ISSN 1407-8880

TELECOMMUNICATIONS
AND ELECTRONICS
TELEKOMUNIKĀCIJAS UN
ELEKTRONIKA

2009-8880

AN ADVANCED ALGORITHM FOR FORECASTING TRAFFIC LOADS BY NEURAL NETWORKS

INOVATĪVAIS NOSLODZES PROGNOZĒŠANAS ALGORITMS PIELIETOJOT NEIRONU TĪKLUS

I. Klevecka, Mr.Sc.Ing., Doctoral Degree Candidate
Faculty of Electronics and Telecommunications / Riga Technical University
12 Azenes St., Riga LV-1048, Latvia, ph.: +371 26006970, e-mail: klevecka@inbox.lv

Keywords: traffic forecasting, neural networks, algorithm, telecommunications, packet-switched networks, telephone networks

Abstract - Neural networks have been gaining wide popularity in time series forecasting including the prediction of traffic loads. However, there are still some unsolved problems in this filed. The inability to produce and replicate a stable solution, a high probability of over-training and the absence of strictly defined criteria for selecting a final forecasting model are the most substantial of them. An advanced algorithm introduced in the paper addresses some of these problems and specifies a procedure aimed at facilitating the search of the optimal solution and increasing the reliability of produced forecasts.

Introduction

Forecasting the dynamic behavior of network traffic plays an important part in design, deployment and optimization of modern telecommunications networks. Short-term forecasts of network traffic allow avoiding the overload of transmission channels and sustaining the required level of quality of service.

The technology of neural networks is becoming more and more popular in traffic forecasting. In comparison to traditional linear methods (e.g. ARIMA and exponential smoothing), its main advantage is the ability to model the non-linear dynamics of time series and to recognize the chaotic behavior of processes.

Nevertheless, the application of neural networks for forecasting purposes still raises some substantial problems which involve:

- the inability to produce and replicate a stable solution;
- a high probability of over-learning;
- the lack of strictly defined criteria for selecting a final forecasting model ;
- the influence of a network specification on the quality of forecasts.

Besides, in contrast to popular belief, neural networks are sensitive to the quality of input data [1].

Taking that into account, the main aim of the research paper was to propose a functional procedure for specifying a neural network suitable for producing reliable forecasts and to examine its effectiveness for real traffic traces.

