB environment with SIT and NNT Toolboxes, Energy Market*, 2009. EEM 2009. 6th International Conference on the European. IEEE Xplore Digital Library.

[13] Tchórzewski J. Ruciński D. (2016), *Neural Modeling Safe Development of the Electricity Stock Exchange Using Quotations of Day-Ahead Market,* Wolters Kluwer, Warszawa.

[14] Wierzchoń S., Kłopotek M. (2015), *Cluster analysis*. IPI PAN. Warszawa.

[15] Żurada J., Barski M., Jędruch W. (1996), *Sztuczne sieci neuronowe. Podstawy teorii i zastosowania*, PWN, Warszawa.

[16] http://www.mathworks.com/help/nnet