Cezary Orłowski[*]
Arkadiusz Sarzyński

A MODEL FOR FORECASTING PM10 LEVELS WITH THE USE OF ARTIFICIAL NEURAL NETWORKS

Summary. This work presents a method of forecasting the level of PM10 with the use of artificial neural networks. Meteorological data was taken into account in the construction of the model, and unidirectional networks were used to implement it due to their ease of learning. Then, the configuration of the network (built on the basis of the developed model) was established, defining the number of layers and neurons, as well as the activation function. 4 methods of propagation (BackPropagation, ResilientPropagation, ManhattanPropagation and ScaledConjugateGradient) were applied in the network learning process to select the best method. The obtained results were then compared with real values and the complete network configuration (minimizing the forecast error) was determined. After completion of the learning process, the developed network was used to forecast the particulate matter levels in Gdansk.

1. Introduction

The consequence of technological and industrial development in the modern economy is the increase in emissions of harmful substances and particulates. PM10 is especially dangerous for living organisms. The World Health Organization estimates that over two million people die every year as a result of complications caused by PM10. Organizations around the world seek to limit its emissions. The maximum allowable average daily level of particulates is $50\mu g/m^3$ and it is necessary to control this level. What is also important is that the level needs to be forecast, as operational activities can be determined on the basis of such a forecast, especially in the case when the alarm level is exceeded. The existing forecasting methods only allow for the determination of the average daily level. This paper will present an attempt to build a model for forecasting hourly levels of PM10 (24 hours into the future). The model will then be implemented using a unidirectional network. Thus, conditions for forecasting the level of PM10 for 24 hours in advance will be created, which will have a reasonable limit of error.

2. The construction of the model to predict the level of particulate matter

The first stage necessary to begin the work was the construction of the model. The input and output variables were defined after initially identifying the variables which affect the level of PM10. It was established that the future value of the PM10 level depends on the prevailing meteorological conditions and the current value of the level of particulate matter [2]. It was also concluded that virtually all meteorological data may affect the future level of PM10. Therefore, each of them was studied independently to select only those parameters that have an impact on the level of air pollution. Research was carried out with the use of the method of statistical analysis to examine the correlation coefficients of the individual parameters.

---

[*] Gdańsk University Of Technology, Faculty of Management and Economics, Department of Information Technology Management
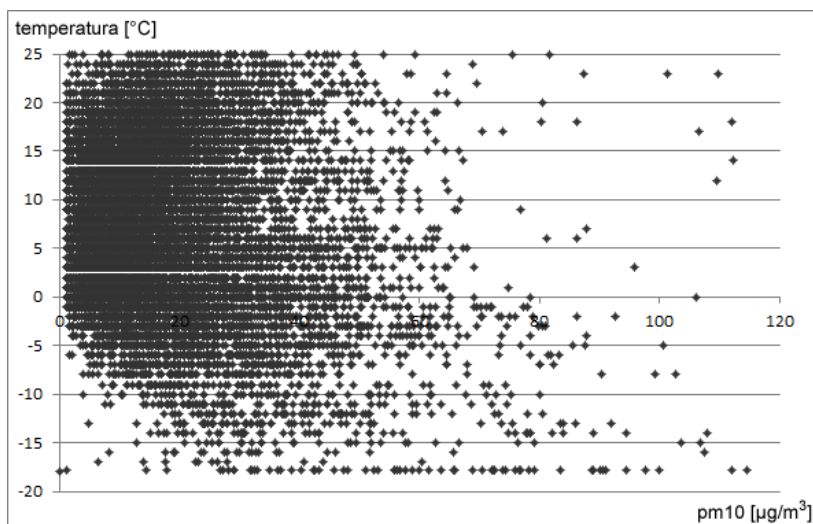
Figure 1. The results of the correlation analysis of variables: air temperature and PM10

Figure 1 shows the results of the correlation analysis of variables: air temperature and PM10. Data used in the study involved temperature values ranging from -20 to 25 degrees and the level of particulate matter from 0 to 120 µg/m^3. It can be concluded (on the basis of the data in Figure 1) that the level of particulate matter decreases with increasing temperature. Thus, changing air temperature can be used in the construction of the model.
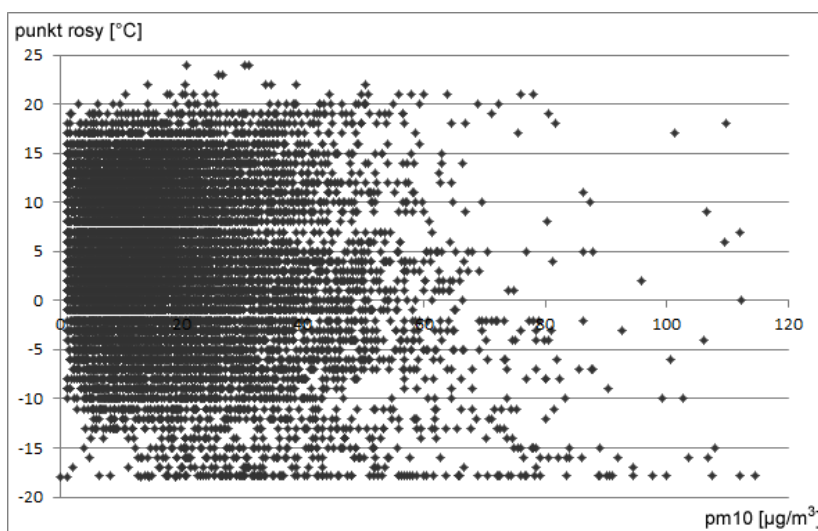

Figure 2. The results of the correlation analysis of variables: dew point and PM10

