ARCHIVES OF ENVIRONMENTAL PROTECTION
ARCHIWUM OCHRONY ŚRODOWISKA
vol. 31 no. 2 pp. 15 - 26 2005

PL ISSN 0324-8461

ARTIFICIAL NEURAL NETWORKS IN AIR POLLUTION PREDICTION – IMPORTANCE OF INPUT VARIABLES

WIOLETTA ROGULA[1], JACEK ŻELIŃSKI[2]

[1] Instytut Podstaw Inżynierii Środowiska Polskiej Akademii Nauk,
ul. M. Skłodowskiej-Curie 34, 41-819 Zabrze
[2] Politechnika Śląska, Wydział Inżynierii Środowiska i Energetyki, Katedra Ochrony Powietrza
ul. Akademicka 2, 44-100 Gliwice

Keyword: multilayer perceptron, neural network, backpropagation, artificial intelligence, air pollutants, prediction, input variables, output variables, meteorological conditions.

SZTUCZNE SIECI NEURONOWE W PROGNOZOWANIU ZANIECZYSZCZENIA POWIETRZA – ISTOTNOŚĆ ZMIENNYCH WEJŚCIOWYCH

W prezentowanej pracy badano istotność doboru zmiennych wejściowych (mechanizmów i czynników meteorologicznych) w predykcji stężeń zanieczyszczeń powietrza za pomocą sztucznych sieci neuronowych. Posłużono się danymi pomiarowymi ze stacji monitoringu powietrza w Gliwicach. Do analizy danych zastosowano program Statistica Neural Networks firmy StatSoft. Podczas tworzenia sieci neuronowych, dla wszystkich zmiennych wyjściowych (stężeń kolejnych zanieczyszczeń), przetestowano ponad 3500 modeli neuronowych. Przy pomocy najlepszych modeli określono oddziaływanie danego parametru na poziom stężenia zanieczyszczenia (Analiza Wrażliwości Sieci). Na podstawie wykonanych analiz wyciągnięto wnioski, co do wagi konkretnych parametrów meteorologicznych.

Summary

The essentiality of variables in Artificial Neural Networks (ANN) application in predicting concentrations of pollutants in the ambient air is considered in the paper. Evaluation of the essentiality was based on the data on concentrations of pollutants and meteorological conditions recorded by an automatic station monitoring the air quality in Gliwice. The data were analysed with the use of the StatSoft's Statistica Neural Networks (SNN) software, which is designed to simulate performance of artificial neural networks. In total, for all output variables (concentrations of SO_2, NO, NO_2, PM_{10}), more than 3500 models were tested to create the final neural networks. The best performing models were used to determine the influence of each input variable on levels of pollutant concentrations. Based on these analyses the conclusions were drawn concerning the importance of individual meteorological parameters.

INTRODUCTION

Proper selection of variables in computational modelling is important, especially for accuracy of computations. Thus, the evaluation of the importance of any particular variable, as well in the models applying "usual", sequential, computations to simulate physical

phenomena as in applications of various types of the parallel processing of information (e.g. Artificial Neural Networks, ANN, as well classifying as predicting), is also important. In ANN applications, the importance of a variable may be determined by using various methods and measures [2–4] – generally to know the effect of the variable on the output and eventually to discard the variable if this effect is small.

In the widely applied models of propagation of pollutants in the atmospheric air, concentrations of a substance in the air depend on meteorological conditions referred to explicitly in the computations (e.g. wind speed), or affecting the results as hidden factors comprised in aggregates of some number of such factors (diffusion coefficient, meteorological exponent) [8]. The essentiality of these variables to computation of the concentrations is difficult to asses due to their entanglement in the model and empirical character or inadequacy of mathematical formulas used.

The Artificial Neural Networks have been proved to be applicable in predicting the pollutant concentrations [5–7, 9–11]. The ANN called Multilayer Perceptrons (MLP) appeared to be neural networks that basing on meteorological data they work very well perform in predictions of air pollution.

This paper presents an attempt to use the data from an automatic station monitoring the ambient air quality to generate MLPs capable of making predictions of concentrations of air pollutants by using meteorological data. By analysing ranks of particular input variables the influence of these variables on the network output – i.e. concentration – was determined. Using a very large data set, i.e. a great number of cases analysed during the network training, allowed independent evaluation of the effect of the meteorological conditions from the domain defined by domains of particular, used during the training, variables on concentrations of pollutants.

