BULLETIN OF THE POLISH ACADEMY OF SCIENCES
TECHNICAL SCIENCES, Vol. 67, No. 3, 2019
DOI: 10.24425/bpasts.2019.129650

Local dynamic integration of ensemble in prediction of time series

S. OSOWSKI[1] and K. SIWEK[2]*

[1]Military University of Technology, Warsaw, Poland
[2]Warsaw University of Technology, Warsaw, Poland

Abstract. The paper presents local dynamic approach to integration of an ensemble of predictors. The classical fusing of many predictor results takes into account all units and takes the weighted average of the results of all units forming the ensemble. This paper proposes different approach. The prediction of time series for the next day is done here by only one member of an ensemble, which was the best in the learning stage for the input vector, closest to the input data actually applied. Thanks to such arrangement we avoid the situation in which the worst unit reduces the accuracy of the whole ensemble. This way we obtain an increased level of statistical forecasting accuracy, since each task is performed by the best suited predictor. Moreover, such arrangement of integration allows for using units of very different quality without decreasing the quality of final prediction. The numerical experiments performed for forecasting the next input, the average PM10 pollution and forecasting the 24-element vector of hourly load of the power system have confirmed the superiority of the presented approach. All quality measures of forecast have been significantly improved.

Key words: neural networks, ensemble of predictors, dynamic integration, time series prediction.

1. Introduction

The time series prediction is an important practical problem in forecasting the next terms of the process. The problem concerns different phenomena, for example the daily average air pollution of CO_2, NO_x, PM_{10}, O_3 or the load, forecasted for the succeeding hours of the day in the power system.

Many different methods have been discussed in the past. The examples are presented in papers [1–10]. Some of them build complex mathematical models of dynamics of the processes using autoregressive linear or nonlinear approach [6]. Recently, neural networks have been the most often used tools in different aspects of prediction and classification problems [11, 12]. The examples in prediction tasks include multilayer perceptron (MLP) or radial basis function (RBF) [7, 8, 11], support vector machine [4, 9] as well as Elman network [10]. The ensemble of many neural predictors has been also proposed [4, 13] to get more accurate forecast.

It was found the combination of various methods outperforms, on average, the individual specific methods and provides better accuracy of prediction. Application of parallel predictors or classifiers forming an ensemble is actually well known method for increasing the accuracy of prediction and classification tasks [14]. Ensembles of predictors are among the most competitive forms of solving predictive tasks. To get good performance of an ensemble the independence of its members should be provided. This may be done in different ways, for example applying the bagging, created through the use of different random bootstrap samples of the original training set, or using different types of predicting units, for example, MLP, RBF, SVM, random forest, autoregression, etc. [13]

The important point in ensemble approach to either classification or regression task is providing the efficient integration (fusion) of the results of its members. In the case of classification different techniques of fusion have been developed: linear or logistic regression, Dempster-Schafer theory or heuristic decision rules [14, 15].

In the case of regression problem, the weighted averaging is used the most often, with the weights dependent on the prediction accuracy estimation of each member [13]. However, this way of fusion may produce the final statistical results inferior to the best unit in an ensemble. The problem is that even the best unit in statistical terms may not show the best performance for the particular days (the best for *i*th day and the worst in *j*th day).

The other approach to integration is the separation of time series predictions made by different units into independent time series and elimination of terms corresponding to the identified noise. Such approach involves the independent component analysis (ICA), after which the reconstruction of time series is performed (so called deflation) using only the important components [13]. The most difficult problem in this procedure is recognition of the noisy terms, that should be eliminated.

This paper proposes different approach to the integration of the ensemble. It is called the local dynamic method. Its idea is somewhat similar to the dynamic integration of classifiers [16, 17]. The final forecast for each testing sample is done by only one unit of the ensemble, best suited for the particular task. The best predictor is selected on the basis of its prediction accuracy for the proper learning sample in the neighbourhood of the actual testing sample. The quality of each member of ensemble is checked on the learning data closest to the actual

*e-mail: krzysztof.siwek@ee.pw.edu.pl

Manuscript submitted 2018-09-24, revised 2018-11-20, initially accepted for publication 2018-11-23, published in June 2019.

testing sample. The most competent predictor, providing the smallest prediction error in the learning mode is chosen. Thanks to this we can get the increased level of forecasting accuracy, since each task is performed by the predictor best suited to it.

The theoretical considerations have been illustrated here by the numerical experiments using two types of prediction problems. One is the prediction of daily average PM10 for the next day (forecasting the single value at a time). The other task is prediction of the 24-elements representing the electrical load for the next day (prediction of vector). The results of experiments, obtained by using Matlab, have shown that our approach leads to the significant improvement of prediction accuracy. The detailed results of these experiments are given and discussed in the paper.

2. The method of local dynamic integration of prediction ensemble

The important step in our approach is the efficient integration of the results of individual predictors into final verdict of the ensemble. The classical approaches to this problem, applying the weighted average method or application of independent component analysis are found not very efficient. We propose here different approach, based on the so called local dynamic principle [16, 17].

