

Pre-Processing of Input Data of Neural Networks: The Case of Forecasting Telecommunication Network Traffic

IRINA KLEVECKA, JANIS LELIS



Irina Kleveckā is a PhD candidate at Riga Technical University (Riga, Latvia)



Janis Lelis is Executive Director of the Latvian Telecommunications Association (Riga, Latvia)

The aim of our research was to create a functional algorithm of preprocessing of input data taking into account the specific aspects of teletraffic and properties of neural networks. The practical application to forecasting telecommunication data sequences shows that the procedure of data preprocessing decreases the time of learning (what is particularly important in the case of large data sets) and increases the plausibility and accuracy of the forecasts. The algorithm can be applied to forecasting the intensity of plain telephone networks and IP networks.

1 Introduction

The prediction of network traffic plays an important role in designing, optimization and management of modern telecommunications networks. Network traffic is a non-linear time series generated by a complex dynamic system, and incorporating both stochastic and deterministic components.

The algorithm of neural networks is considered to be one of the most appropriate methods for the prediction of future behavior of such complicated processes. Neural networks are computational methods inspired by the human brain and nerve system. They are a class of non-linear and non-parametric statistical models. Their main advantage over the traditional statistical methods lies in the fact that they can flexibly model complex non-linear relationships without prior knowledge about the nature and statistical properties of the analyzed data set.

The special type of feed-forward neural networks, so-called time-delayed neural networks (TDNN), are applied here for analyzing and forecasting time series (Figure 1.1a). In a feed-forward neural network (also called a multilayer perceptron) all the processing units (the neurons) are arranged in layers. This type of neural networks is often denoted as $I \times H \times O$,

where I – the number of input units, H – the number of hidden units and O – the number of output units. Each unit consists of layers which receive their input from units from a layer directly below and send their output to units a layer directly above the unit. There are no connections within a layer [2].

A TDNN is stimulated through a short-term memory. In the case of only one variable (and it is often the case of telecommunication traffic) the input vector consists of p past values $x_{t-1}, x_{t-2}, \dots, x_{t-p}$ stored in a delayed line memory of order p , and the free parameters of the neural network are adjusted to minimize the mean-square error between the output of the network x'_t , and the desired response x_t . The multilayer perceptrons rear have more than one hidden unit as it has been proved that only one layer of hidden units suffices to approximate any function with finitely many discontinuities to arbitrary precision (this rule is known as the *universal approximation theorem*). Increasing the number of hidden layers (as well as the number of hidden neurons) without any necessity we can face a serious problem – over-learning. As a result of that, network does not extract the characteristic features of the data but simply memorize them along with all the noises.

The diagram of a single neuron is shown in Figure 1.1.b. The neuron receives input signals (ie. inputs weighted by weights) from each neuron of the previous layer. Additionally it receives so-called bias input x_0 with weight b . The activation function $f(\Sigma)$ of a neuron transforms the sum of all the weighted inputs into the output signal x' which serves as an input signal for all the neurons of the next layer. Here the output neuron is assumed to be linear and the activation function of hidden neurons is usually sigmoidal.

In the theory of neural networks the procedure of weight adjustment is called *training*. The back-propagation is a traditional method of training of multilayer perceptrons but some other second-order optimization algorithms (eg. the Lavenberg – Marquardt algo-

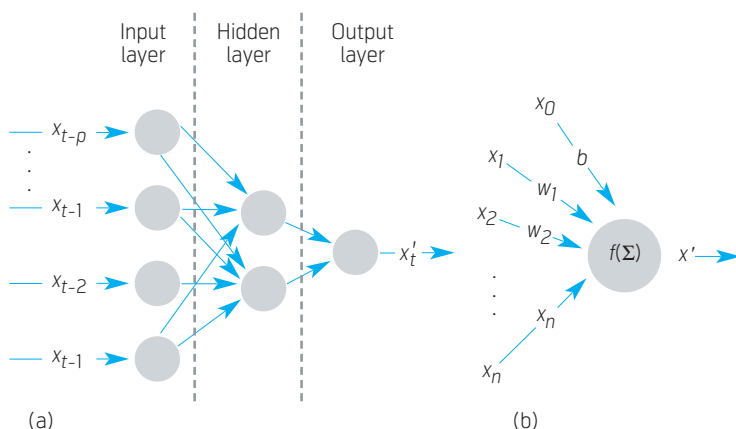


Figure 1.1 (a) Time-delayed neural network, (b) Single artificial neuron

rithm, the conjugate gradient descent etc) can also be applied with success. During one training phase, which is usually reported as an *epoch*, all the input patterns are processed one by one through layers of neurons. The number of patterns is determined by the memory order p and the size N of the training sample.

For the more detailed information on designing and training of neural networks we refer the reader to [2] [10].

Despite the fact that neural networks have been widely applied to solving the tasks of time-series analysis, pattern recognition and pattern classification, it is important to keep in mind that the presentation of data set and its statistical properties influence the results of forecasting not less than the selection of an appropriate network architecture and method of learning. At present the information on prior processing of input/output data of neural networks is incomplete and inconsistent. Therefore, among the primary tasks of this research was to create a functional algorithm of pre-processing of input data in order to facilitate the training process and to increase the plausibility of produced forecasts.