PRINCIPLE OF NEURON FUNCTIONING AND ANN TRAINING – GOALS FOR THE ANALYSIS

The scheme of a neuron is presented in Fig. 1.

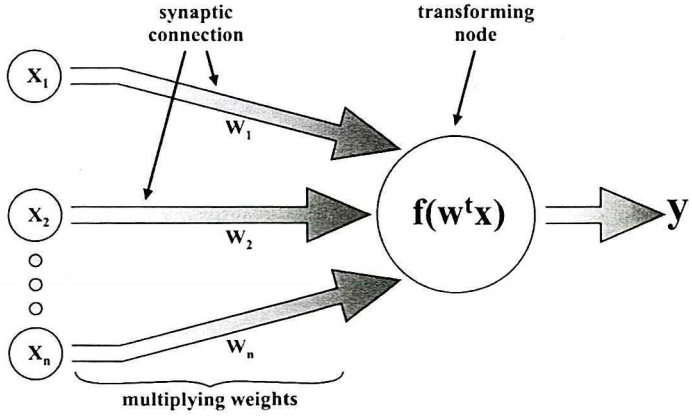

Fig. 1. General scheme of a neuron

The neuron consists of a transforming node with some number of synaptic inputs and one output. The input i is assigned a weight w_i. All inputs and output are one-way signal transducers: inputs – to, output – from the node. The output signal of the neuron is defined by the equation:

$$y = f(w^t x) = f(\sum_{i=1}^{n} w_i x_i),$$

where:

$w \overset{def}{=} [w_1, w_2, \ldots w_n]^t$ – vector of weights (vector of weighted synaptic connections)

$x \overset{def}{=} [x_1, x_2, \ldots x_n]^t$ – vector of input signals,

f – transfer (activation) function.

The scalar product of vectors w and x:

$$net = w^t x = \sum_{i=1}^{n} w_i x_i,$$

is called the weighted sum of the inputs and f relates *net* with the output:

$$y = f(net).$$

Sometimes, the weighted sum of inputs *net* is modified by adding a constant w_0:

$$e = net + w_0 = w^t x + w_0 = \sum_{i=1}^{n} w_i x_i + w_0,$$

and then the neuron emits a value equal to $y = f(e)$.

The transfer function f relates e with the output. The function f should be nonlinear – if it was not, the perceptron would simulate only linear functions. For purposes of this work the sigmoid function has been chosen as the transfer function f [16]:

$$y = f(e) = \frac{1}{1 + \exp(-\beta\ e)},$$

where β is a constant.

The idea of putting all above concepts together is as follows. The signals x_i on inputs of a neuron are multiplied by weights w_i, and summed up, the sum is eventually augmented by w_0. The result is then transformed with the use of f and sent out from the neuron through the neuron output. Because the weights w_i are different the input signals are not equivalent – some are of lesser some of greater importance, contributing to the output signal more or less.

Amount of information accumulated in the neuron, and consequently in the whole network, is contained in values of the connecting weights. Therefore the method for selecting these values is crucial to the network ability to simulate the objective relationship. The

connecting weights are determined in the process of the network training. The phase of training a network consists in repeatedly presenting the network with the patterns from a set of training data and adjusting the weights until the response is satisfactory. During the training the network acquires ability to accurately generalise data not included into the training data set.

There are two basic methods for network training: supervised and unsupervised learning. The Mutilayer Perceptrons are the neural networks being applied to prediction of concentrations of air pollutants and they are trained in the supervised manner [16,17].

In the supervised training the network input is supplied with a vector x from the training data set and the desired output d. The network answers with the output o. The actual output o is compared to d to calculate the error $\rho[o,d]$, and next $\rho[o,d]$ is used to alter the weights of the network to lower the network error. The procedure is repeated until the network error is below an assumed value. The process of the supervised training is presented in Fig. 2.

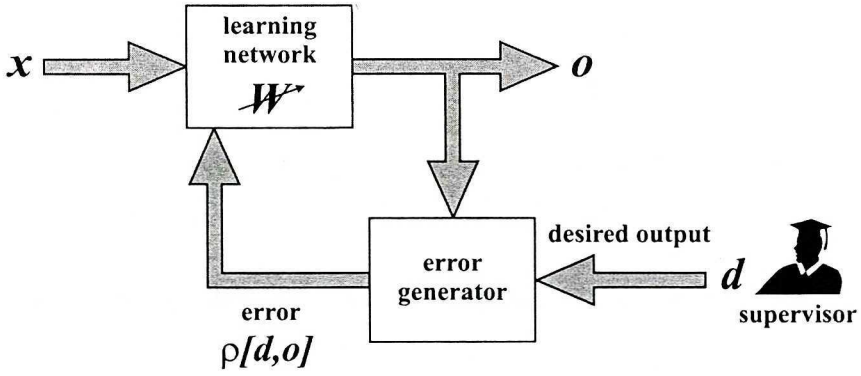

Fig. 2. Scheme of supervised training [17]

It may be said that during the process of supervised learning the network modifies its inner parameters until its answers to inputs from the training set are satisfactorily close to proper desired outputs, i.e. generated errors are below some *a priori* assumed level .