Another analysis of a variable, from the perspective of the model proposed in this work, is the analysis of the impact of the dew point (in the range from -20 to 25) on the level of PM10. Figure 2 (the correlation of the dew point and the level of PM10) shows that the level of particulate matter decreases with the increasing dew point. For this reason, the dew-point variable can be used in building the model.
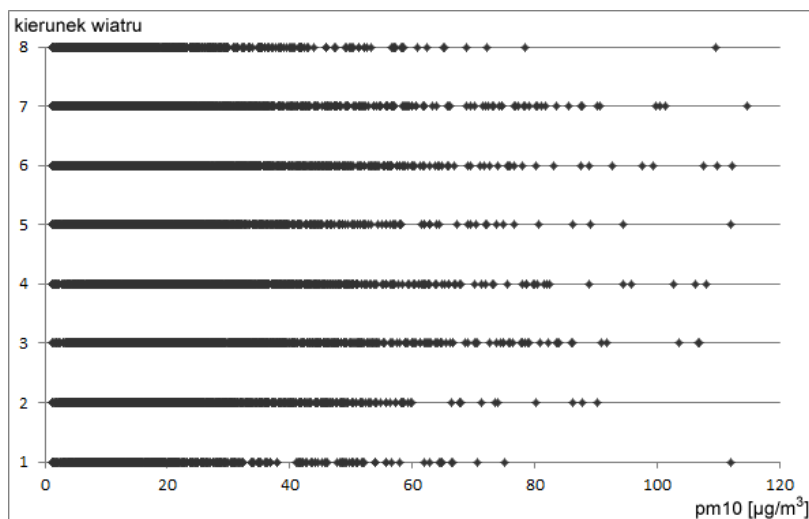
Figure 3. The results of the correlation analysis of variables: wind direction and PM10.

Another variable analyzed in reference to the proposed model was the wind direction. The values of this variable (0-360 degrees) were assigned (for ease of calculation) to one of the eight zones (0-45, 45-90, 90-135, 135-180, 180-225, 225-270, 270-315, 315-360). The correlations of the variables: wind direction and the level of PM10 were analyzed. Figure 3 shows the results of these analyses. A relationship between these variables was identified. The variables corresponding to the adopted zones have an impact on the level of particulate matter. For example, the wind blowing from the direction marked as 1 and 8 caused much less indication of the level of PM than that blowing in the direction marked as 3 and 7.
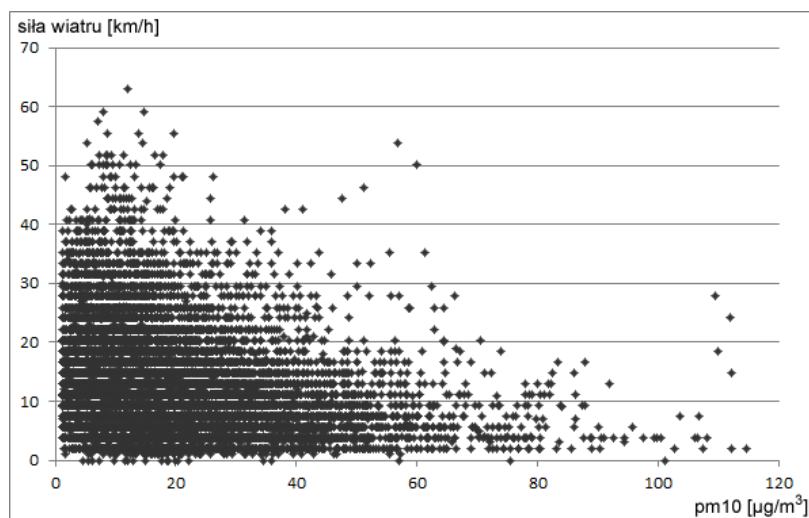
Figure 4. The results of the correlation analysis of variables: wind power and PM10.

The power of the wind was another analyzed variable. Analyses similar to those presented above were carried out. According to Figure 4 (showing the correlation of the wind power variable and the level of particulate matter variable), this correlation is high. In low wind, the level of PM10 reaches a higher level, and when the wind is strong, the content of PM in the air is lower.

Figure 5. The results of the correlation analysis of variables: air pressure and PM10

Atmospheric pressure is another variable which needs to be taken into account in the construction of the model. Figure 5 shows that the level of correlation of the variables: air pressure and the level of particulate matter is significant. It is visible that an increase in the level of PM10 takes place with an increase in atmospheric pressure.
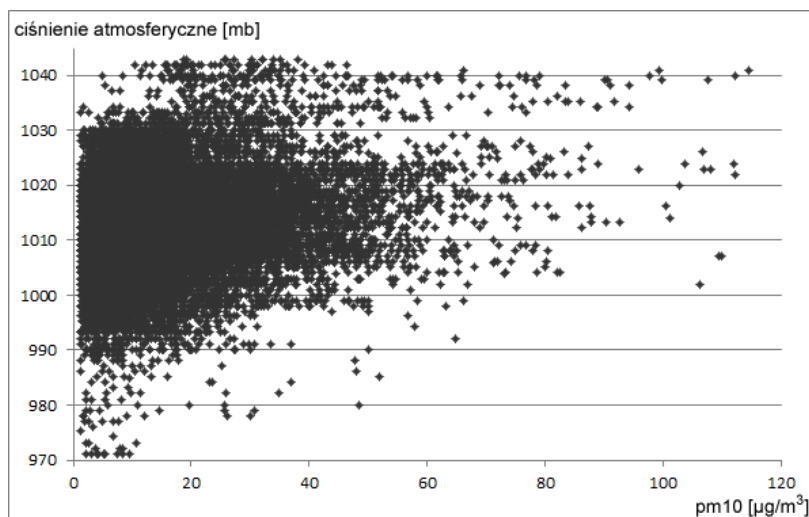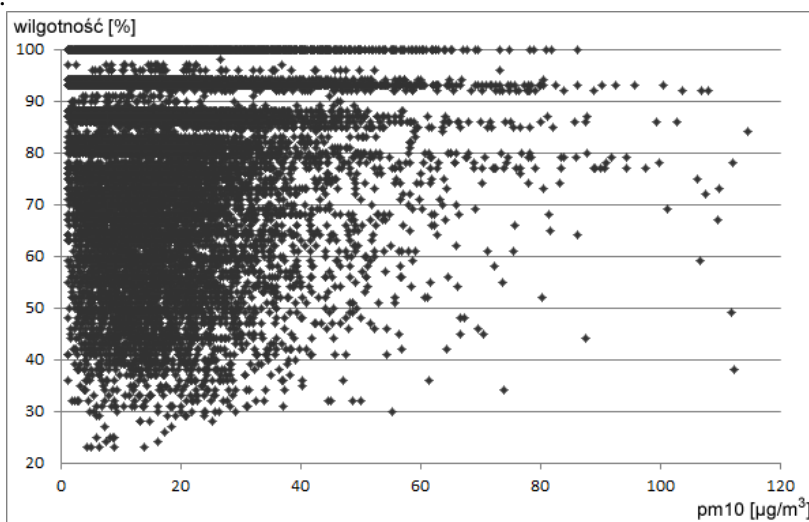