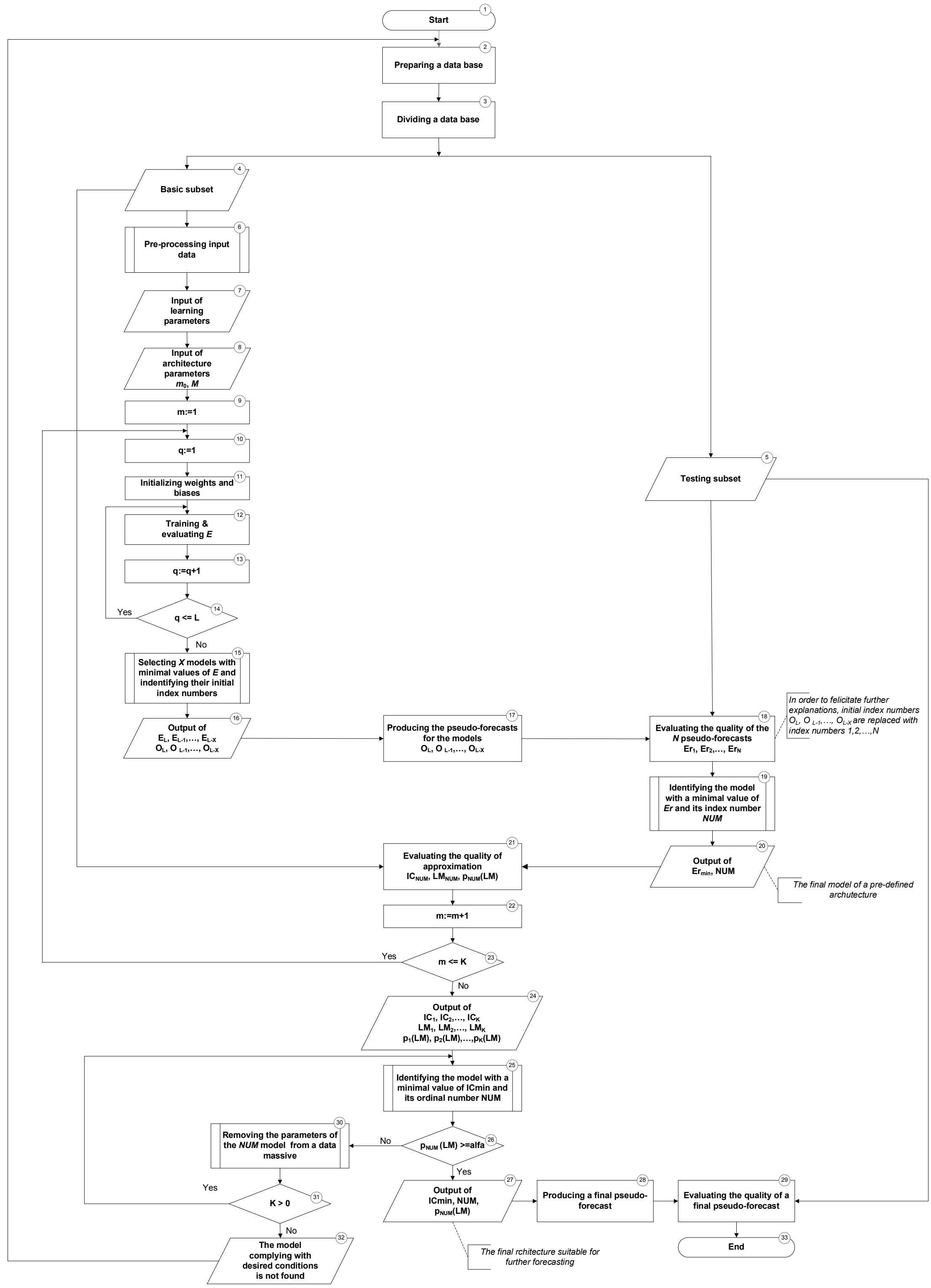
A Brief Description of the Algorithm

The block diagram of the proposed algorithm aimed at solving the task of network traffic forecasting is shown in Fig.1. It is assumed that a special type of neural networks – a time lagged multilayer perceptron [2][3], is used here and that the measurements of network traffic are represented as time series.

In contrast to the classic algorithm of setting up a neural network to solve a pre-defined task (see, for example, [4][5]), the proposed algorithm contains three novel blocks, the detailed description of which is given in the next section

They are:

- implementing multiple cycles of the initialization of weights and biases;
- producing and evaluating the quality of pseudo-forecasts (i.e. out-of-sample testing);
- selecting a final forecasting model taking into account the estimates of information criteria and the level of the autocorrelation of residuals.



Notes

m0 – the size of an input window
M – the number of output neurons
m – the number of hidden neurons
K – the maximum number of hidden neurons
q – the index number of the cycle of initialization of weights and biases
L – the maximum number of initialization cycles
E – training error (evaluated at the end of each training epoch for q=1,2,...L)
O – the initial index number of a model
X – the number of models with a minimal training error E which are chosen for further analysis
Er – the parameter of the quality of a pseudo-forecast
NUM – the index number of a model
IC – information criterion
LM – the Lagrange multiplier type test
p(LM) – the significance level of the the LM type test

Fig.1. An advanced algorithm aimed at solving a forecasting task by means of neural networks

A forecasting task is solved with the assumption that the optimal training parameters as well as an appropriate training algorithm are already pre-defined according to some rules or procedures, and stay invariable during the whole training process.

Such parameters of network architecture as the number of input and output neurons stay unchanged during all the stages of training a neural network and further forecasting as well. The number of hidden layers of a regression network should not exceed one according to the universal approximation theorem [6]. Thus, the optimal value of only one variable parameter, the number of hidden neurons, has to be defined through experimental routine.

Only few hidden neurons are often satisfactory for purposes of modeling and forecasting time series. Therefore, in order to identify the optimal number of hidden neurons, it is advised to apply a constructive method which is implemented in the proposed algorithm. It involves gradual increase of the number of hidden neurons with subsequent testing and verification of each architecture.

The important stage of the algorithm is pre-processing of input data as well. Some procedures of pre-processing are discussed in another research paper of the author [1]. These procedures include reducing a time series to stationary behaviour, identifying seasonal and/ or cyclic components etc.