In this method only one predictor from the ensemble is used for generating the final forecast. It is this one which was found the best for the learning input sample closest to the actual testing vector. Given an unknown input vector $\mathbf{x}_t$ in testing mode we search for its closest neighbor $\mathbf{x}_l$ among all input vectors existing in the available learning set. The Manhattan distance measure is used

$$d(\mathbf{x}_t, \mathbf{x}_l) = \left\|\mathbf{x}_t, \mathbf{x}_l\right\|_1. \tag{1}$$

This norm was applied because it reflects the differences between individual elements of vectors in the most clear way, not disturbed by the highest value element, as it is in Euclidean norm.

In the next step we compare the quality measure of the applied regression units forming ensemble, in regression task for this vector $\mathbf{x}_l$. The member of the smallest prediction error corresponding to $\mathbf{x}_l$ is selected and used in prediction task at the application of $\mathbf{x}_t$. Its result is regarded as the final verdict of the whole ensemble. In the case of predicting the vector each element of this vector might happen to be predicted by different units of ensemble.

Occasionally, two or more predictors might show the similar highest local accuracy for the tested vector $\mathbf{x}_l$. In such case all of them are used in the prediction task. Final decision of such ensemble is their average result.

The other approach is to use the ensemble of regression networks selected on the basis of few learning vectors, which are the closest to the actual testing vector $\mathbf{x}_t$. For the selected learning vectors the best predictors are chosen and used in testing mode. Final result is the average of verdicts of these predictors supplied by the testing vector.

The data sets used in the training of different individual regression networks forming an ensemble might be different to provide highest independence in their performance.

3. Application to PM10 prediction

3.1. Statement of problem. Prediction of PM10 (particle matters of the diameters up to 10 μm) is especially important, since this pollution level is strictly associated with direct impact on human health via inhalation [18]. Actually, the daily average PM is of importance in all European countries because of the European Air Quality Directive 2008/50/EC defining the restrictions for the yearly and 24 h averages PM_{10} concentrations.

The most important difficulty in forecasting the PM10 level for the next day is very high variability of its concentration from hour to hour and from day to day. This is well seen in Fig. 1, presenting the daily averaged values of PM10 pollution measured by the meteorological station situated in suburb Ursynów of Warsaw in one year $\left(\text{all in } \mu\text{g}/\text{m}^3\right)$. It is evident that the higher the variability of the time series the more difficult is the prediction task.

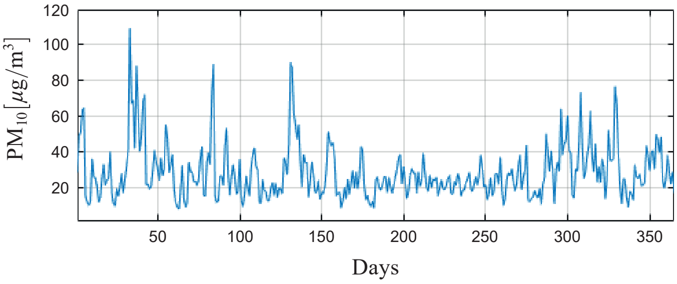

Fig. 1. The changes of PM10 concentration from day to day in 2014 (daily averages)

In building the prediction model we should take into account the relations between PM10 and the diagnostic variables. The applied Hinich test for checking the linearity of the time series has indicated the weak nonlinearity of the process [4, 20]. It means that the nonlinear model of prediction should be better than the linear one.

To get good results of prediction we should develop the diagnostic features serving as the input attributes to the predicting network well correlated with PM10 concentration and presented no correlation among themselves. The most important parameters having the highest impact on the mechanism of pollution creation are: temperature, speed of the wind and humidity. Cross correlation coefficients between pairs of the environmental variables show that they are uncorrelated except the humidity and temperature [13]. Therefore, all of them may be included as the candidates in definition of the diagnostic features in the prediction model. The selection process will discover, which of them should be eliminated.

To get the highest accuracy of prediction the nonlinear model of the process, based on application of few neural networks arranged in the ensemble, will be applied. The approach to prediction of PM10 level is split into few stages. In the first step the diagnostic features are defined on the basis of meteorological variables and past pollution. The selected features are applied as the input attributes to ensemble of predictors, which are trained on the available learning data. In the last step the dynamic local integration of the results of individual predictors is performed.

3.2. Extraction of diagnostic features. The performed investigations of the daily average PM10 changes have shown the significant factors influencing the average pollution level of the day [4]. They include such atmospheric variables, as temperature, wind, humidity, pressure, insolation, the pollution level from the previous day and dependence on the season of the year and type of the day. For example the week days are characterized by the higher level of air pollution compared to the weekend days. Similarly, the highest level of pollution is observed in the winters. These aspects of pollution analysis have been analyzed in [4].

In our considerations we have defined the potential diagnostic features using different mathematical preprocessing methods. The first subset of features represents the environmental parameters for the next day announced by the National Institute of Meteorology: the forecasted 24-hour average value of temperature, wind speed, wind direction, humidity, pressure and insolation.

The next subset takes into account the known past day parameters. They include the average, maximum and minimum values of temperature and pressure, the average and maximum pollution corresponding to the previous day, linear trend of changing the hourly pollution of the previous day, the prediction of the average pollution for the forecasted day, made on the basis of this linear trend and the codes of the season of the year (2-element binary code representing winter, spring, summer and autumn) and day of the week (binary code representing working or weekend days).