While changing its weights the network differentiates importance of the particular inputs. But a signal on the neuron output – its magnitude, whether it appears or not – still depends not so much on a signal in a particular dendrite as rather on a whole configuration of signals fed to the neuron. Some configurations of neuron inputs activate a neuron, some not.

GENERATING NEURAL MODELS (TRAINING THE NETWORK)

The data recorded by the automatic station of air quality monitoring in Gliwice were used in the presented work. The station measures 30-minute concentrations of SO_2, NO, NO_2, CO, PM_{10}.

The set of data consisted of records each of which comprised direction and speed of wind, air temperature, solar radiation, relative air humidity, number of Pasquil's stability

category of atmosphere [8], and concentrations of SO_2, NO, NO_2, CO, PM_{10}. The set comprised about 1400 records with these data [12]. The set of data used to train the network for a particular pollutant consisted of these records but with only one concentration – the concentration of just modelled pollutant, the rest of the concentrations from the record were not taken under consideration.

The data were verified to fit them formally for the circumstances they were to be used in, and modified due to specificity of the network performance – e.g. numbers of the atmosphere categories were replaced by the nominal symbols to avoid assigning the weights depending on magnitude of the class numbers to the classes.

The StatSoft's Statistica Neural Networks (SNN) computer program, simulating neural networks, was used during the whole experiment to deal with neural networks [1, 13–17].

It was assumed that the pollutant concentrations were independent and mutually related only by accident. A set of the independent variables (meteorological parameters) was chosen to which the dependent variables (concentrations of pollutants) were to be assigned. The next step was starting the process of creation of neural models able to predict concentrations of the pollutants by using the meteorological parameters.

The set of data for each particular pollutant was divided into three subsets: training, validating and testing set of data, the proportion of their cardinalities was 2 : 1 : 1, respectively. The whole process of training, using the backpropagation training algorithm, was applied to each of the networks [16,17].

The following three step process was recognised as the best method for seeking the optimal network:

1. Selecting a type of the network: from among all networks projected and trained by Automatic Network Designer (AND) to perform the task, a type was chosen that produced the smallest prediction error. After loading arrays of data into the program, all kinds of neural networks provided by the program – the Linear Neural Networks (LNN), Probabilistic Neural Networks (PNN), General Regression Neural Networks (GRNN), Radial Basis Function Networks (RBF), one way Multilayer Networks (MultiLayer Perceptrons, MLP) – were tested. From among all the tested networks, MLP rendered the smallest errors, so the following stages of searching for the best model were narrowed to MLPs of various architecture.
2. Selecting network architecture: at this stage, among MLPs, the model was looked for with the architecture providing the best conformation of the network to performing the prediction task. From among networks of the chosen type the network with the concrete construction was chosen.
3. Changing properties of the network having the concrete architecture: at the latest stage the best one from among the tried networks was edited. The changes consisted in decreasing or increasing a number of neurons in the hidden layer of the perceptron and in recurring several times changes of the transfer function. Each alteration was followed by the network training and the new network error was compared with the original network error.

In total, about 3600 neural networks differing with their architecture were tested. The best ones to perform the task – prediction of concentrations based on meteorological parameters – appeared to be the multilayer perceptrons (MLP), what is consistent with the literature [5–7, 9, 10]. The scheme of such a network is presented in Fig. 3.