Neural networks belong to so called heuristic methods. The proposed algorithm is also based on the practical experience of the author and arises from multiple experiments with real time series. However, the background of including these stages into the algorithm strongly relies on the theory of neural networks and time series forecasting.

The Novelty of the Algorithm

Implementing Multiple Cycles of the Initialization of Weights and Biases

It is required to set some initial values to all the weights and biases of a neural network before a training process starts. The aim of initialization is, probably, to find the best approximation to the optimal solution and, in that way, to decrease the time of training by facilitating the convergence of a training algorithm.

If the initial weights and biases are set to large values, then neurons usually approach the saturation level very quickly. In its turn, it would significantly increase training time. If the initial values are set to small values, then the algorithm works very slowly about the origin of the error surface. This is specifically true in the case of anti-symmetric activation function such as hyperbolic tangent.

It is important to understand that regardless of the initialization method, the starting values of weights and biases influence the final result of training. It is directly connected to the peculiarities of training algorithms as initial weights define the direction of the gradient descent in searching a global minimum of the error surface. Therefore, in practice it can be complicated to produce a stable solution and to replicate it.

In order to find an actual global minimum it is necessary to train one and the same architecture multiple times under the same conditions changing only the initial values of weighs and biases.

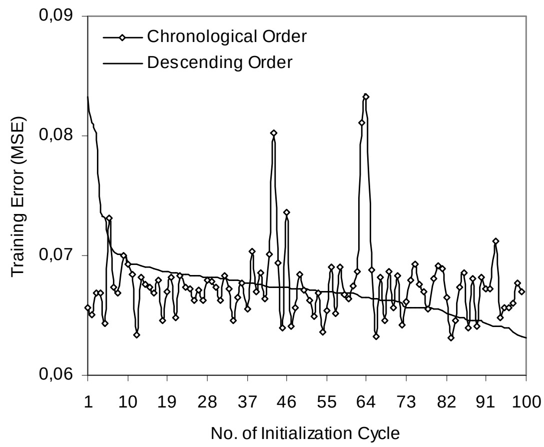


Fig.2. The training errors of a neural network (without applying a cross-validation) produced for a hundred cycles of initialization of weights and biases

The example shown in Fig.2 illustrates an uncertainty in producing a final solution and its dependence on the starting values of weights and biases being set in the process of initialization. The values of training errors (in this particular case – mean square errors) are displayed in chronological order of their evaluation at the end of each training epoch as well as sorted in descending order to facilitate their comparison.

The training errors shown here are produced as a result of training a neural network of the same architecture and applying the same training parameters but changing the initial values of weights and biases. It is easy to notice that the difference between the largest and smallest errors is rather prominent what can be significant for the choice of a forecasting model and the quality of the produced forecasts.

Despite these complications, many researchers still do not pay a proper attention to this issue. It is possible to find references to 50 [7], 10 [8], 5 [9] cycles of initialization. However, in most research papers only one initialization cycle is applied what can create the illusion that a fitted model is adequate even if it is not.

During the last two decades many different heuristic methods of initialization have been proposed, the main aim of which is to improve the convergence of the training algorithm of a neural network. Despite this, the optimal solution has not been still found. The most popular method of initialization is the random initialization of weights and biases from a uniform distribution with a narrow range of small values. The method was first suggested in a fundamental work [10].

The number of initialization cycles is usually chosen mandatory, and depends on the complexity of a task as well as on time resources a researcher has on his / her disposal.

Producing and Evaluating the Quality of Pseudo-Forecasts

Out-of sample testing allows approaching the conditions of real-life forecasting. It is a common method in time series forecasting (see, for, example, [11][12]) but it has not still gained a wide acceptance in predicting the dynamic behaviour of network traffic.

In general, out-of-sample testing involves splitting a time series into two independent subsets. The first, largest data subset is used for the selection and verification of a statistical model. It is called a basic or retrospective subset. The second, testing subset is used for examining the quality of so called pseudo-forecasts (*ex post* forecasts) comparing them with historical data. It provides the opportunity to evaluate the forecasting abilities of the model fitted to the basic subset.

In the case of linear statistical models, pseudo-forecasts are produced after the model is fully verified, i.e. the quality of approximation and the autocorrelation of the residuals are already tested. However, it is important to keep in mind that non-linear neural models are inclined to over-learning, i.e. to the simple memorization of input data. An over-trained neural network has poor generalization ability, and consequently, a forecasting ability even if the quality of approximation is high.