The last subset of features is created by the pollution level for the most influential hours of the previous day, which are characterized by the extremal values.

As a result the set of 55 potential features has been created [4]. These features are subject to selection in order to provide the most influential subset. The normalization of the data was implemented by dividing the real values of the particular feature by their mean calculated for all observations.

3.3. Selection of diagnostic features. The features generated in an automatic way should undergo a selection process, discriminating the most influential subset. This process was implemented in the work by using two selection methods: the stepwise linear fit applying the backward and forward selection [21] and the genetic algorithm [22]. The 20% of randomly chosen observations have been used for selection of the features which have the highest impact on the forecasted average values of pollution for the next day.

The stepwise fit starts from an empty feature set and performs sequentially the process of adding (forward selection) not yet chosen features and removing (backward elimination) the features existing in the actual subset [20]. Each candidate feature subset is checked in 10-fold cross-validation by repeating the prediction with different training and testing subsets of observations. Both forward and backward operations interlace each other. In each stage, after adding the new variable, a test is made to check if some variables from the actual set should be deleted without increasing the error of regression. The stepwise fit terminates when the quality measure of the classification model is maximized. The set of 17 features has been selected as a result of application of this stepwise fit procedure.

The genetic selection used in experiments represents the features coded in the chromosomes in a binary way. The unity element value means inclusion of the feature in the input vector **x** to regression system and zero – its exclusion.

Each chromosome is associated with the input vector **x** applied to the SVM classifier (the value 1 means real inclusion of the feature and zero – no such feature in a vector). The elitist strategy of passing to the next generation the two fittest chromosomes in population was applied. This guarantees that the fitness is never declined from one generation to the next.

The learning and validation sets were applied in GA training. The testing error on the validation data forms the basis for the definition of the fitness function. The genetic algorithm maximizes this value (equivalent to the minimization of the error function) by performing the subsequent operations of selection of parents, the crossover among the parents and finally the mutation. The roulette wheel has been applied for selection.

The SVM of Gaussian kernel in regression mode was used as the predictor. The genetic population applied in experiments was equal 100 chromosomes, crossover rate 80% and mutation rate 1%. The sequentially performed genetic operations (crossover, mutation and selection) lead to the minimum of the objective function for the validation data. The unity elements of the best chromosome point to the optimal set of features, which corresponds to the minimum of the validation error. The performed genetic experiments have selected the set of 19 best features.

As a results of application of these two selection methods we get two sets of features. Each of them will form the input attributes to the predicting networks. The ensemble of predictors will contain the multilayer perceptron, radial basis function network, support vector machine in regression mode (SVM) and linear auto-regression with exogenous variables (ARX). The ARX was used only to find how linear model is compared to the nonlinear ones, represented by neural networks.

3.4. Individual predictors. MLP and RBF networks belong to the most universal neural approximators. The most important difference between them is the activation function. MLP applies the sigmoid, belonging to global type of activation function. Thanks to this all neurons participate in formation of the output signal in the whole range of values of input attributes. RBF network operates with Gaussian function of local character and represents local approximation ability. The learning algorithms of both networks are different in a significant way. This is the reason why their output signals in response to the same excitation may also differ and their performance may be treated as statistically independent.

SVM for regression of the Gaussian kernel, transforms regression task into classification by defining some tolerance region of the width ε around the destination [23, 24]. The learning task is reduced to the quadratic optimization problem and is dependent on few hyperparameters: the regularization constant C, the width parameter σ of the Gaussian kernel and the tolerance ε. All of them have been adjusted by repeating the learning experiments for the limited set of their predefined values and accepting this one, which results in the minimum error on the validation data set.

ARX model belongs to the linear predicting systems [20] and is often used for non-stationary time series. The output signal of this model uses the lags of itself and the lags of the exogenous variables. The p lags of the output series form the autoregressive terms, and q lags refer to the exogenous input variables.

Each of these four regressors has been supplied by the set of features chosen by both selection methods. As a result eight predicting systems are constructed by combining the set of features with the regression networks. All regressors have been learned using randomly selected subsets from the available data base. The results of PM10 prediction of each regression unit might be different for each day, since they have been generated by applying different mechanisms of data processing and the applied learning data sets.

3.5. Results of numerical experiments. The experiments have been carried out using the data of PM10 concentration measured in suburb Ursynów in Warsaw within 4 years. The available data have been pre-processed and normalized according to the presented procedure and then split randomly into two subsets: two third of samples have been used in learning and the remaining subset left for testing the trained system. To enhance the independence of individual predictors we have learned them on partially different data samples. To get the most objective results of experiments, the learning and testing have been repeated several times and performed at randomly chosen composition of learning and testing data. Four networks (MLP, RBF, SVM and ARX) associated with two types of feature selection (8 different individual predictors) have been investigated.

The mean values of the results have been calculated and compared to the real PM_{10} concentration for the appropriate days.