2 The Algorithm of Pre-Processing of Input Data

In order to facilitate the training procedure and to increase the performance of neural networks we suggest applying a five-step algorithm of pre-processing of input/output data. The stages of the algorithm include the estimation and replacement of outliers, testing for stationarity, the evaluation of seasonal components, testing for normality and scaling to the range of reasonable values of the activation function.

2.1 Estimation of Outliers and Atypical Patterns

Outliers are atypical, infrequent observations which do not appear to follow the statistical characteristics of the rest of the data. Neural networks are noise tolerant. However, there is a limit to this tolerance. If there are occasional outliers far outside the range of normal values for a variable, they may bias the training. The best approach to such outliers is to remove or convert them.

Some methods which are commonly used for identification of outliers assume that the data are from a normal distribution, and identify observations which are deemed 'unlikely' based on mean and standard deviation. However, some limitations are applied to these methods. First of all, all deterministic components – a trend and cyclic variations should be subtracted prior to the analysis. In the second place, we have to be

absolutely sure that the distribution of our variable is normal. Otherwise, the produced results might be incorrect.

In order to identify outliers of non-linear data we suggest implementing the robust Tukey's 53H procedure. The Tukey's 53H procedure uses the principle that the median is a robust estimator of the mean to generate a smooth time series that can be further subtracted from the original time series. This 53H procedure consists of several passes of moving average/median smoothing and incorporates the following steps [15][19]:

1. Construct a sequence x'_t from the median of the five data points from x_{t-2} to x_{t+2}

2. Construct a sequence x''_t from the median of the three data points from x'_{t-1} to x'_{t+1}

3. Construct the Hanning smoothing filter:

$$x'''_t = \frac{1}{4}x''_{t-1} + \frac{1}{2}x''_t + \frac{1}{4}x''_{t+1} \quad (3.1)$$

4. Construct the sequence $\Delta_t = |x_t - x'''_t|$ and reject the point if $\Delta_t \geq k$, where k is a predetermined threshold (usually set between 3 and 9).

Even the Tukey's 53H procedure can be fully automated, a manual analysis is still desirable. Determining whether or not an observation is an outlier is ultimately a subjective exercise. It may reflect genuine properties of the underlying variable, or be due to measurement errors or other anomalies. However, obvious outliers which appear due to occasional damages of a network should be obligatorily removed.

Besides, a time series can incorporate atypical patterns which occur, for example, when a national holiday falls on a regular weekly workday. That usually produces irregular high or, vice versa, low intensity of traffic. Such patterns lead to the deterioration of the measures of the goodness of fit and should be replaced by the more regular ones.

2.2 Testing for Stationarity

TDNN have some difficulties in modelling and predicting non-stationary time series. Therefore it is particularly important to stabilize both the mean and variance of the data set prior to the training procedure. The distribution of a time series representing telecommunication traffic is usually far from normal. In order to verify the stability of mean and variance over time we suggest here to apply non-parametric (distribution-free) tests based on the records (ranks) of data sequence.

One such easy procedure which is useful for detecting underlying trends in data records is the reverse arrangement test. The intensity of telecommunication traffic is usually represented by the long-range time sequences. Therefore, it is suggested to split a time series into equal groups prior to the analysis. The procedure incorporates the following steps [1][11]:

1. Split a time sequence into N equal successive time intervals.
2. Estimate mean and variance of each of the intervals and align these sample values in time sequence (ie. two new time sequences are produced).
3. Count a reversal every time then $x_i^* > x_j^*$ for some i and j with $i < j$ for both mean and variance:

$$h_{ij} = \begin{cases} 1, & x_i > x_j \\ 0, & x_i < x_j \end{cases} \quad (3.2)$$

4. The reversal count is the total number of reversals A according to the rule:

$$A = \sum_{i=1}^{N-1} A_i \quad A_i = \sum_{j=i+1}^N h_{ij} \quad (3.3)$$

5. The null hypothesis that data sequence is stationary in mean and variance is tested at the α significance level. The table of critical limits of the number of reversals depending on the number of intervals (data points) can be found in [1].

Some other methods of this type might be also recommended – the runs test [1], the Foster-Stuart test [7] and quick sign tests [5].

Further analysis is required if the hypothesis is rejected. The presence of a trend component is one of the reasons of non-stationarity in mean. In the case of long time series the trend component of time series is its long-term feature that is manifested through the local or global increase or decrease of data values as a consequence of superposition of true time series values and a disturbance with upward or downward trend [20]. To identify the general character of the trend present, the analysis is accomplished using linear or non-linear (polynomial, exponential, etc) regression technique. Further subtraction of the trend component usually reduces the data to stationarity in mean.

If the test reveals that a time-series is non-stationary in variance, the Box-Cox power transformations are very useful in this case. They stabilise variance and also make a distribution of time series more normally distributed reducing the influence of 'fat tails'.

2.3 Identification of Seasonal Components

The seasonal component, often referred to as seasonality, is a characteristic of a time series in which the data experiences regular and predictable changes which recur every calendar year. Seasonal effects are different from cyclical effects which can span time periods shorter or longer than one calendar year.