Different initial values of weights and biases result in different final solutions, and consequently, in different values of training errors. Already at this stage, the proposed algorithm involves selecting some models with a minimum training error and testing the quality of the pseudo-forecasts produced for each of them. This approach allows identifying the model which is suitable for further forecasting *in advance* and avoiding a common situation when a final fully-verified architecture with a high quality of approximation produces very poor forecasts.

The last historical values of an analyzed time series are traditionally used for developing a testing subset. However, it is necessary to understand that these observations influence the direction of a further forecast much more than the earlier ones. Therefore, the last historical data are the most valuable in the process of selecting a relevant forecasting model, and using them as a testing subset is not always reasonable.

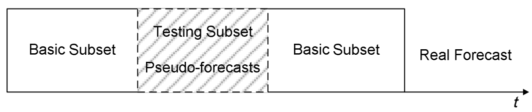


Fig.3 Chronological division of a time series into the training and testing subsets in order to test the quality of a pseudo-forecast

Implementing a proposed algorithm it is recommended to divide a time series into training and testing subsets according to the scheme shown in Fig.3. This original approach allows increasing the quality of real forecasts as the last historical observations are not used for testing but for fitting a statistical model.

Selecting a Final Forecasting Model

It is advised to select a final forecasting model (architecture) taking into account two

parameters – the value of an information criterion and the level of the autocorrelation of residuals.
Some standard parameters such as the correlation coefficient (R), mean square error (MSE), mean absolute error (MAE) etc. are traditionally used for the evaluation of the general forecasting ability of a statistical model. However, the relevance of these classic criterions has raised some arguments. The mean square residuals are usually decreasing once the model is becoming more „complicated" with addition of new free parameters. In particular, for neural networks this issue refers to the necessity of searching the optimal values of extra weights and biases.
However, once a certain limit is reached, the gain in quality of fitting with addition of new parameters tends to be insignificant. On the other hand, time required for selecting the optimal values of free parameters can lead to a sharp decrease in the performance of a network. Therefore, it is very important to look for the optimum between the preciseness of approximation and the complexity of a statistical model.
Information criteria have been found to be quite useful in solving this problem. The estimate of the criterion consists from the penalty for poor fitting and the penalty for over-parameterization.
The first suggested criterion of this type is the Akaike's information criterion (AIC) given by [13][14]:

$$AIC(k) = \tau \ln \hat{\sigma}_{\varepsilon}^{2} + 2k \tag{1}$$

where
τ – the number of observations to which the model is fitted;
k – the number of adjusted parameters;
$\hat{\sigma}_{\varepsilon}^{2} = \frac{\sum_{t=1}^{\tau} \hat{\varepsilon}_t^2}{\tau}$ – estimate of the residual variance (i.e. mean square error).

Another popular information criterion is the Bayesian modification of the Akaike's criterion – the Bayesian information criterion (BIC) [13][14]:

$$BIC(k) = \tau \ln \hat{\sigma}_{\varepsilon}^{2} + 2k + k\ln(\tau) \tag{2}$$

The BIC puts a stronger penalty on over-parameterization as compared to the AIC. Therefore, it "selects" the models of smaller dimension.
Information criteria are calculated separately for each analyzed specification (architecture) of the model. The model that possesses the minimum value of the criterion should be selected for further analysis. It has been also noticed that in practice the BIC "selects" very parsimonious models with only few parameters. Therefore, this criterion is often used for non-linear models where insignificant gain in fitting quality is directly connected to the necessity of calculating a large number of additional parameters. Some other modifications of information criteria were proposed as well. The Schwarz's Bayesian Criterion (SBC) and the information criterion of Hannan and Quinn are among them.
Another important aspect of selecting an adequate forecasting model is the lack of autocorrelation in residuals. The residuals of the produced model are defined as *n* differences given by $\varepsilon_t = x_t - \hat{x}_t, t = 1, 2, \ldots, n$, where x_t is the observed value and $\hat{x}_t$ is a corresponding predicted value produced by means of a fitted statistical model. These differences cannot be explained by a produced forecasting model. Therefore, we can consider residuals ε_t as observed errors.
The model is relevant if only the hypothesis about the presence of autocorrelation in residuals is rejected at the pre-defined significance level. It means that the residuals are similar to a white noise and further analysis will not discover any statistically significant dependencies.
The Durbin-Watson criterion [15] is traditionally used for testing the autocorrelation of residuals in classic regression analysis. However, this test is not suitable for testing the autocorrelation of residuals if the regressor is a lagged explanatory variable [16]. On the account of the same reason, the Box-Pierce and Ljung-Box [17] criterions cannot be applied to neural networks even the last one is widely used, and despite its theoretical inconsistence, is still included in most statistical and econometric software packages.
The most relevant estimate of the presence or lack of autocorrelation in residuals of neural networks gives a powerful Lagrange Multiplier (LM) type test (also called the Breusch-Godfrey test) introduced in [16][18]. The LM type test belongs to classic asymptotic tests and serves for indentifying the autocorrelation of mandatory order.