Figure 6. The results of the correlation analysis of variables: humidity and PM10.

The last of the studied variables is humidity. When analyzing the correlation of the variables: the level of air humidity and PM10 (Figure 6) it was discovered that, as in the case of atmospheric pressure, the level of air pollution increases with increasing humidity. Similarly (to the atmospheric pressure variable) this variable also affects the level of PM10.

After conducting the correlation analyses, the results of which are shown in Figures (1-6), the structure of the input variables in the proposed model was established. The input variables are to include the current value of the level of air pollution with PM10, as well as 6 essential weather parameters (air temperature, dew point, wind direction, wind power, atmospheric pressure and humidity). A change in any of these parameters differently affects the future value of PM10 levels, either raising or lowering its value. It is thus necessary to take into account all of these variables in predicting the level of air pollution.

The next step in the construction of the model was to specify the output variables. The aim of the study was to forecast the future value of PM10 levels, therefore the output variable achieved after applying the model should be a single variable, corresponding to the future value of the level of air pollution with PM10. To sum up, the proposed model consists of 7 input variables.

3. The implementation of the model - unidirectional multi-layer artificial neural network

The following stage was the implementation of the model described above. The method of implementation had to be chosen on the basis of the model, which would allow the forecast of the future value of the PM10 level. The issue of forecasting particulate matter is a non-linear problem. An artificial neural network is a method which is used for solving this type of problem. It is a method involving modelling complex non-linear functions with numerous independent variables [3]. In the case of forecasting PM10 levels, the independent variables are meteorological factors and the current level of PM10. There are no predetermined algorithms processing data within the network. It is assumed that the network adapts itself to the problem being solved. This is the main advantage of neural networks - their versatility and applicability in terms of solving various issues, including predicting the future. The choice of a neural network as the forecast method is therefore justified.

Two main types of neural networks were taken into consideration: unidirectional and recursive. For most issues, unidirectional networks are implemented. This is due to the fact that they assist in solving the majority of problems, and are also much easier to implement. Recursive networks are used to solve more complicated problems, such as complex optimization problems. Hence, a unidirectional network was chosen for the forecasting of the PM10 level, and also used for solving other prediction problems.

An important step in the construction of a unidirectional neural network is to determine whether a single-layer or multi-layer network will be used. Single-layer networks are of little use. Apart from the input layer (where no calculations are made), it only has one layer of neurons (individual network elements), which serves as the output layer. The neurons of this layer operate independently of each other. The capability of such a network is limited to the capability of individual neurons. Single-layer networks are used mainly in simple decision problems. In the case of forecasting, a multi-layer network needs to be used. It is characteristic for multi-layer networks that in addition to the input and output layer, they have at least one hidden layer. Through the use of more than one layer, they allow closer mapping of any given complex issue. The vast majority of problems can be solved with the use of only one hidden layer, including problems associated with forecasting. More complicated problems, such as image recognition, require the use of two or more hidden layers. The constructed network will contain only one hidden layer.

The next step in the construction of the neural network is to determine the number of neurons in each layer of the network (the input, output and hidden layers). The input layer contains as many neurons as there are input variables. The adopted model assumes 7 input variables so there will be exactly 7 neurons in the input layer. The output layer of the network contains neurons corresponding to the variables which the network should come up with after completing the calculation. As stated in the model, it is agreed that the output should involve only one variable, therefore the output layer will consist of only one neuron. It is more complicated to determine the amount of neurons in the hidden layer. There are no methods or algorithms for determining their specific number. Determining the optimal (in terms of network utilization) number of neurons in the hidden layer needed for solving the problem of forecasting the levels of PM10 will constitute one of the studies described in the following paragraphs of this work. The model of the structure of the implemented neural network, constructed on the basis of such assumptions, is shown in Figure 8.
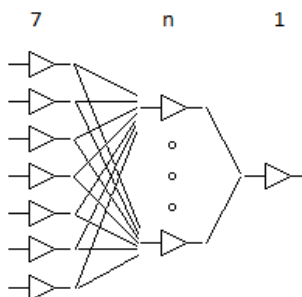