The network traffic usually incorporates daily, weekly and annual cycles. Periodic components can be easily identified by applying the spectral Fourier analysis [3]. The cyclic/seasonal components can be expressed as the sum of polyharmonic functions with different periods taking into account the coefficients of the Fourier decomposition.

The deseasonalization (ie. the subtraction of seasonal components) of input data is still the point at issue (see, for example, [18] [25]). Some authors insist on the opinion that a neural network can simulate all the seasonal effects and prior deseasonalization is not necessary. However, our research produces the arguments in defense of deseasonalization in the case of predicting future development of teletraffic. In our empirical case the subtraction of a seasonal component gave a possibility to decrease the number of input neurons and, consequently, to decrease the time of training.

2.4 Testing for Normality

One of the commonly mentioned advantages of neural networks is that they can work effectively with non-normally distributed data. However, it is still the issue under discussion as some papers state the opposite point of view (see, for example, [4][9]). Further research is required, but we are sure that normality of input data is desirable.

First of all, in the case of non-normality the estimates of mean and variance are distorted. Therefore, these estimates should not be used to make inferences about central tendency and data spread, or to produce confidence and/or prediction intervals about the central value of the data distribution. Another aspect refers to the standard requirement of the normality of residuals. However, if the raw data significantly diverse from a normal distribution, we cannot expect the normality of residuals as well.

In order to test the hypothesis whether the distribution of a variable comes from a normal distribution we suggest applying a non-parametric chi-square (Pierson's) criterion. It is one of the most widely used statistical criteria; therefore we omit the description.

When data are not normally distributed, it is advisable to determine the cause of non-normality and to take the appropriate remedies. The already mentioned

procedures – the elimination of outliers, the Box-Cox transformations and the subtraction of cyclic components are such remedial actions that may help to make data normal. Of course, it is not possible to reduce the distribution to normality in all the cases but we should try to make it as normal as possible.

2.5 Normalization / Scaling

It is important that the target values (desired response) be chosen within the range of the sigmoidal activation function [10]. In order to simplify this procedure, usually all the input and output variables of multilayer perceptrons are scaled to the range of reasonable values. This procedure is called the normalization.

This is the only stage which has to be applied to data sequence irrelatively of other pre-processing procedures. Without scaling, a great deal of information from the data is likely to get lost, since the neurons will not be able to recognize the data outside the limits of its activation function. There are several methods to scale input/output data into the range of acceptable values. The most popular of them are linear scaling and standardizing of time series.

The linear scaling function makes use of the maximum and minimum values of the series. It transforms a variable in the following way:

$$\tilde{x}_t = \frac{(x_t - x_{t,\min})(b - a)}{(x_{t,\max} - x_{t,\min})} + a \quad (3.4)$$

where $[a, b]$ – target values within the range of the activation function;

$[x_{t,\min}; x_{t,\max}]$ – the minimum and maximum values of the variable;

\tilde{x}_t – normalized (scaled) variable.

The target values should be within the range of the sigmoid, but not at the asymptotic values. On the other hand, care must be taken to ensure that the node is not restricted to only the linear part of the sigmoid. It is recommended to set target values to the point of the maximum second derivative of the sigmoid that takes advantage of the nonlinearity without saturating a sigmoid [12].

Standardizing procedure involves transforming of input / output variables according to a standard normal distribution. Besides scaling, this method maximizes the entropy of input data making the distribution of the input data more uniformly distributed. However, this kind of transformation does not obligatorily ensure that all the data points fall into the range of the sigmoid. Therefore, it is suggested to scale data in a two-step procedure. The first step is standardiz-

ing a time series, and the second step is taking the sigmoid transformation of the standardized data [16]:

$$z = \frac{x_t - M(x_t)}{\sigma(x_t)}; \quad \tilde{x}_z = \frac{1}{1 + e^{-z}} \quad (3.5)$$

Which method of scaling to choose depends on the statistical characteristics of data, quality of results and prior pre-processing. There is no universal way to decide which scaling function works best, on a priori grounds, given the features of the data. The best strategy is to estimate the model with different types of scaling and to find out which one gives the best performance, based on in-sample criteria [16].

3 Empirical Case

3.1 The Description of Forecasting Methodology

The above described algorithm was applied to pre-processing and further forecasting of the outgoing international traffic of the IP network. Unlike the traffic of plain telephone networks, the IP traffic is a fractal (self-similar) process and shows strong long-range dependence [13][22]. Therefore, the production of reliable forecasts of the IP traffic usually creates difficulties, and this example is especially interesting from both theoretical and practical points of view.

The traffic was measured every 15 minutes during fourteen weeks. The time series consists of 8736 data points, 8064 of which were used for training and selection of the appropriate forecasting model (in-sample evaluation), and 672 latest data points were held out in order to compare the genuine forecasts with historical data points (out-of-sample evaluation).

Different structures of a multilayer perceptron were implemented. The number of input neurons was equal to the largest period of a cyclic component. The intensive test-and-trial routine was implemented in order to identify the relevant number of hidden neurons, as a reliable mathematical rule for its determining is still not discovered. All the architectures with the number of hidden neurons varying from one to nine have been tested and verified.

The initial weights were drawn from a uniform distribution. It is worth noting that the starting weight values affect the eventual results achieved. Therefore it is necessary to reinitialize the network at least several times to choose the best available configuration, or, to form an ensemble from those networks which perform better than others. In our case each network architecture was reinitialized a hundred times, and the best available model of each architecture was selected for further analysis.