In all experiments of learning the parameters of predicting systems (the number of hidden neurons in MLP, RBF, the hyperparameters of SVM and length of ARX) remained the same and have been chosen optimal by using pre-learning experiments on small chosen validation set of data (10% of learning data). The best neural network structures chosen in these experiments contained 8 sigmoidal hidden units in MLP and 40 Gaussian hidden units in RBF. The number of Gaussian kernels of SVM network was automatically adjusted by the learning procedure [22], and in each experiment was different, changing from 18 to 47. The ARX adaptation procedure was found the best for $p = 4$ and $q = 1$ in the ARX model.

To assess the obtained results in the most objective way we have applied different measures of prediction quality. Four measures have been used here: the mean absolute error (MAE)

$$MAE = \frac{1}{n}\left(\sum_{i=1}^{n}\left|t_i - y_i\right|\right) \tag{2}$$

mean absolute percentage error (MAPE)

$$MAPE = \frac{1}{n}\left(\sum_{i=1}^{n}\frac{\left|t_i - y_i\right|}{t_i}\right)\cdot 100\% \tag{3}$$

root mean squared error (RMSE)

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left|t_i - y_i\right|^2} \tag{4}$$

and correlation coefficient (R) of the observed and predicted data

$$R = \frac{C_{yt}}{std(y)std(t)}. \tag{5}$$

The symbol n used in these definitions is the number of data points, y_i – the predicted value, t_i – the really observed value (target), C_{yt} – the covariance value between the really observed and predicted data, and std denotes standard deviation of the appropriate variable.

The statistical results related to application of individual predictors for two applied methods of selection are presented in Table 1. They refer to the average values of 5 runs of procedure on the testing data not taking part in learning. The linear

Table 1
The average values of quality measures obtained for individual predictors in the testing PM10 data (not taking part in learning)

Quality measure	Stepwise fit selection				Genetic selection			
	MLP	RBF	SVM	ARX	MLP	RBF	SVM	ARX
MAPE [%]	26.03	27.07	23.89	35.43	25.55	26.28	22.76	32.45
MAE $[\mu g/m^3]$	7.621	7.719	7.548	10.564	7.637	7.814	7.206	9.879
RMSE $[\mu g/m^3]$	10.880	10.970	10.610	17.113	11.740	11.850	11.860	16.723
R	0.893	0.890	0.892	0.523	0.874	0.869	0.871	0.554

Table 2
The comparison of the average values of quality measures in prediction the daily average PM10 pollution level obtained by an ensemble integrated using different methods of integration

Quality measure	Stepwise fit selection		Genetic selection		Stepwise fit + genetic selection
	Classical approach	Local dynamic	Classical approach	Local dynamic	Weighted averaging of both selection methods
MAPE [%]	23.63±2.34	**20.51**±1.56	22.88±2.12	**18.81**±1.53	**18.62**±1.46
MAE [μg/m^3]	7.12±0.97	**6.29**±0.87	6.99±0.89	**5.82**±0.65	**5.79**±0.59
RMSE [μg/m^3]	10.39±1.78	**10.37**±1.42	10.27±1.69	**9.99**±1.39	**9.90**±1.31
R	0.911±0.051	**0.921**±0.042	0.920±0.0.060	**0.939**±0.032	**0.935**±0.033

ARX model is evidently of the inferior quality in respect to all measures.

The detailed results of performance of all predictors for the days of the learning data have been memorized and used in dynamic integration of the ensemble for the whole testing data set.

For example let us consider the forecast for the tested day represented by the normalized vector $\mathbf{x}_t$ of features (17 element vector) selected by the stepwise fit

$$\mathbf{x}_t = [1.4679\ 0.2615\ 0.0442\ -0.2307\ 0.1846\ 1.3082\ -0.0053\ 1.0679\ -1.0698\ 0.9198\ -1.3643\ 1.2897\ 1.7712\ 1.1412\ 2.0535\ 0\ 1].$$

The learning feature vector in the data base found as the closest to it is as follows

$$\mathbf{x}_l = [1.4482\ 0.2615\ 0.0394\ -0.2317\ 0.1849\ 1.3085\ -0.0065\ 1.0574\ -1.0688\ 0.9192\ -.0746\ 1.2897\ 1.5810\ 1.3848\ 1.9851\ 0\ 1].$$

The error values committed in the past by the applied predictors for this learning vector were as following:

$$\varepsilon_{\text{MLP}} = 20.86\%,\ \varepsilon_{\text{RBF}} = 2.85\%,$$
$$\varepsilon_{\text{SVM}} = 14.07\%,\ \varepsilon_{\text{ARX}} = 27.45\%.$$

There are large differences between the learning errors for this particular vector committed by different predictors. The smallest learning error corresponds to the application of RBF. Hence this network is used for prediction task. The obtained error corresponding to the testing vector $\mathbf{x}_t$ was equal $\varepsilon_{\text{RBF}}(\mathbf{x}_t) = 1.97\%$. Application of other predicting networks have resulted in much larger values of errors: $\varepsilon_{\text{MLP}}(\mathbf{x}_t) = 19.42\%$, $\varepsilon_{\text{SVM}}(\mathbf{x}_t) = 10.64\%$ and $\varepsilon_{\text{ARX}}(\mathbf{x}_t) = 28.45\%$. In a similar way we can apply the procedure for any feature vector in the data base used in experiments.