Figure 7: The applied model of the artificial neural network

The next stage of constructing the network is the adoption of the activation function. The choice of this function depends on the problem which the network must solve. For multi-layer neural networks, where the output values are expected to be in a specified range, non-linear activation functions are used. Neurons then

show the greatest ability to learn. This process involves a smooth mapping of any relationship between the input and output network variables. There are two main non-linear activation functions [8]: unipolar sigmoid (Figure 9) and bipolar sigmoid (Figure 10).
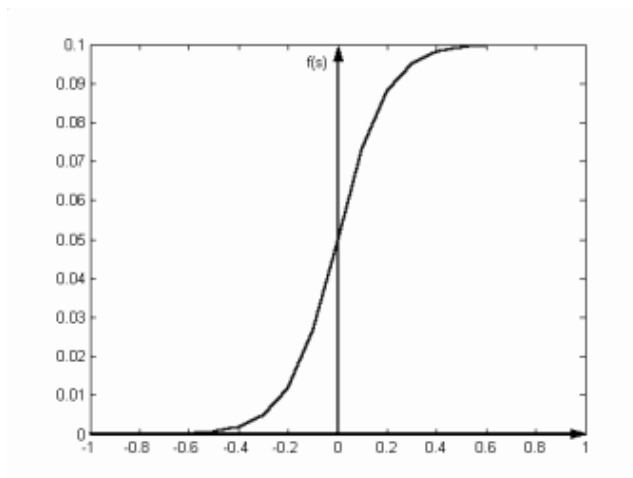

Figure 8. Unipolar sigmoid activation function
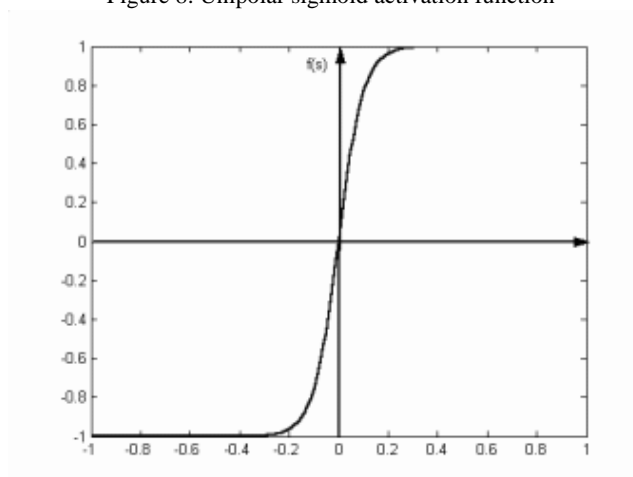
Figure 9. Bipolar sigmoid activation function

It was decided that, from the point of view of the study, it does not matter whether the unipolar or bipolar sigmoid function is selected. Each of these functions can be easily converted to the other without loss of accuracy. For the constructed network, the bipolar sigmoid function will be used, which takes values in the range of (-1, 1). Based on the above set configuration parameters (unidirectional network, one hidden layer, 7 neurons in the input layer, one neuron in the output layer, bipolar sigmoid activation function), the developed model was implemented with the use of an artificial neural network.

4. The learning process of the developed model

It is assumed that the implementation of the artificial neural network is not sufficient for achieving the correct results (forecasting PM10 levels). For reasoning, we should have structured knowledge [1, 4]. After the model implementation phase using the unidirectional network, it becomes necessary for the network to learn. Doing so involves providing the network with the values of the input and output variables. Entering the values is followed by a process of modifying weights at the connections between neurons in the adjacent layers. The value of the network output variable depends on the values of these weights. This process is repeated until the network calculation error reaches the setpoint (minimum) value.

The supervised learning method was assumed to be carried out for the network learning process. This means that when entering the input values into the network, the values of the output variables were already predetermined. To obtain those values, the weights were modified within the network structure. To use this

method, first the input and output variables which the network will use to learn on must be obtained. Another important requirement, in relation to the values of the variables, was their cyclical one-hour reading on the measuring equipment. This approach resulted from forecasting the level of PM10 in 24-hour cycles. Archival values of PM levels were acquired from the Foundation: Agency of Regional Air Quality Monitoring in the Gdańsk metropolitan area (ARMAAG). The meteorological data was freely available on the Internet. The location of the measurement stations which were the source of the data was also very significant. The station located in the shortest distance from the meteorological measurement station was selected to be the source of data on the particulate matter, which was located on Ostrzycka Street in Gdansk (Szadółki district). The distance between the two stations was 10 km. The data obtained from them covered two full calendar years (2012 and 2013).

After gathering the training data, the best method of learning had to be chosen, which could be applied to the implemented network structure. The basic method of supervised learning for multilayer, unidirectional networks is the BackPropagation method. The BackPropagation algorithm is based on the decline in the value of the sum of squared errors of learning so as to get closer to the assumed minimum value. This process begins by changing the weights on the connections between the output neurons and the penultimate layer of neurons. Then, it heads back toward the input layer. The weights are updated according to the principle of minimizing the error. The speed with which the training error decreases depends on the network learning coefficient. It is often difficult to find the right learning coefficient. The BackPropagation algorithm was used for training the network proposed in this work, which was expanded with the automatic selection of the learning coefficient. Faster learning algorithms were also considered: resilientpropagation and manhattanpropagation. In these algorithms, the weights of the input neurons are not determined by the size of the gradient and the learning coefficient (as is the case in the regular BackPropagation). To change the weights, both methods use the relationship given in formula 1 [5]

$$\Delta w_{ij} = -sign(\frac{\partial E}{\partial w_{ij}})\Delta_{ij} \qquad (1)$$

When changing the weights, the two methods differ only in terms of the Δij value. To update the manhattanpropagation algorithm, a constant value of Δij is used. In the resilientpropagation algorithm, the Δij value varies and depends on whether the minimum value of the network error was obtained in the last iteration. The effectiveness of training the network with the use of a fixed Δij method - manhattanpropagation, and a variable Δij - resilientpropagation, will also be verified. The final learning algorithm taken into account is the Scaled Conjugate Gradient algorithm (SCG). It is used to solve problems in large-scale networks. SCG belongs to the class of conjugate algorithms which show a high superlinear similarity in a large number of nonlinear problems [6]. Through the use of SCG, the time-consuming linear search of the per one iteration of learning is avoided. This approach makes the algorithm faster than other learning algorithms, in the case of some problems, and gives better results. The learning algorithms discussed above (BackPropagation, ResilientPropagation, ManhattanPropagation, ScaledConjugateGradient) have been implemented and adapted to cooperate with the proposed neural network. After learning, the knowledge base(neural network) will be built, which will be used for reasoning [7].