In order to avoid the effect of over-learning, the principle of cross-validation was implemented as well [2]. Random splitting into training, selection and test subsets was applied for each configuration.

A two-stage training process was implemented. During the first stage a multilayer perceptron was trained by applying the backpropagation during one hundred epochs, with learning rate 0.1 and momentum 0.3. It usually gives the opportunity to locate the approximate position of a reasonable minimum. During the second stage, a long period of conjugate gradient descent (several thousand epochs) is used, with a stopping window of 50, to terminate training once convergence stops or over-learning occurs. However, in real applications training usually stops after a couple of hundred epochs. Once the algorithm stops, the best network from the training run is restored.

Weight regularization [24] was applied extensively in multilayer perceptron training. Pruning of both inputs and hidden units was performed based on weights magnitude.

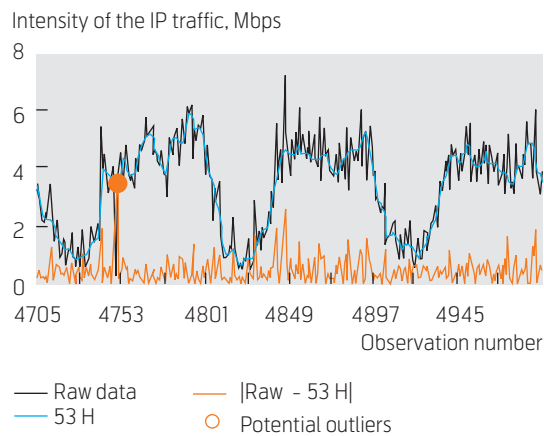


Figure 3.1 Applying the Tukey's 53H procedure to the identification of potential outliers

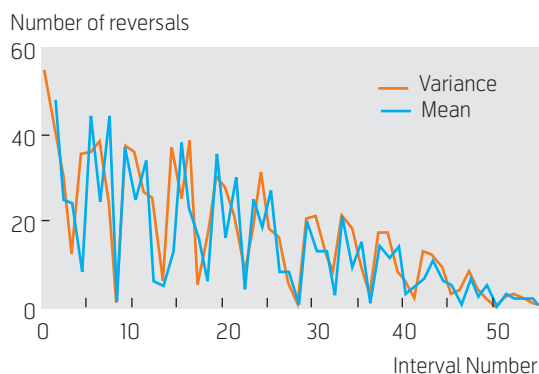


Figure 3.2 The number of reversals of mean and variance

3.2 The Evaluation of Adequacy of Forecasting Models

The estimation of the adequacy of forecasting models involves four stages – the evaluation of the quality of training, in-sample evaluation of the goodness of fit, diagnostic testing of residuals and out-of-sample evaluation of the quality of forecasts.

• Evaluation of the Quality of Training

S.D. ratio characterizes the performance of the networks on the training, selection and test subsets respectively. For a regression network, the S.D. ratio is simply the ratio of the prediction to observation standard deviations.

The sum-squared error rate (E) is the sum of the squared differences between the target and the actual output values on each output unit. This is less interpretable than the performance measure but is the figure actually optimized by the training algorithm (at least, for the training subset).

The best available model of each architecture is selected taking into account both the S.D. ratio and training error evaluated on the selection subset.

• Evaluation of the In-sample Fit

During the second stage the standard measures of the goodness of fit (the quality of approximation) are evaluated. They are the coefficient of correlation (R), the mean absolute error (MAE), the mean square error (MSE) and the mean absolute percentage error (MAPE). The definitions and formulas are omitted here for the sake of space saving.

Special attention should be paid to the application of information criteria. The information criteria combine a measure of fit and the penalty term to account for model complexity. They are very helpful in selecting the appropriate architecture of neural networks. The two most popular of them are the Akaike information criterion (AIC) and the Bayesian information criterion (BIC) [8][21].

In the case of two equivalent models, with both having acceptable residuals, the one having a lower value can be taken as the better one. The BIC penalizes additional parameters more heavily than the AIC. Therefore, the model order selected by the BIC is likely to be smaller than the one selected by the AIC. In practice, the BIC prefers very parsimonious models. This has implications for the use of this criterion in evaluating nonlinear time series models such as neural networks, where sometimes quite a large number of parameters is needed to obtain only a slightly improved fit [8].

- **Diagnostic Testing of Residuals**

If the model is well specified, then there should be nothing further to learn from the residuals. The residuals should be simply uncorrelated meaningless information. Testing the adequacy of an estimated model usually involves examining whether the auto-correlations of the residual series are equal to zero. If it turns out not to be the case, there is a need to modify the model. It has been recently proved that the commonly applied methods to test the residual auto-correlation – such as the Durbin-Watson test [6] or the Box-Ljung test [14] – are not suitable for non-linear neural networks. In the case of neural networks it is recommended to follow the Lagrange Multiplier (LM) principle which is described in detail in [17][23].