The testing procedures applied for all days of testing data, not taking part in learning (the same set of testing samples for individual predictors), have allowed estimating the average values of the quality measures. Table 2 presents the statistical results of local dynamic integration. They have been compared to the best classical approach to integration (the weighted averaging of individual ensemble units). In weighted averaging method the results of individual predictors have been combined with the weights proportional to their average forecasting accuracy obtained for the whole learning data sets.

Three different versions of these approaches have been depicted in the table. One corresponds to the application of only stepwise fit, the second to application of only genetic selection and the third one to the combination of both selection methods. In the latter case the weighted averaging of the results, following from application of local dynamic integration for both selection methods, have been presented (last column). The results show the evident advantage of using local dynamic principle of integration. Irrespective of the feature selection method all quality measures of prediction were the best and much better than the best individual predictor.

For example the best result of MAPE corresponding to SVM cooperating with genetic selection was 22.76%, while the best MAPE of local dynamic integration was only 18.62%. It means 18% of relative improvement. These values are also much better than the results presented up to now in the scientific recent publications [1, 25, 26].

Additional experiments performed at application of the whole set of diagnostic features have shown significantly worse performance. For example the MAPE measure at the best configuration of ensemble was equal 24.67% and the value R = 0.893.

4. Application to 24-hour next day load forecasting

The second example will consider the one-day ahead forecasting of the 24-hour load in a small Łódź region of Polish Power System (PPS) by using local dynamic integration of the ensemble. The accurate prediction of the power need in each hour of the day is important in the electrical power market, since it reduces the cost of generation of the energy and enables better management of the natural resources in its production.

4.1. Learning data. The most important point in this problem is the recognition of factors influencing the mechanism of load changes. The analysis of the available data base concerning the hourly changes of load consumption in the past years has revealed the significant dependence of the predicted pattern on its past values, type of the day (workday or weekends and holy days) and four seasons of the year [13]. It means that these factors should be taken into account in building the appropriate mathematical model of the process. The mathematical system of prediction has considered the input data in the form of 24-hour load pattern of the previous day, code of the day type (working versus weekend days) and 2-bits code of the seasons of the year (winter, spring, summer and autumn). As a result the input vector to the predicting system, responsible for forecasting 24 hour load pattern for $(d+1)$ day is of the form

$$\mathbf{x} = [p(d, 1), \ldots, p(d, 24), \text{season\_code}, \text{day\_code}]. \quad (6)$$

in which $p(d, i)$ represents the normalized load of dth day and ith hour, season_code contains 2 bits (11-winter, 10-spring, 00-summer, 01-autumn) and day_code represent day type (1 – working day, 0 – non-working day). It means 27 input attributes to the neural predictor. The destination vector represents the 24-hour power pattern for the next $(d+1)$ day.

The numerical experiments have been performed using the data base of the last 3 years. The maximum load in this small power region did not exceed the value of 500 MW. Total number of days in this base was 1095. The loads corresponding to the following hours of the day have been normalized dividing the real loads by the value representing the maximum in the learning data base. An example of the hourly changes of the normalized load consumption in these years is presented in Fig. 2.

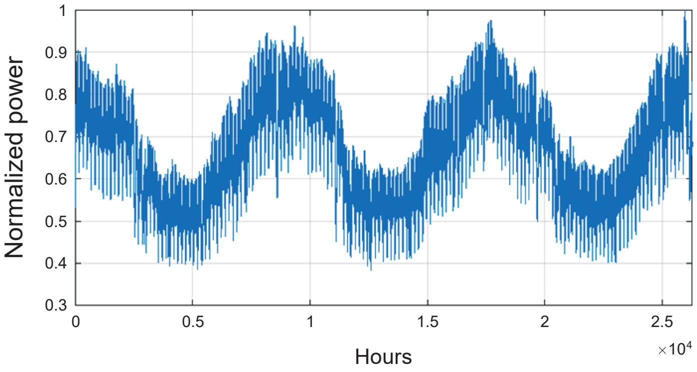

Fig. 2. The hourly change of the normalized power demand in PPS region within three analyzed years

The depicted distribution of the power consumption shows high diversity of the load patterns within the neighboring hours and days, especially between working and weekend days. Large differences are also observed for the days representing different seasons of the year. High variability of values means increased difficulties in their accurate prediction, especially, when we are interested in the whole 24 hour load pattern for the next day. Therefore, in prediction we have applied few individual predictors forming an ensemble.

4.2. Ensemble of predictors. The ensemble of predictors in our application was created on the basis of three different solutions of neural networks: MLP, RBF and SVM, all working in regression mode. These networks have been selected after considering many other possibilities, like Elman network, random forest or linear ARX regression. The features defined by the vector **x** create the input attributes to all members of ensemble.

The predicting system built on the basis of the mentioned above neural networks will be denoted by MLP, RBF and SVM, respectively. All units accept the chosen set of input signals and generate the prognosis of the power consumption for 24 hours of the next day. To increase the independence of ensemble members, each unit is trained on separate set of data chosen randomly from the whole data set. The independence is very important requirement for proper operation of an ensemble.