5. PM10 forecasting process with the use of the developed model

The first necessary step in the network forecasting was to determine the verification method to check the correctness of the results returned by the network. For this purpose, the RealPM10 function was implemented. Its aim is to check the accuracy of forecasts. It converts the values returned by the network (from -1 to 1) to the corresponding levels of PM10. It collects the absolute differences between the expected (real) value and the value obtained by the network, and then calculates their average (mean error). In addition, the maximum and minimum difference is also checked. Errors are reported in the PM10 measurement units - $\mu g/m^3$.

In order to properly check the functioning of the network, it was necessary to prepare a testing set. To achieve this, 48 days were selected from the training set comprising two full calendar years (and removed from the training set). The selection principle was as follows: in each month of the year, two days were selected: one at the beginning of the month (the first day) and one in the middle of the month (the fifteenth day). In this way, a spectrum was obtained covering two full calendar years with an interval of approximately two weeks. Moreover, such hours were selected to cover the scope of 24 hours. All the selected dates are shown in Table 2.

Table 2. Data used in the process of testing the accuracy of the network

12-01-01 01:00	12-01-15 13:00	12-02-01 03:00	12-02-15 15:00	12-03-01 05:00	12-03-15 17:00	12-04-01 07:00	12-04-15 19:00
12-05-01 09:00	12-05-15 21:00	12-06-01 11:00	12-06-15 23:00	12-07-01 13:00	12-07-15 01:00	12-08-01 15:00	12-08-15 03:00
12-09-01 17:00	12-09-15 05:00	12-10-01 19:00	12-10-15 07:00	12-11-01 21:00	12-11-15 09:00	12-12-01 23:00	12-12-15 11:00
13-01-01 01:00	13-01-15 13:00	13-02-01 03:00	13-02-15 15:00	13-03-01 05:00	13-03-15 17:00	13-04-01 07:00	13-04-15 19:00
13-05-01 09:00	13-05-15 21:00	13-06-01 11:00	13-06-15 23:00	13-07-01 13:00	13-07-15 01:00	13-08-01 15:00	13-08-15 03:00
13-09-01 17:00	13-09-15 05:00	13-10-01 19:00	13-10-15 07:00	13-11-01 21:00	13-11-15 09:00	13-12-01 23:00	13-12-15 11:00

Testing the proper operation of the network involved having the network generate output for the selected data (Table 2). Then the RealPM10 function was applied with the generated results as the function input and 3 values of the network error were returned: the average, maximum and minimum error.

After determining the verification method for checking the proper operation of the network, the forecasting process was initiated. The first stage of the study was to choose the best learning methods and the number of neurons in the hidden layer of the neural network. An insufficient number of neurons in the hidden layer is associated with the network having difficulty in learning. The lengthy learning process is caused by problems with the optimal adjustment of weights to a small amount of neurons. On the other hand, if the number of neurons is too high, this may cause the network's inability to generalize its calculations and lead to an increase of the actual error. In both cases, the network learning error falls to a certain level. With seven neurons in the input layer, the reasonable minimum number of neurons in the hidden layer, which is the starting point of the test, is 5.

The upper limit at which the study should be terminated was set at 37 neurons. If the results were inconclusive after the first study, the scope would be extended. The test was run twice and the results were averaged. A 60-minute learning process was carried out for each possible combination of learning methods (BackPropagation, ResilientPropagation, ManhattanPropagation, ScaledConjugateGradient), as well as for the number of neurons in the hidden layer (5 to 37). After that, the network performance was verified with the use of RealPM10. Table 3 and Table 4 show the results for the number of neurons (the 'Neur.' column): 'Error' - network learning error, 'PM10' - the average actual error of PM10, 'MaxPM10' - the maximum actual error of PM10, 'MinPM10' - the minimum actual error of PM10, 'SumPM10' = 'PM10' + 'MaxPM10' + 'MinPM10'. On the basis of the 'SumPM10' column, the configurations giving the smallest errors were marked.

Table 3. Results of the ANN (artificial neural networks) trained with the BackPropagation and ResilientPropagation methods.