Moreover, residuals are usually assumed to come from a normal distribution. For testing the null hypothesis we typically use a chi-square test. Rejection of normality may indicate that there are outliers, that the error process is not homoscedastic, and/or that the selected model should be reconsidered. The violation of normality assumption is not as serious as the violation of the absence of autocorrelation [14]. However, if the model is going to be used for simulating models subject to random normal disturbances, then it would be good to have normal randomly distributed residuals in the estimated models. Besides, the normality is required for applying the *t*-statistic to construct the confidence intervals.

- **Evaluation of Out-of-Sample Forecasts**

In order to evaluate the quality of produced forecasts three standard statistical parameters – the mean absolute error (MAE), the mean square error (MSE) and the mean absolute percentage error (MAPE) – were estimated for each model of neural networks.

3.3 Practical Implementation of the Algorithm

- **Identification of outliers and atypical patterns**

The application of the Tukey's 53H procedure to identifying the anomalous outliers of the IP traffic is shown in Figure 3.1. (the coefficient *k* was set to 3). The Tukey's procedure has revealed three potential outliers of the time series. The spurious data points were replaced by the interpolation of adjacent values. None of irregular patterns were revealed.

- **Testing for stationarity**

Figure 3.2 illustrates the application of the reverse arrangement test. The data sequence was divided into 56 equal subsets.

The mean, variance and the number of reversals were estimated for each of them. The whole number of

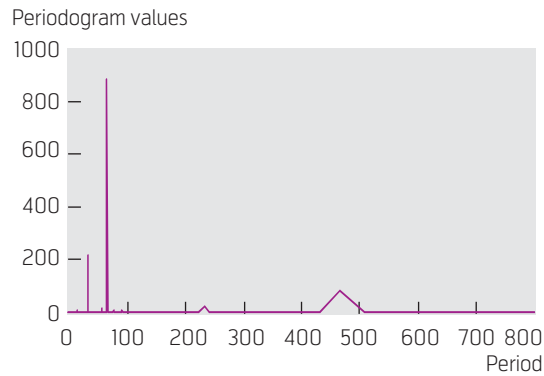


Figure 3.3 The periodogram of a time series

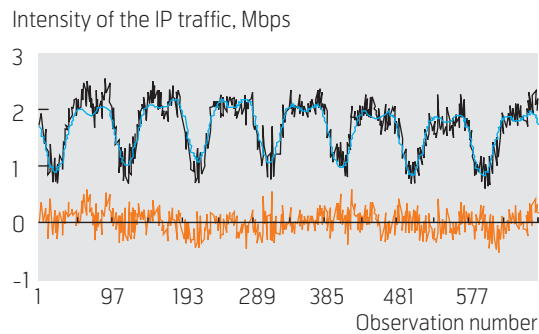


Figure 3.4 Input data after the subtraction of a trigonometric trend

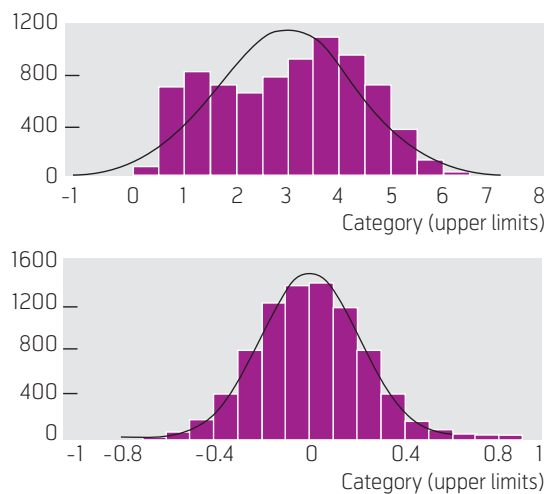


Figure 3.5 Testing the normality of input data before and after pre-processing

reversals is 821 and 927 for the mean and variance respectively. The null hypothesis that the sequences of mean and variance values do not incorporate underlying trends or variations should be accepted at the 0.05 significance level if the following is true:

$$[A_{56, 0.975} < A \leq A_{56, 0.025}] = [637 < A \leq 914]$$

The null hypothesis about the constancy of mean over time is accepted and the hypothesis about the constancy of variance over time is rejected at the 0.05