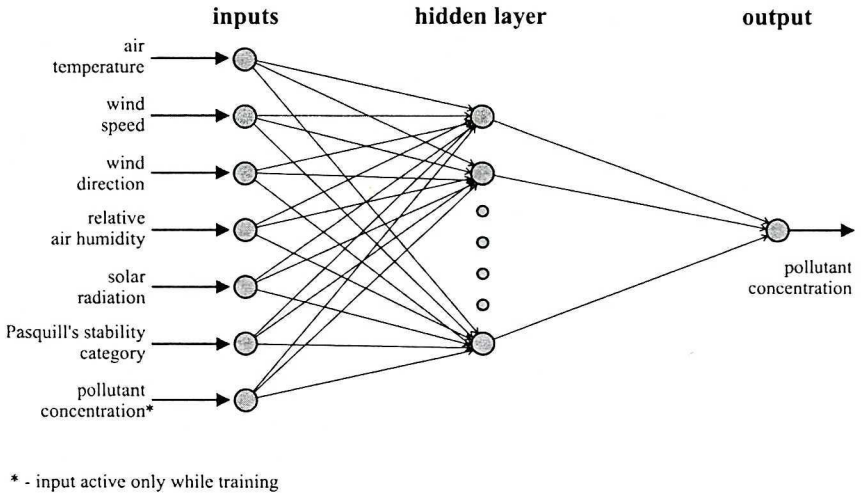

Fig. 3. Architecture of the three-layer feed-forward neural network applied in the study

The main properties to classify the neural networks are:

Type: mentioned above – LNN, PNN, GRNN, RBF or MLP. Only MLP appeared to be of interest. So the networks of other types are merely mentioned.

Structure of a network: architecture of a network is described with a network configuration code. The code consists of three sections *a*, *b*, *c* separated by colons: *a*:*b*:*c*. The sections *a* and *c* are numbers: *a* is the number of the network input variables, *c* – the number of the network output variables. The section *b* informs of the number of internal layers of the network: it consists of a sequence of numbers separated by hyphens, the k-th number is the number of neurons in the k-th layer. For instance, 4:5-6-3:1 denotes a network with four input variables, one output variable and three layers comprising 5 input neurons, 6 hidden neurons and 3 output neurons, respectively.

Regression coefficient: or proportion of standard deviations. Its value greater than or equal to 1 (dimensionless) characterizes a model yielding not better results than the model outputting always the same prognosed signal being the mean of the earlier observed output values. Value less than 1 is for a model with the better output fitting – the lower regression coefficient, the better the model.

Correlation: the coefficient (dimensionless number) expressing relation between values on input and output of a network; it varies in the interval 0 – 1; the closer the correlation to 1, the higher quality of the received network.

Error: the network error is the square root of the sum of squares of errors of particular cases determined by the network error function. It is the dimensionless quantity.

In Table 1 characteristics of selected neural models are presented.

Table 1. Selected neural networks [12]

Output variable	Type and structure of network	Regression coefficient	Correlation	Error
SO_2	LNN - 6:40-1:1	0.69	0.72	22.82
NO	MLP - 6:40-6-1:1	0.80	0.60	13.86
NO_2	MLP - 6:40-12-4-1:1	0.91	0.75	8.82
CO	MLP - 6:40-2-1:1	0.79	0.63	0.39
PM_{10}	MLP - 6:40-3-1:1	0.62	0.79	25.46

As the criterion, the magnitude of the error was assumed for selecting the network [12, 16, 17]. In the case of close network errors, the selection of the network depended on the correlation.

SENSITIVITY ANALYSIS OF THE CONSTRUCTED NEURAL MODELS – EVALUATION OF THE ESSENTIALITY OF METEOROLOGICAL PARAMETERS TO PREDICTION OF CONCENTRATIONS OF AIR POLLUTANTS

The network output after the network training shows different sensitivity to changes made to the different input variables. The sensitivity analysis allows for gaining insight into usefulness of particular input variables. It points out the variables that without loss of the network quality may be neglected as well as the variables that are crucial to the problem and should never be ignored. The selection of the essential variables consists in determination of the contribution of each variable to performance of a neural network and rejecting the least important ones. The importance of the variable may be determined by applying various techniques [3, 4], including the methods with the use of the neural networks themselves [2].

The sensitivity analysis reveals the damage caused by rejecting the particular variable to the whole system.

Sensitivity of the network was defined by:

- Error – the overall error of the network received by rejecting the variable; the greater value of the error, the greater variable significance [12–15];
- Quotient – the proportion of the error to the overall error of the complete network with all its variables (rejecting the variable with the quotient less than 1 should yield an improved neural network) [12–15].

Results of the sensitivity analysis of the networks received for particular pollutants are presented in Table 2.