BackPropagation					ResilientPropagation				
Neur.	Error[%]	PM10	MaxPM10	MinPM10	SumPM10	Neur.	Error[%]	PM10	MaxPM10	MinPM10	SumPM10
5	1.47209	8.39246	30.60900	0.13300	39.13446	5	1.46186	8.38830	37.78150	0.08550	46.25530
6	1.46567	8.19068	30.60900	0.13300	38.93268	6	1.44244	8.46648	37.78150	0.08550	46.33348
7	1.45000	8.20002	30.60900	0.08550	38.89452	7	1.43660	8.52513	43.17750	0.02850	51.73113
8	1.41401	8.06906	29.18400	0.28500	37.53806	8	1.40992	8.00058	30.49500	0.23750	38.73308
9	1.45427	8.16485	33.15500	0.28500	41.60485	9	1.39947	7.99445	31.70150	0.14250	39.83845
10	1.42826	8.22733	33.87700	0.06650	42.17083	10	1.38667	7.96687	31.70150	0.14250	39.81087
11	1.39013	8.02354	31.06500	0.72200	39.81054	11	1.38684	7.95744	34.53250	0.06650	42.55644
12	1.43853	8.19316	31.06500	0.65550	39.91366	12	1.36687	8.00959	34.53250	0.06650	42.60859
13	1.44030	8.09064	31.06500	0.65550	39.81114	13	1.36737	8.22014	36.57500	0.06650	44.86164
14	1.36676	8.00355	30.40000	0.19950	38.60305	14	1.35315	8.09321	29.69700	0.25650	38.04671
15	1.41688	7.97970	30.40000	0.19950	38.57920	15	1.35149	8.09994	31.06500	0.25650	39.42144
16	1.42158	8.02156	30.40000	0.12350	38.54506	16	1.34237	8.03667	31.06500	0.00950	39.11117
17	1.37681	7.82266	27.44550	0.12350	35.39166	17	1.32581	8.06689	35.43500	0.35150	43.85339
18	1.42423	7.96971	32.70850	0.12350	40.80171	18	1.32836	8.24758	35.43500	0.12350	43.80608
19	1.40559	7.93870	32.70850	0.00000	40.64720	19	1.31762	8.16419	35.43500	0.12350	43.72269
20	1.35906	7.78723	29.26950	0.08550	37.14223	20	1.32296	7.81652	31.14100	0.02850	38.98602
21	1.40861	8.05046	29.26950	0.08550	37.40546	21	1.30283	7.95922	39.01650	0.02850	47.00422
22	1.42086	8.07071	31.31200	0.08550	39.46821	22	1.30697	8.03416	39.01650	0.02850	47.07916

23	1.35437	7.90875	32.84150	0.34200	41.09225	23	1.28206	9.04578	49.19100	0.01900	58.25578
24	1.41916	7.97535	32.84150	0.08550	40.90235	24	1.30626	8.71209	49.19100	0.01900	57.92209
25	1.41340	7.96133	32.84150	0.08550	40.88833	25	1.27713	8.55627	49.19100	0.01900	57.76627
26	1.34265	7.78406	31.62550	0.11400	39.52356	26	1.25315	7.86956	29.57350	0.21850	37.66156
27	1.42834	7.99385	32.96500	0.11400	41.07285	27	1.31265	8.07253	33.25950	0.09500	41.42703
28	1.40594	8.04175	32.96500	0.11400	41.12075	28	1.27251	8.15206	35.24500	0.09500	43.49206
29	1.33171	8.11973	29.97250	0.02850	38.12073	29	1.26381	7.48244	40.33700	0.01900	47.83844
30	1.40242	8.00504	29.97250	0.02850	38.00604	30	1.28057	7.90746	40.33700	0.01900	48.26346
31	1.40191	8.04030	29.97250	0.02850	38.04130	31	1.27036	8.24791	40.33700	0.01900	48.60391
32	1.31708	8.22324	33.03150	0.18050	41.43524	32	1.22728	7.78169	31.60650	0.03800	39.42619
33	1.41072	8.02988	33.03150	0.18050	41.24188	33	1.26536	7.98069	31.60650	0.03800	39.62519
34	1.41593	8.09334	33.03150	0.12350	41.24834	34	1.27900	7.82134	31.60650	0.03800	39.46584
35	1.31274	8.37979	30.67550	0.04750	39.10279	35	1.23477	8.46846	47.12000	0.03800	55.62646
36	1.40871	8.28093	34.18100	0.04750	42.50943	36	1.25137	8.10093	47.12000	0.02850	55.24943
37	1.40454	8.16076	34.18100	0.03800	42.37976	37	1.26581	8.11610	47.12000	0.02850	55.26

Table 4. Results of the ANN trained with the ManhattanPropagation and ScaledConjugateGradient methods.