The forecasts produced by individual predicting units are subject to fusion into one final outcome. We applied here the so called local dynamic integration approach, similar to that used in prediction of the pollution level. However, this time 24 values are expected on the output side of predictor. Moreover, on the basis of additional experiments we have found that the best results are achieved, when few learning vectors closest to the actual testing vector are taken into account in prediction procedure.

In this case the final forecast of the load pattern for the next day is the average of results of individual predictors selected on the basis of similarity of the testing vector $\mathbf{x}_t$ to the learning vectors. As a result the computer searches for few learning input vectors $\mathbf{x}_l$ in the learning data base, which are closest to the actual vector $\mathbf{x}_t$. The decision in selecting $\mathbf{x}_l$ is based on the Manhattan distance between the testing and learning vectors in the base.

The neural networks showing the highest accuracy for the particular hour in the learning phase for these selected learning vectors are chosen as the members of ensemble for this hour. This way each testing vector $\mathbf{x}_t$ is associated with the chosen predicting units of ensemble. Different ensemble units might be chosen for each hour. However, in each case the team contains the neural networks, which were the best for the selected learning vectors. The typical example of performance of three applied neural networks in prediction task of 24 hours for one randomly chosen day is presented in Fig. 3. It represents the

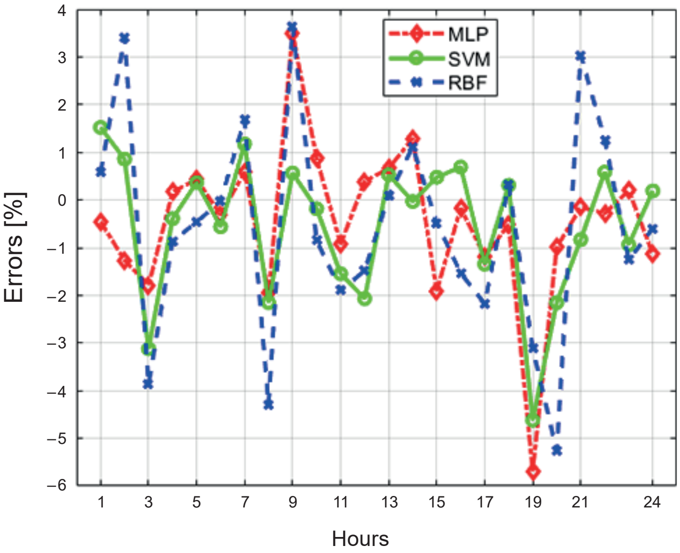

Fig. 3. The relative forecasting errors committed by MLP, RBF and SVM for 24 hours of one day ahead used in learning

relative errors committed by MLP, RBF and SVM for 24 hours of the chosen day from the learning data base.

We can see the changing prediction quality of these networks for different hours. In predicting the power need for any hour of the testing vector, the ensemble integration procedure chooses this network, which was the best (the smallest error) for the particular hour in the learning mode.

In the case presented in Fig. 3 the MLP will be responsible for the forecast of 14 hours, the hours: 1, 3, 4, 7, 8, 11, 12, 16–18, 20–23; SVM for 5 hours, the hours: 2, 5, 14, 15, 24 while RBF also for 5 hours, the hours: 6, 9, 19, 13, 19. These networks produced the best quality predictions in these hours for the learning vector. Therefore, the results of prediction for the tested vector closest to this learning one are expected to be of the highest accuracy. Observe, that the polarity of errors in particular hours for different predictors is changing, which creates the space for compensation in averaging procedure.

The proposed integration method is of the local character, changing dynamically with the vector subject to testing. Thanks to such approach the final forecast of ensemble is never worse than the best result of the individual predictor. This is in contrast to the traditional way of fusing the individual results on the basis of statistics of all units, in which the influence of the worse unit may decrease the final accuracy of the ensemble.

4.3. Statistical results of experiments. In predicting the load pattern for the next day the original input signals for the neural networks were formed by the 24-hour load patterns corresponding to the preceding day, one bit code of the day type and two code bits representing the seasons of the year. In this way the predictors were supplied by 27 input signals, subject to nonlinear processing. The output signals of the predicting units formed the expected 24 hour load pattern for the next day.

The whole data base was split into learning set (the samples representing 70% of the data base) and testing (the remaining data). The learning and testing sets were chosen randomly in each run. The experiments have been repeated 10 times changing randomly the contents of learning and testing sets. The quality of prediction process has been assessed on the basis of prediction error defined as the MAE, MAPE, RMSE and correlation coefficient R, treated in power prediction systems as the most representative factors.

Three neural networks (MLP, RBF and SVM) supplied by the input attributes, have been adapted in introductory learning procedure to obtain their best structure. This was done in some introductory experiments. In all cases we have trained the specialized structure for each hour of the day. In the case of MLP the optimal structure was found as 27–20–1. RBF network used 120 Gaussian neurons (structure 27–120–1). SVM applies adaptive type of learning, in which the number of kernel Gaussian functions are automatically adjusted according to the assumed values of regularization constant C (C = 1000), width σ of Gaussian kernel ($\sigma = 0.9$) and tolerance margin ε (ε = 0.01). The output layer of the learned networks contained only one neuron, representing the load of one particular hour of the day.