Conclusion

The proposed algorithm was applied to sixteen time series of different length and aggregation degree. The time series represent actual measurements of the traffic of both telephone and packet-switched IP-networks.

The experiments proved that the application of the algorithm to solving the task of traffic forecasting allows facilitating the search of the optimal solution and increasing the quality of produced forecasts. The algorithm provides a systematic approach to searching an adequate solution and allows avoiding a common situation when the model with good approximation quality is not suitable for further forecasting due to over-training.

The algorithm can be applied to other time series with similar statistical properties as well, for example, in forecasting the level of energy consumption.

References

1. Klevecka, I., and J. Lelis. "Pre-Processing of Input Data of Neural Networks: The Case of Forecasting Telecommunication Network Traffic." *Telektronikk* 3/4 (2008): 168-178.
2. Haykin, S. *Neural Networks: A Comprehensive Foundation* (2nd Ed.). Prentice Hall, 1998.
3. Mozer, M. C. "Neural Network Architectures for Temporal Pattern Processing." *Time Series Prediction: Forecasting the Future and Understanding the Past* Eds. A.S. Weigend and N. A. Gershenfeld. Perseus Book Publishing, 1993. 243-264.
4. Fildes, R. and K.-P. Liao. "The Accuracy of Procedural Approach to Specifying Feedforward Neural Networks for Forecasting." *Computers & Operations Research* 32.8 (August 2005): 2151-2169.
5. Комарцова Л., Максимов А. Нейрокомпьютеры: Учеб. пособие для вузов. – М.: Издательство МГТУ имени Н.Э. Баумана, 2002. – 320 с.
6. Cybenko, G. "Approximation by Superpositions of Sigmoidal Function." *Mathematics of Control, Signals, and Systems* 2 (1989): 303-314.
7. Faraway, J., and C. Chatfield. "Time Series Forecasting with Neural Networks: A Comparative Study Using the Airline Data." *Applied Statistics* 47.2 (1998): 231-250.
8. Tang, Z., and and P.A. Fishwick. „Feedforward Neural Nets as Models for Time Series Forecasting." *ORSA Journal on Computing* 5.4 (Fall 1993): 374-385.
9. Kaastra, I., and M. Boyd. "Designing a Neural Network for Forecasting Financial and Economic Time Series." *Neurocomputing* 10 (1996): 215-236.
10. Rumelhart, D.E., J.L. McClelland, and the PDP Research Group. *Parallel Distributed Processing*, Vol. 1: Foundations. Cambridge: The MIT Press, 1986.
11. Tashman, L.J. "Out-of-sample Tests of Forecasting Accuracy: An Analysis and Review." *International Journal of Forecasting* 16.4 (October 2000): 437-450.
12. Makridakis, S. and R.L. Winkler. "Sampling Distributions of Post-sample Forecasting Errors." *Applied Statistics* 38.2 (1989): 331-342.
13. Priestley, M.B. *Spectral Analysis and Time Series*. London: Academic Press, 1981.
14. Franses, H.P., and D. van Dijk. *Nonlinear Time Series Models in Empirical Finance*. Cambridge University Press, 2000.
15. Дрейпер Н., Смит Г. Прикладной регрессионный анализ, 3-е изд.: Пер. с англ. – М.: Издательский дом «Вильямс», 2007. – 912 с.
16. Thomaidis, N.S., and G. Dounias. "A Comparison of Statistical Tests for the Adequacy of a Neural Network Regression Model." January 2007. <http://papers.ssrn.com/sol3/papers.cfm?abstract_id=957892>.
17. Ljung, G.M., and G.E. Box. "On a Measure of Lack of Fit in Time Series Models." *Biometrika* 65 (1978): 297–303.
18. Medeiros, M.C., T. Teräsvirta and G. Rech. "Building Neural Network Models for Time Series: A Statistical Approach." *Journal of Forecasting* 25(2006): 49-75.