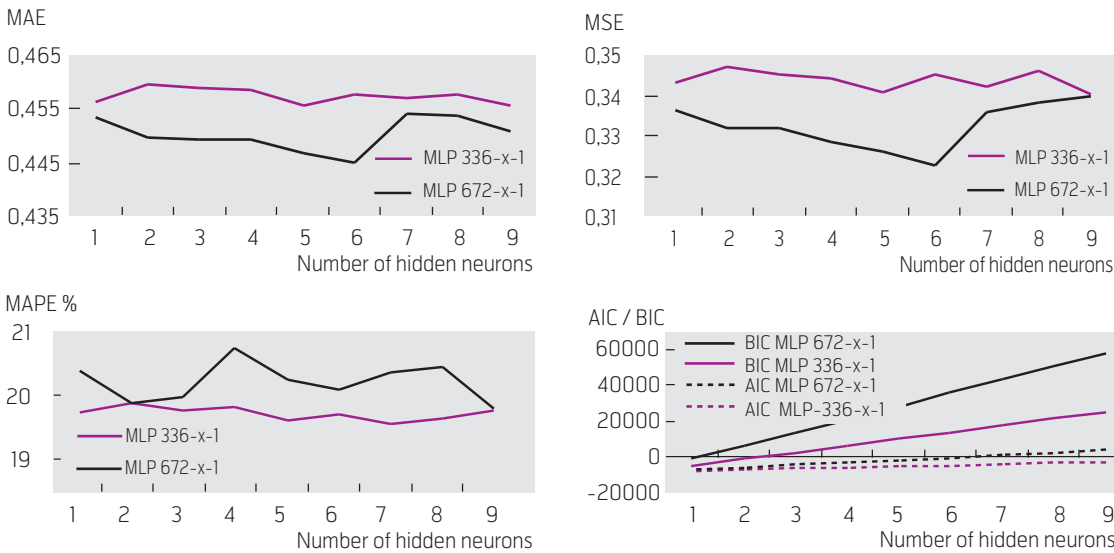


Figure 3.6 The results of in-sample evaluation

level. Therefore, further actions for stabilizing variance should be taken. It has been revealed during a test-and-trial procedure that a simple extraction of square root of the original time series works best in the case of application to the algorithm of neural networks.

• **Identification of seasonal components**

The Fourier analysis was applied to a time series after the subtraction of mean. The periodogram produced taking into account the coefficients of the Fourier decomposition is shown in Figure 3.3. At least three seasonal components are clearly identifiable. Their periods correspond to 12 hours, 24 hours and seven days. In contrast to a widely spread opinion, telecommunication traffic does not incorporate any monthly cycles.

The cyclic component (also called a trigonometric trend) can be expressed in mathematical terms taking into account the coefficients of the Fourier decomposition:

$$\begin{aligned}
 x_c(t) = & 1.6815 - 0.065 \cdot \cos\left(\frac{2\pi}{96}t\right) - \\
 & - 0.472 \cdot \sin\left(\frac{2\pi}{96}t\right) + 0.220 \cdot \cos\left(\frac{2\pi}{48}t\right) - \\
 & - 0.087 \cdot \sin\left(\frac{2\pi}{48}t\right) - 0.082 \cdot \cos\left(\frac{2\pi}{672}t\right) + \\
 & + 0.118 \cdot \sin\left(\frac{2\pi}{672}t\right)
 \end{aligned}$$

The next step is the subtraction of the trigonometric trend (see Figure 3.4). This type of deseasonalization is called analytical as the cyclic components are explicitly described by a formula. Another type of deseasonalization is called algorithmic and usually involves the use of moving average techniques.

It is worth noting that simple subtraction of the trigonometric trend does not obligatorily eliminate the presence of all the seasonal/cyclic components but at least eliminates the influence of the largest periodic components. In most cases it gives the opportunity to decrease the size of the input window, and consequently, to decrease the number of adjusted parameters of a neural network and training time.