Table 2. Results of sensitivity analysis for particular pollutants

	Wind direction	Wind speed	Temperature	Humidity	Solar radiation	Pasquil's stability category
LINEAR 6:40-1:1 for SO_2						
ERROR	32.07	31.57	51.16	31.45	31.48	32.89
QUOTIENT	1.02	1.00	1.62	1.00	1.00	1.04
MLP 6:40-6-1:1 for NO						
ERROR	28.62	35.76	31.79	27.55	27.71	29.98
QUOTIENT	1.04	1.30	1.50	1.00	1.01	1.09
MLP 6:40-12-4-1:1 for NO_2						
ERROR	13.24	14.99	21.36	14.68	12.91	15.21
QUOTIENT	1.07	1.22	1.73	1.19	1.05	1.23
MLP 6:40-2-1:1 for CO						
ERROR	0.65	0.82	0.69	0.65	0.65	0.66
QUOTIENT	1.00	1.26	1.06	1.00	1.00	1.01
MLP 6:40-3-1:1 for PM10						
ERROR	41.40	55.52	54.91	44.42	57.28	43.93
QUOTIENT	1.10	1.47	1.46	1.18	1.52	1.17

On the basis of information on the network error and quotient, the rank – i.e. the share in formation of the network output signal – for each meteorological parameter was defined in particular models. The rank is the natural number, from 1 to 6, denoting the place of a variable in the hierarchy of essentiality (the most essential variable was of rank equal 1).

Table 3. Ranks of meteorological parameters in some models

	Wind direction	Wind speed	Temperature	Humidity	Solar radiation	Pasquil's stability category
LINEAR 6:40-1:1 for SO_2						
RANK	3	4	1	6	5	2
MLP 6:40-6-1:1 for NO						
RANK	4	1	2	6	5	3
MLP 6:40-12-4-1:1 for NO_2						
RANK	5	3	1	4	6	2
MLP 6:40-2-1:1 for CO						
RANK	6	1	2	4	5	3
MLP 6:40-3-1:1 for PM10						
RANK	6	2	3	4	1	5

From the sensitivity analysis following inferences were drawn:

- The input variables with the greatest effect on the output were the air temperature (SO_2 with error 51.16, NO_2 with error 21.36) and wind speed (NO with error 31.97, CO with error 0.69);
- The lowest effect on the output value of concentration had air humidity (ranks of this input variable were 4 or 6).

The presented results did not terminate computations. Attempts were undertaken to generate better networks by eliminating, while training, the variables weakly correlated with the investigated phenomenon. The variables with the highest ranks were rejected. Such reasoning is supported by the literature [13–15]. Simplification of the network structure by rejecting the variables with low effect on the network performance may improve the network quality. The new neural models were created by rejecting the input variables with ranks 6, 5, and 4, i.e.: for PM_{10} – wind direction, relative air humidity, atmosphere stability category; for SO_2 – relative air humidity, solar radiation, wind speed; for NO – relative air humidity, solar radiation, wind direction, for CO – relative air humidity, solar radiation, atmosphere stability category; for NO_2 – relative air humidity, solar radiation, wind direction [12]. Computations were performed by using AND. The basis for this experiment were so far constructed neural models. Results are presented in Table 4.

Table 4. Characteristics of neural models obtained by rejecting input variables with the greatest ranks

Output variable	Type and structure of network	Regression coefficient	Correlation	Error
SO_2	LNN - 3:37-1:1	0.69	0.72	32.61
NO	MLP - 3:8-1-1:1	0.84	0.55	28.91
NO_2	MLP - 3:8-15-15-1:1	0.95	0.34	12.93
CO	MLP - 3:32-4-1:1	0.73	0.69	0.59
PM_{10}	MLP - 3:3-15-1:1	0.73	0.70	44.60

As follows from Table 4, for any pollutant no better neural network was found. In spite of rejecting the variables of small importance, or even seeming useless, attempts to improve parameters of the previously generated networks failed.

Such results forced the conclusion that with the software and hardware used, the presented in Table 1 neural models are the optimum solutions to the considered problem. Moreover, it may be stated that the generated neural networks, predicting SO_2, NO, NO_2, CO, PM_{10} concentrations, perform better when they have more input parameters (independent variables). More diversified input data provided better accuracy of the network performance in this case.

CONCLUSIONS

The essentiality of hierarchy of variables was built up by assessing the effect of each variable in the neural model on the pollutant concentrations. This effect was expressed as worsening of the model ability to simulate the relation between input and output for the

training data after rejecting this variable. Adequacy of such assessment depends equally on accuracy of the model in expressing the concentrations as many variable functions and on individual effect of each of the variables on the concentration.