The adapted parameters of the members of the ensemble were fixed and the system was ready to predict power consumption for the testing samples not taking part in learning. The analysis of prediction results of the applied units on the set of learning data has shown high diversity of results among the applied predictors. For example for the testing (normalized) vector $\mathbf{x}_t$

$$\begin{aligned}\mathbf{x}_t = [&0.5601\ 0.5436\ 0.5423\ 0.5366\ 0.5419\ 0.5422\\ &0.5779\ 0.6128\ 0.6177\ 0.6156\ 0.6220\ 0.6272\\ &0.6232\ 0.5916\ 0.6003\ 0.6031\ 0.5968\ 0.5826\\ &0.5899\ 0.6831\ 0.6733\ 0.6048\ 0.5739\ 0.5757\ 1\ 0\ 1]\end{aligned}$$

the learning vector $\mathbf{x}_l$, which was closest to this particular testing vector was of the form

$$\begin{aligned}\mathbf{x}_l = [&0.5333\ 0.5286\ 0.5263\ 0.5272\ 0.5370\ 0.5456\\ &0.5715\ 0.6069\ 0.6182\ 0.6216\ 0.6184\ 0.6267\\ &0.6213\ 0.5881\ 0.5946\ 0.5922\ 0.5895\ 0.5811\\ &0.5970\ 0.6852\ 0.6623\ 0.5920\ 0.5621\ 0.5456\ 1\ 0\ 1].\end{aligned}$$

The smallest errors of prediction for the particular hours of the learning vector have been distributed among all three neural networks. Only one, the best unit is applied in final prediction of the load for each hour. As a result the MAPE for this particular testing sample was equal 1.47%, only slightly worse than for the learning sample (MAPE = 1.41%). Similar individual cases analyzed for different days have confirmed this general tendency.

On the basis of performed experiments we have selected 6 learning vectors, which were closest to each testing vector $\mathbf{x}_t$. Thus, the ensemble was formed from 6 units, working in an independent way. For succeeding hours we have used the networks which were the best in learning mode for the actually selected learning vectors. The final forecast is the average of the results of all these 6 predictors.

Table 3 presents the statistical results of experiments for the testing data (not used in learning). They represent the average values of the selected quality measures in predicting the next day 24-hour pattern of load obtained in 10 runs of experiments. The results are related to application of three individual neural networks, their classical integration using the ordinary mean of all three predictors (classical fusion) and our local dynamic fusion (local dynamic). The results of application of the proposed local dynamic integration are the best in all quality measures.

Table 3
The statistical quality measures of predicting the next day 24-hour pattern of loads

Prediction method	MAE [MW]	MAPE [%]	RMSE [MW]	R
SVM	17.34	1.99	35.73	0.988
MLP	15.32	1.79	31.82	0.990
RBF	17.67	2.14	40.60	0.987
Classical average	14.10	1.67	33.15	0.990
Local dynamic	**13.40**	**1.62**	**29.54**	**0.991**

It is difficult to compare these results to other publications, since they refer to different load patterns and their characteristic profiles. These patterns change from year to year or season to season even in the same country, depending on the climate, the actual atmospheric parameters, economic development of the region, size of the power system, size of the country and its population, etc. The exemplary results change a lot, for example the paper [7] declared MAPE = 3.30% for Cyprus, while the paper [9] 1.51% of RMSRE for Shaanxi province of China. As we can see they differ a lot, even in choosing the quality measures.

5. Conclusions

The paper has presented the novel approach to the integration of an ensemble in time series prediction problems. Individual predictor applied in such task leads usually to the solution, which is optimal from the point of view of the applied method, however, not necessarily in the global scale. Ensembles of predictors are well-known answers to such problem by taking advantage of diversity among the models to reduce both the bias and variance components of the prediction error. However, the success in using ensemble depends in large degree on integration of results of the individual units forming ensemble. The classical approaches rely on statistics of prediction errors of the individual predictors obtained in the learning phase. However, it is known that in such methods the final result may be inferior to the best unit of the ensemble.

This paper has proposed the new approach to integration of an ensemble, called the local dynamic method. In contrast to the existing classical methods the final verdict of ensemble is produced by only one member, selected as the best for the particular testing sample. The other, worse units, are simply ignored in this stage. In the prediction task for the particular testing sample different units of the ensemble might win, depending on their local accuracy for this specified sample.

The numerical experiments performed for the prediction of daily average of PM10 pollution as well as forecasting the 24-hour load consumption for the next day have shown the superiority of our approach. All measures of prediction quality have been significantly improved in comparison to either individual predictors or the predictors arranged in the ensemble, but integrated using the classical approaches, based on statistics of learning stage.

Future work will include a larger exploration of this topic by adding more prediction units in an ensemble, each operating on different principle. Further numerical experiments covering wider horizon of data used in prediction process should be also performed.

References

[1] H. Taheri Shahraiyni and S. Sodoudi, "Statistical modeling approaches for PM10 prediction in urban areas; A review of 21st-century studies", *Atmosphere*, 7, 1–24, doi:10.3390/atmos7020015 (2016).