I. Kļevecka. Inovatīvais noslodzes prognozēšanas algoritms, pielietojot neironu tīklus

Neironu tīklu izmantošana telekomunikāciju tīklu dinamiskās uzvedības prognozēšanā iegūst lielu popularitāti. Tie rada papildus iespējas nelineāro parādību modelēšanā un haotiskās uzvedības noteikšanā. Tomēr, neironu tīklu pielietošana laika rindu prognozēšanā joprojām ir saistīta ar būtiskām problēmām, tādām kā sarežģīta stabilo risinājumu iegūšana un to atkārtošana, tīkla augsta pārmācības varbūtība un izvēlēto gala

prognozēšanas modeļa (arhitektūras) precīzo vērtēšanas kritēriju trūkums. Darbā piedāvātais algoritms risina dažas no nosauktajām problēmām. Atšķirībā no klasiskām risināšanas shēmām, piedāvātais algoritms iekļauj trīs principiāli jaunus blokus, saistītos ar:

- *neironu tīkla svaru un sliekšņu daudzkārtējās inicializācijas ieviešanu;*
- *pseidoprognozes radīšanu un novērtēšanu, izvēloties prognozēšanas modeli;*
- *gala prognozēšanas modeļa izvēli, novērtējot informācijas kritērijus un atlikuma autokorelācijas līmeni.*

Algoritms ļauj atvieglot optimālā risinājuma meklēšanu, paaugstināt prognožu kvalitāti. Algoritms piemērots arī citām laiku rindām ar līdzīgām īpašībām, piemēram, enerģijas patēriņa līmeņa prognozēšanā.

I. Klevecka. An Advanced Algorithm for Forecasting Traffic Loads by Neural Networks

The mechanism of neural networks is gaining more and more acceptance in forecasting the dynamic behaviour of telecommunications networks. Neural networks provide additional opportunities in modeling non-linear phenomena and recognizing chaotic behavior of processes. However, the application of neural networks to time series forecasting still raises some substantial problems. The inability to produce and replicate a stable solution, a high probability of over-training and the absence of strictly defined criteria for selecting a final forecasting model are among them. An advanced algorithm introduced in the paper is aimed at solving some of these problems. In contrast to the classic algorithm of setting up a neural network to solve a pre-defined task, this algorithm contains three novel blocks which involve

- *implementing multiple cycles of initialization of weights and biases;*
- *producing and evaluating of the quality of pseudo-forecasts;*
- *selecting a final forecasting model taking into account the values of information criteria and the level of the autocorrelation of residuals.*

The algorithm allows facilitating the search of the optimal solution and increasing the reliability of produced forecasts. It can be applied to other time series with similar statistical properties as well, for example, in forecasting the level of energy consumption.

И. Клевецкая. Новаторский алгоритм прогнозирования сетевой нагрузки посредством нейронных сетей

Механизм нейронных сетей получает все большее признание в области прогнозирования динамического поведения телекоммуникационных сетей. Нейронные сети предоставляют дополнительные возможности в моделировании нелинейных явлений и распознавании хаотического поведения. Тем не менее, их применение в области прогнозировании все еще сопряжено с существенными проблемами, среди которых можно выделить сложности в получении и повторении стабильного решения, высокую вероятность переобучения и отсутствие четко определенных критериев для выбора конечной модели прогнозирования. Предложенный в работе алгоритм направлен на решение некоторых из этих проблем. В отличие от классической схемы настройки нейронной сети на решение задачи, данный алгоритм содержит три принципиально новых блока, связанных с

- *введением множественных циклов инициализации весов и порогов;*
- *построением и оценкой псевдопрогноза при выборе модели прогнозирования;*
- *выбором конечной модели прогнозирования, принимая во внимание показатели информационных критериев и уровень автокорреляции остатков.*

Алгоритм позволяет упростить поиск оптимального решения и повысить достоверность получаемых прогнозов. Алгоритм может применяться и к другим временным рядам со схожими статистическими свойствами, к примеру, при прогнозировании уровня энергопотребления.