ManhattanPropagation					ScaledConjugateGradient					
Neur.	Error[%]	PM10	MaxPM10	MinPM10	SumPM10	Neur	Error[%]	PM10	MaxPM10	MinPM10	SumPM10
5	1.46782	8.62897	32.63250	0.15200	41.41347	5	1.46820	8.34060	30.29550	0.38000	39.01610
6	1.46310	8.46767	32.63250	0.15200	41.25217	6	1.47201	8.35129	32.56600	0.27550	41.19279
7	1.45256	8.42307	37.65800	0.15200	46.23307	7	1.45451	8.29977	32.56600	0.27550	41.14127
8	1.42992	8.19533	34.80800	0.09500	43.09833	8	1.41893	8.08015	29.62100	0.36100	38.06215
9	1.42301	7.92547	34.80800	0.06650	42.79997	9	1.46651	8.20186	32.57550	0.36100	41.13836
10	1.40405	7.90301	34.80800	0.06650	42.77751	10	1.45193	8.28941	32.57550	0.00950	40.87441
11	1.40316	7.85294	30.13400	0.68400	38.67094	11	1.42090	8.05640	33.07900	0.06650	41.20190
12	1.41630	8.17772	32.91750	0.68400	41.77922	12	1.45257	8.19494	33.07900	0.06650	41.34044
13	1.39579	8.08034	32.91750	0.20900	41.20684	13	1.45991	8.20068	33.07900	0.06650	41.34618
14	1.37476	7.66353	29.26000	0.12350	37.04703	14	1.46699	8.38692	31.83450	0.52250	40.74392
15	1.38287	7.78426	30.61850	0.12350	38.52626	15	1.47146	8.43254	34.55150	0.13300	43.11704
16	1.39847	7.96562	30.61850	0.12350	38.70762	16	1.46123	8.29746	34.55150	0.13300	42.98196
17	1.34958	8.32180	28.85150	0.86450	38.03780	17	1.43451	7.94913	30.43800	0.07600	38.46313
18	1.39293	8.18316	36.14750	0.04750	44.37816	18	1.60173	8.38543	30.43800	0.07600	38.89943
19	1.39191	8.17501	36.14750	0.04750	44.37001	19	1.48138	8.45210	34.20000	0.07600	42.72810
20	1.34560	7.98297	33.27850	0.09500	41.35647	20	1.41985	8.23017	30.78950	0.85500	39.87467
21	1.38652	7.98366	33.27850	0.09500	41.35716	21	1.47604	8.32606	30.78950	0.32300	39.43856
22	1.40329	8.07962	33.27850	0.09500	41.45312	22	1.48738	8.35789	30.78950	0.21850	39.36589
23	1.32823	7.94259	30.54250	0.05700	38.54209	23	1.45109	8.33981	34.50400	0.49400	43.33781
24	1.39608	8.03433	30.54250	0.05700	38.63383	24	1.50453	8.35159	34.81750	0.29450	43.46359
25	1.37739	8.05422	30.54250	0.05700	38.65372	25	1.49513	8.36858	34.81750	0.29450	43.48058
26	1.30343	7.85729	31.07450	0.05700	38.98879	26	1.43127	8.33566	30.93200	0.37050	39.63816
27	1.38882	7.83710	32.69900	0.05700	40.59310	27	1.50023	8.33734	31.52100	0.22800	40.08634
28	1.38934	7.84331	32.69900	0.02850	40.57081	28	1.48660	8.32965	31.52100	0.22800	40.07865
29	1.32295	7.71143	30.02000	0.02850	37.75993	29	1.42375	8.24818	32.25250	0.16150	40.66218
30	1.38002	7.80692	31.52100	0.02850	39.35642	30	1.51721	8.42264	32.25250	0.16150	40.83664
31	1.37025	7.86521	33.78200	0.02850	41.67571	31	1.50607	8.43283	35.91000	0.16150	44.50433
32	1.29550	8.15674	33.17400	0.04750	41.37824	32	1.44027	8.39622	31.30250	0.24700	39.94572
33	1.39937	8.16584	33.17400	0.04750	41.38734	33	1.52094	8.40562	31.30250	0.24700	39.95512
34	1.39496	8.09605	33.17400	0.04750	41.31755	34	1.51725	8.40176	38.41800	0.24700	47.06676
35	1.31573	8.23393	32.57550	0.00950	40.81893	35	1.45979	8.70932	32.84150	0.14250	41.69332
36	1.39603	8.14457	32.57550	0.00950	40.72957	36	1.55040	8.70724	35.00750	0.08550	43.80024
37	1.38762	8.04373	32.57550	0.00950	40.62873	37	1.53416	8.42571	35.00750	0.08550	43.51871

Comparing the results from Table 3 and Table 4, it can be concluded that the BackPropagation learning method returns the smallest average error among all the tested methods. The average errors are as follows: BackPropagation - 39.7297, ResilientPropagation - 45.4439, ManhattanPropagation - 40.7727, ScaledConjugateGradient - 41.3029. It was decided that the BackPropagation method will be used for further research.

The next stage of research was to determine the number of neurons in the hidden layer. Analyzing the results obtained by the network trained with the BackPropagation method, there were errors generated by the network with 8, 17, 20, and 21 neurons in the hidden layer. They are lower than the errors for other structures (other numbers of neurons in the hidden layer). In particular, a network with 17 neurons is worth noting - total error = 35.39166, much lower than any other. We can therefore conclude that an effective network structure is trained with the BackPropagation method with 17 neurons in the hidden layer (average PM10 error = 7.82266, maximum PM10 error = 27.44550, minimum PM10 error = 0.12350).

After selecting the number of neurons in the hidden layer, the suggested network had yet to be tested in terms of forecast error. Table 5 shows detailed stages of testing the network.

Tabela 5. Testing the network after the BackPropagation learning method.

Date	Fore.pm10	Real.pm10	Difference	Date	Fore.pm10	Real.pm10	Difference
12-01-01 01:00	11.286	4.503	6.783	13-01-01 01:00	12.9675	18.297	5.3295
12-01-15 13:00	13.908	19.703	5.795	13-01-15 13:00	25.783	31.901	6.118
12-02-01 03:00	33.934	6.6025	27.3315	13-02-01 03:00	11.97	9.101	2.869
Explanation: drop in wind + night (light traffic)				13-02-15 15:00	21.489	17.898	3.591
12-02-15 15:00	37.2685	33.896	3.3725	13-03-01 05:00	16.1785	11.9035	4.275
12-03-01 05:00	11.3905	5.301	6.0895	13-03-15 17:00	14.364	1.7955	12.5685
12-03-15 17:00	12.977	12.502	0.475	13-04-01 07:00	14.269	22.097	7.828
12-04-01 07:00	10.564	6.6975	3.8665	13-04-15 19:00	22.648	15.7985	6.8495
12-04-15 19:00	12.1125	14.801	2.6885	13-05-01 09:00	28.234	47.5	19.266
12-05-01 09:00	14.269	21.204	6.935	13-05-15 21:00	14.1455	14.2975	0.152
12-05-15 21:00	21.717	3.496	18.221	13-06-01 11:00	12.5115	21.8975	9.386
Explanation: increase in temp by 5deg. in 4h+drop in wind				13-06-15 23:00	22.0305	12.7015	9.329
12-06-01 11:00	11.533	15.903	4.37	13-07-01 13:00	15.903	13.604	2.299
12-06-15 23:00	15.3425	15.0005	0.342	13-07-15 01:00	11.2195	5.301	5.9185
12-07-01 13:00	23.579	15.2	8.379	13-08-01 15:00	15.6655	9.6995	5.966
12-07-15 01:00	8.4645	2.603	5.8615	13-08-15 03:00	10.944	10.0035	0.9405
12-08-01 15:00	18.772	21.4035	2.6315	13-09-01 17:00	10.5735	3.002	7.5715
12-08-15 03:00	14.9625	10.298	4.6645	13-09-15 05:00	13.3285	3.6955	9.633
12-09-01 17:00	12.7015	6.403	6.2985	13-10-01 19:00	19.3135	6.6975	12.616
12-09-15 05:00	9.177	9.3005	0.1235	13-10-15 07:00	16.3875	32.1005	15.713
12-10-01 19:00	21.6505	49.096	27.4455	Explanation: no wind + morning (heavy traffic)			
Explanation: increased wind + direction from city				13-11-01 21:00	14.9055	3.496	11.4095
12-10-15 07:00	15.6845	13.3	2.3845	13-11-15 09:00	18.1925	15.0005	3.192
12-11-01 21:00	13.509	23.3035	9.7945	13-12-01 23:00	8.455	13.8035	5.3485
12-11-15 09:00	24.9185	49.5995	24.681	13-12-15 11:00	16.188	5.1015	11.0865
Explanation: clouds disappeared + morning (heavy traffic)				**Average error**			7.82265
12-12-01 23:00	19.969	23.199	3.23	**Average error without indicated dates**			5.89588