[2] A. Paschalidou, S. Karakitsios, S. Kleanthous, and P. Kassomenos, "Forecasting hourly PM_{10} concentration in Cyprus through artificial neural networks and multiple regression models: implications to local environmental management", *Environ. Sci. Pollut. Res.* DOI 10.1007/s11356–010–0375–2 (2010).

[3] A. Z. Ul-Saufie, S. Yahya, and N. A Ramli, "Comparison between multiple linear regression and feed forward back propagation neural network models for predicting PM10 concentration level based on gaseous and meteorological parameters", *Intern. J. of Applied Science and Technology* 1 (4), 42–49 (2011).

[4] K. Siwek and S. Osowski, "Data mining methods for prediction of air pollution*", International Journal of Applied Mathematics and Computer Science* 26 (2), 467–478 (2016).

[5] G. Gennaro, L. Trizio, A. Gilio, J. Pey, N. Pérez, M. Cusack, A. Alastuey, and X. Querol, "Neural network model for the prediction of PM10 daily concentrations in two sites in the Western Mediterranean", *Science of The Total Environment* 463–464, 875–883 (2013).

[6] A. Daly and P. Zannetti, "Air Pollution Modeling – An Overview", in chapter 2 of "Ambient air pollution" of P. Zannetti, D. Al-Ajmi, and S. Al-Rashied (eds). The EnviroComp Institute, 15–28 http://www.envirocomp.org/ (2007).

[7] M. Marko, E. Kyriakides, and M. Polycarpou, "24-Hour ahead short term load forecasting using multiple MLP", *Intern. Conference on Deregulated Electricity Market Issues in South-Eastern Europe (DEMSEE*), Cyprus (2008).

[8] S. Shah, H.N. Nagaraja, and J. Chakravorty, "Short term load forecasting model for UGVCL, MGVCL, DGVCL and PGVCL using ANN", *International Journal of Recent Trends in Electrical & Electronics Eng*. 5(2):21–30 (2017).

[9] D. Niu, Y. Wang, and D.D. Wu, "Power load forecasting using SVM and ant colony optimization", *Expert Systems with Applications* 37:2531–2539 (2010).

[10] R. Chandra and M. Zhang, "Cooperative coevolution of Elman recurrent neural networks for chaotic time series prediction", *Neurocomputing* 86, 116–123 (2012).

[11] M. Luzar, Ł. Sobolewski, W. Miczulski, and J. Korbicz, "Prediction of corrections for the Polish time scale UTC(PL) using artificial neural networks", *Bull. Pol. Ac.: Tech.* 61(3), 589–594 (2013).

[12] M. Walenczykowska and A. Kawalec, "Type of modulation identification using Wavelet Transform and Neural Network", *Bull. Pol. Ac.: Tech.* 64(1), 257–261 (2016).

[13] S. Osowski, K. Siwek, and R. Szupiluk, "Ensemble neural network approach for accurate load forecasting in the power system", *International Journal of Applied Mathematics and Computer Science*, 19 (2), 303–315 (2009).

[14] L. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms*, Wiley, New York (2004).

[15] L. Xu, A. Krzyzak, and C.Y. Suen, "Methods of combining multiple classifiers and their applications to handwriting recognition", *IEEE Trans. Systems, Man, and Cybernetics* 22 (3), 418–435 (1992).

[16] A.S. Britto, R. Sabourin, and L. Oliveira, "Dynamic selection of classifiers – a comprehensive review", *Pattern Recognition* 47, 3665–3680 (2014).

[17] K. Woods, W.P. Kegelmeyer, and K. Bowyer, "Combination of multiple classifiers using local accuracy estimates", *IEEE Trans. Pattern Analysis and Machine Intelligence* 19 (4), 405–410 (1997).

[18] M. Martuzzi, F. Mitis, I. Iavarone, and M. Serinelli, "Health impact of PM10 and ozone in 13 Italian cities" *WHP report* (2005).

[19] L. Nikias and A. Petropulu, *Higher order spectral analysis,* Prentice Hall, New York (1993).

[20] *Matlab user manual*, *MathWorks*, Natick, USA, (2016).

[21] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection", *J. Mach. Learn. Res*. 3, 1157–1182 (2003).

[22] R.L. Haupt and S.E. Haupt, *Practical Genetic Algorithms*, Wiley, New York (2004).

[23] B. Schölkopf and A. Smola, *Learning with kernels*, MIT Press, Cambridge MA.,USA, 2002.

[24] I. Goodfellow, Y. Bengio, and A. Courville, 2016. *Deep learning,* MIT Press, Cambridge (2016).

[25] P. Romeu, F. Zamora-Martinez, P. Botella-Rocamora, and J. Pardo, "Stacked denoising autoencoders for short-term time series forecasting", in P. Koprinkova-Hristova et al. (eds), *Artificial Neural Networks*, Springer series in Bio-Neuroinformatics 4, 2015, doi: 10.1007/978–3-319–09903–3_23 (2015).

[26] F. Taşpınar, "Improving artificial neural network model predictions of daily average PM10 concentrations by applying principle component analysis and implementing seasonal model", *Journal of the Air & Waste Management Association* 65, 800–809 (2015).