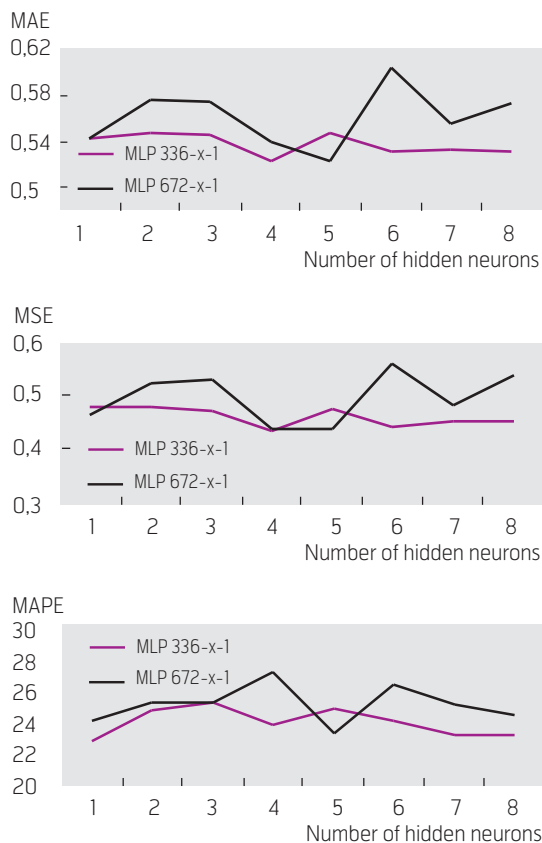


Figure 3.7 The results of out-of-sample evaluation

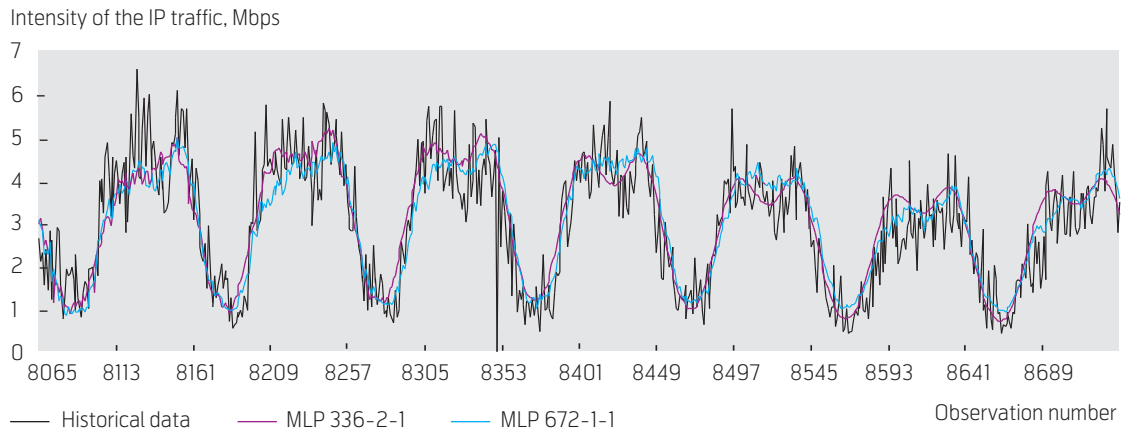


Figure 3.8 Forecasting the intensity of the IP traffic one week ahead

• Testing for normality

A non-parametric chi-square criterion was applied after the removal of outliers, stabilization of variance and subtraction of seasonal components. The same test was applied to the raw data without any pre-processing.

As we can see in Figure 3.5 the distribution of the raw data is far from normal. It is not only non-symmetric but also incorporated two pick values. On the other hand, the distribution of pre-processed data is very close to normal. Implementing the chi-square test we can accept the hypothesis about the normality of the distribution at the 0.01 significance level.

• Normalization/scaling

In our case the hyperbolic tangent was chosen for the activation of neurons. Therefore, all input/output data were scaled to the range of values between -1 and 1. The linear transformation was applied as the most appropriate in this case.

3.4 Evaluation of the Models and Final Forecasts

The results of in-sample verification of the trained neural networks are shown in Table 1. The in-sample verification is directly connected to the quality of a model and the preciseness of approximation. Standard measures of the goodness of fit, information criteria and adequacy of residuals were evaluated for raw data without any pre-processing, pre-processed data and post-processed data transformed back to the original form. Some of these parameters are also shown in Figure 3.6.

As we can see, the MAE and MSE are lower for the time series without pre-processing. On the other hand, the information criteria and the MAPE are lower for the pre-processed data. We can also conclude that pre-processing makes the residuals uncorrelated and therefore increases the performance and adequacy of forecasting models. However, in all the

cases the residuals are non-normal; that means that we cannot apply the Student's t-statistic for evaluating the confidence limits of the variable.

The out-of-sample stage refers directly to the production of *genuine* forecasts. The results of verification on the hold-out subset are shown in Figure 3.7. The potential outliers were excluded from the out-of-sample subset before the estimation of statistical parameters of both pre-processed and raw data sets.

Taking into account the results of in-sample and out-of-sample verification and following the principle of parsimony, one model for each case (with and without pre-processing) was selected for producing final forecasts. They are shown in Figure 3.8.

4 Main Findings

The algorithm of pre-processing of input data of neural networks is suggested for the application to telecommunications forecasting. It was applied to several real time series of different lengths representing the intensity of telephone traffic and the intensity of the total and international outgoing traffic of the IP network. For the sake of space saving, only one empirical example has been shown here. However, the main conclusions have been drawn taking into account the whole set of produced forecasts and the complete results of verification.

1 Despite the resistance of neural networks against noise, the prior identification of outliers is desirable. The standard measures of fit and the training error function are non-robust. The presence of only few anomalous outliers distorts and decreases the real values of these parameters significantly as neural networks (as well as any other forecasting methods) are not capable of modelling non-typical values. It also true for any irregular patterns. Spurious data points and patterns should be excluded or

replaced by more typical regular values in order to preserve the clearness of the experiment.

- 2 The size on the input window (ie. the number of input neurons) has to correspond to the largest identifiable period of a cyclic component estimated by the Fourier analysis. This requirement is explicitly connected to the informativity (entropy) of input patterns. On the other hand, further subtraction of a cyclic component decreases the number of relevant input neurons, and consequently, the number of adjusted parameters and training time.
- 3 A common assumption in many time series techniques is that the data are stationary. In actual practice reliable forecasts can be produced only if this condition is fulfilled. Non-parametric tests based on the ranks can be applied for testing the stationarity with much success. In the case of telecommunication traffic the main reason for non-stationarity is the instability of variance. This problem can be partially solved by applying the Box-Cox transformations or other similar methods. Non-stationarity in mean influenced by the presence of a monotonic trend is non-typical for telecommunication traffic and appears only with very long data sequences, the length of which exceeds two – three years. The shape of a monotonic trend can be easily identified by applying the regression techniques. Its further subtraction usually reduces a time series to stationarity.
- 4 One of the obvious advantages of neural networks is that they can work successfully with non-normally distributed data. However, we can affirm with certainty that normality is desirable. It ensures the unbiasedness of the standard estimates and facilitates the evaluation of confidence and/or prediction bands. It also increases the entropy of input patterns and makes the training process easier.
- 5 One of the most important aspects of reliable forecasting is the choice of the parameters of verification. Traditional statistical parameters (MAE, MSE, etc) often appear to be non-effective, especially in the case of such complicated non-linear statistical mechanisms as neural networks. Here we suggest selecting the final forecasting model taking into account two statistical options – the smallest value of the information criteria and the absence of autocorrelation of residuals tested by the Lagrange multiplier at the pre-defined level of significance. It ensures the choice of the most parsimonious and adequate forecasting model.