On the basis of the results shown in Table 5, the cases when the value of the PM10 forecast error was high were selected and analyzed. Some interesting situations when the increased value of the forecast error occurred include these days and times: 12-02-01 03:00, 12-05-15 21:00, 12-10-01 19:00, 12-11-15 9:00, 13-10-15 7:00. A meteorological analysis of these cases followed. After examination, it turned out that the causes refer to phenomena which are very difficult to predict. In these cases they involved for instance an increase in wind power, or snow occurring quite suddenly. On the basis of measurements taken over 24 hours before, they were difficult to predict. After eliminating these cases, the average forecast error decreased from 7.82265 to 5.89588 and the maximum value of the forecast error fell from 27.44550 to 12.5685.

6. Summary

The purpose of the work was to build a model to predict the level of PM10 in urban areas. In the model construction process, first the usefulness of the input variables for the model was assessed, and then a unidirectional neural network was used for implementing the model. The choice of a unidirectional neural network was due to its possibility of solving nonlinear prediction problems with plenty of independent input variables.

The selection of the network was followed by the construction of its structure. It was decided that a three-layer network with one layer of hidden neurons will be used. The choice of only one layer of hidden neurons was due to the nature of the problem to be solved, namely the forecasting of the PM10 level. There is no need to use a larger number of hidden layers for this type of problem. The chosen activation function was bipolar sigmoid, because then neurons show the greatest ability to learn. This function also allows for smooth mapping of any relationship between the input and output variables.

The next step was to develop a method for testing the network. To do so, the results for dates evenly spread over two years were calculated, and the results were given in real PM10 values.

After selecting the network structure and the testing method, the learning algorithm was chosen. The following algorithms were taken into account: BackPropagation, ResilientPropagation, ManhattanPropagation, ScaledConjugateGradient. BackPropagation was eventually applied for the learning process due to the smallest learning error returned by the network.

After selecting the network learning method, the optimal number of neurons in the hidden layer was determined. 5 to 37 neurons were considered. A neural network with 17 neurons in the hidden layer was selected, as it showed the smallest forecast error.

In the next stage, the results of testing were examined in detail. It was concluded that the higher forecast errors are caused by sudden changes in weather conditions that were difficult to predict in the past.

Using the developed model, an average forecast error equal to 7.82265 $\mu g/m^3$ could be obtained.

The most labor-consuming stage in the construction of the network was the stage of learning and collecting results, because learning takes time, and the training configurations were numerous. The selection of input variables proved to be a relatively simple stage, as most of the variables were analyzed in a similar manner and the process ran smoothly.

With the use of a neural network, the future value of PM10 can be predicted, although it is worth considering how to improve the process. There are plans to improve the functioning of the network through adding neurons in the input layer. These neurons would be responsible for forecasting the meteorological conditions, derived from specialized models for predicting the weather. The following process in building a model, in order to increase the accuracy of calculations, would be to increase the size of the training set.

7. References

1. Czarnecki A., Orłowski C., Sitek T., Ziółkowski A.*: Information Technology Assessment Using a Functional Prototype of the Agent Based System//* Foundations of Control and Management Sciences. - 2008, no 9, s. 7-28 : 5 rys., 5 tab. - Bibliogr. 8 poz. - ISSN 1731-2000

2. Drzeniecka-Osiadacz A., Netzel P.: *Wpływ warunków meteorologicznych oraz cyrkulacji atmocferycznej na stężenie PM10 we Wrocławiu*, 2010.

3. Kosiński R.,: *Sztuczne sieci neuronowe Dynamika nieliniowa i chaos*, 2005.

4. Kowalczuk Z., Orłowski C.: *Design of Knowledge-based Systems in Environmental Engineering, Cybernetics and Systems*, vol. 35 nr 5-6, 2004, s. 487-498. 19 rys. 1 tab. bibliogr. 11 poz.

5. Neutelings J.,: *SmarTee: A neural network based navigational bot*, 2011.

6. Peterson C., Rögnvaldsson T., Lönnblad L.,: *JETNET 3.0: A Versatile artificial neural network package*, 1994.

7. Sitek T., Orłowski C.: *Model of Management of Knowledge Bases in the Information Technology Evaluation Environment* // W: Information systems architecture and technology : models of the organisation's risk management / editors: Zofia Wilimowska [et al.]. - Wrocław 2008 : Oficyna Wydawnicza Politechniki Wrocławskiej, 2008. - (Biblioteka Informatyki Szkół Wyższych). - S. 221-231 : 5 rys. - Bibliogr. 9 poz. - ISBN 978-83-7493-413-8

8. Tadeusiewicz R., *Wprowadzenie do praktyki stosowania sieci neuronowych*.