The influence of an input variable on fitting the neural model for real data (for measured quantities) is expressed by contribution of this variable (here meteorological parameter) to formation of the neural model output signal (in this case – concentration). So, analysing the results of above considerations one can assess how far real meteorological conditions affect concentrations of air pollutants. This may be supported by consistency of the received information on essentiality of particular meteorological parameters (to prediction of pollutant concentrations by using neural networks) with the general tendency of influence of meteorological factors on pollutant propagation. The results confirming predominant effect of wind and air temperature on pollutant propagation may be considered a proof of this.

Despite of the quotient being equal 1 for some variables (theoretically they did not affect the network quality) never the network error of the neural network constructed after their rejecting was lower than the network error of the neural network built with all input variables. Consequently, it may be concluded from the carried out analysis that the more input variables the neural network – created to predict concentrations of substances in the air on the basis of meteorological data – posses, the better it performs.

If one assumes that on the basis of the sensitivity analysis the effect of meteorological conditions on pollutant concentration may be evaluated, then one may expect closeness of ranks of variables depending on each other. Indeed, comparing ranks of, for example, solar radiation, air temperature and atmosphere stability category in a neural model for some pollutant one may see similar participations of these variables in formation of output (for PM_{10} the solar radiation rank is 1, air temperature rank is 2, stability class rank is 4; for NO the solar radiation rank is 3, air temperature rank is 2, stability class rank is 4).

REFERENCES

[1] Bilski J., L. Rutkowski: *Sieci neuronowe i neurokomputery*, Wydawnictwo Politechniki Częstochowskiej, Częstochowa 1996.

[2] Cibas T., F. Fogelman-Soulié, P. Gallinari, S. Raudys: *Variable selection with neural networks*, Neurocomputing, 12, 223–248 (1996).

[3] Egmont-Petersen M., J.L. Talmon, A. Hasman, A.W. Ambergen: *Assessing the importance of features for multi-layer perceptrons*, Neural Networks, 11, 623–635 (1998).

[4] Férod R., F. Clérot: *A methodology to explain neural network classification,* Neural Networks, 15, 237–246 (2002).

[5] Gardner M.W., S.R. Dorling: *Artificial Neural Networks (The Multilayer Perceptron) – A Review of Applications in the Atmospheric Sciences*, Atmos. Environ., 32, 2627–2636 (1998).

[6] Gardner M.W., S.R. Dorling: *Neural network modeling and prediction of hourly NO_x and NO_2 concentrations in urban air in London*, Atmos. Environ., 33, 709–719 (1999).

[7] Jorqueva H., R. Perez, A. Cipriano, A. Espejo, M. V. Letelier, G. Acuna: *Forecasting ozone daily maximum levels at Santiago, Chile*, Atmos. Environ., 32, 3415–3424 (1998).

[8] Juda J., S. Chróściel: *Ochrona powietrza atmosferycznego*, Wydawnictwo Naukowo-Techniczne, Warszawa 1974

[9] Perez P.: *Prediction of sulfur dioxide concentrations at a site near downtown Santiago, Chile*, Atmos. Environ., 35, 4929–4935 (2001).

[10] Perez P., J. Reyes: *Prediction of maximum of 24-h average of PM10 concentrations 30 h in advance in Santiago, Chile*, Atmos. Environ., 36, 4555–4561 (2002).

[11] Perez P., A. Trier: *Prediction of NO and NO_2 concentrations near a street with heave traffic in Santiago, Chile*, Atmos. Environ., 35, 1783–1789 (2001).

[12] Rogula W.: *Identyfikacja mechanizmów dominujących w rozprzestrzenianiu zanieczyszczeń przy użyciu sztucznych sieci neuronowych*, Politechnika Śląska w Gliwicach, 2003, praca dyplomowa niepublikowana.
[13] Statistica Neural Networks PL: *Kurs użytkowania programu na przykładach*, StatSoft, 2001.
[14] Statistica Neural Networks PL: *Przewodnik problemowy*, StatSoft, 2001.
[15] Statistica Neural Networks PL: *Wprowadzenie do sztucznych sieci neuronowych*, StatSoft, 2001.
[16] Tadeusiewicz R.: *Sieci neuronowe*, Akademicka Oficyna Wydawnicza RM, Warszawa 1993.
[17] Żurada J., W. Barski , W. Jędruch: *Sztuczne sieci neuronowe*, Wydawnictwo Naukowe PWN, Warszawa 1996.

Received: September 27, 2004; accepted: February 10, 2005.