On the whole, the implementation of the algorithm of pre-processing before the training phase leads to a

decrease in the number of input neurons and the time of training, as well as to an increase in the reliability of forecasting models and out-of-sample forecasts.

References

- 1 Bendat, J S, Piersol, A G. *Random Data: Analysis and Measurement Procedures*. 3rd Ed. Wiley-Interscience, 2000.
- 2 Bishop, C. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.
- 3 Bloomfield, P. *Fourier analysis of Time Series: An Introduction*. 2nd Ed. John Wiley & Sons, Inc., 2000.
- 4 Chao, P Y, Hwang, Y D. An Improved Neural Network Model for the Prediction of Cutting Tool Life. *Journal of Intelligent Manufacturing*, 8, 107-115, 1997.
- 5 Cox, D R, Stuart, A. Some Quick Sign Tests for Trend in Location and Dispersion. *Biometrika*, 66, 80-95, 1955.
- 6 Draper, N R, Smith, H. *Applied Regression Analysis*. 3rd edition. Wiley, 1998.
- 7 Foster, F G, Stuart, A. Distribution-Free Tests in Time-Series Based on the Breaking of Records. *Journal of the Royal Statistical Society, Series B (Methodological)*, 16 (1), 1-22, 1954.
- 8 Franses, H P, van Dijk, D. *Nonlinear Time Series Models in Empirical Finance*. Cambridge University Press, 2000.
- 9 Guh, R-S. Robustness of the Neural Network Based Control Chart Pattern Recognition System to Non-Normality. *International Journal of Quality & Reliability Management*, 19 (1), 97-112, 2002.
- 10 Haykin, S. *Neural Networks: A Comprehensive Foundation*. 2nd Ed. Prentice Hall, 1998.
- 11 Himmelblau, D M. *Process Analysis by Statistical Methods*. New York, Wiley, 1970.
- 12 LeCun, Y, Bottou, L, Orr, G B, Müller, K-R. Efficient BackProp. In: Orr, G B, Müller, K-R (Eds.). *Neural Networks: Tricks of the Trade*, 9-50. Berlin, Springer, 1998.
- 13 Leland, W E, Taqqu, M S, Willinger, W, Wilson, D V. On the Self-Similar Nature of Ethernet Traf-

- fic (Extended Version). *IEEE/ACM Transactions on Networking*, 2 (1), 1-15, 1994.
- 14 Ljung, G M, Box, G E. On a Measure of Lack of Fit in Time Series Models. *Biometrika*, 65, 297-303, 1978.
 - 15 Mallows, C L. Some Theory of Nonlinear Smoothers. *The Annals of Statistic*, 8 (4), 695-715, 1980.
 - 16 McNelis, P D. *Neural Networks in Finance: Gaining Predictive Edge in the Market*. Elsevier Academic Press, 2005.
 - 17 Medeiros, M C, Teräsvirta, T, Rech, G. Building Neural Network Models for Time Series: A Statistical Approach. *Journal of Forecasting*, 25, 49-75, 2006.
 - 18 Nelson, M, Hill, T, Remus, W, O'Connor, M. Time Series Forecasting Using Neural Networks: Should the data be Deseasonalized First? *Journal of Forecasting*, 18, 359-367, 1999.
 - 19 Otnes, R K, Enochson, L. *Applied Time Series Analysis: Basic Techniques*. John Wiley & Sons Australia, Lim., 1978.
 - 20 Palit, A K, Popovic, D. *Computational Intelligence in Time Series Forecasting: Theory and Engineering Applications*. London, Springer-Verlag, Lim., 2005.
 - 21 Priestley, M B. *Spectral Analysis and Time Series*. London, Academic Press, 1981.
 - 22 Sheluhin, O I, Smolskiy, S M, Osin, A V. *Self-Similar Processes in Telecommunications*. John Willey & Sons, 2007.
 - 23 Thomaidis, N S, Dounias, G. *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
 - 24 Weigend, A S, Rumelhart, D E, Huberman, B A. Generalization by Weight-Elimination with Application to Forecasting. *Proceedings of the 1990 Conference on Advances in Neural Information Processing Systems*, 3, 875-882, 1990.
 - 25 Zhang, G P, Qi, M. Neural Network Forecasting for Seasonal and Trend Time Series. *European Journal of Operational Research*, 160, 501-514, 2005).

Irina Klevecka received the BSc degree in Electrical Engineering (2000) and the MSc degree in Telecommunications (2004) from Riga Technical University. She has recently finished her post-graduate studies at the Institute of Telecommunications of Riga Technical University and is due to defend her PhD thesis in the beginning of 2009. Her main research interests include technology forecasting and the development of new electronic communications services.

klevecka@inbox.lv

Janis Lelis holds the PhD degree from the Banch-Bruevich Saint-Petersburg State University of Telecommunications (1982) and the DrScIng degree from Riga Technical University (1992). He has been Executive Director of the Latvian Telecommunications Association since 1999. He has also been Associated Professor of the Faculty of Electronics and Telecommunications of Riga Technical University for more than twenty years. His main research interests are telecommunications regulation, the development of telecommunications networks and fibre optics.

jlelis@latnet.lv