Time Series Prediction Using Artificial Neural Networks: Influence of The Input Vector Size

DAVID SAMEK¹, PAVEL VARACHA²
Faculty of Technology¹, Faculty of Applied Informatics²
Tomas Bata University in Zlin
nam. T. G. Masaryka 5555, 760 01 Zlin
CZECH REPUBLIC
samek@ft.utb.cz¹, varacha@fai.utb.cz² http://www.utb.cz

Abstract: - Artificial neural networks provide powerful tools for linear and nonlinear system modeling and prediction. They are commonly used in various fields, such as economics, medicine, industry, aerospace, chemistry etc. Artificial neural networks are capable comprehend single input – single output as well as multiple input – multiple output functions. This paper is focused on prediction of non-artificial time series that are typically considered as single input – single output systems from the point of view of the predictor. The paper presents study of the influence of the input vector length to prediction quality. All simulations were done in MATLAB.

Key-Words: - artificial neural network, time series, prediction, benchmark,Santa Fe competition, Matlab

1 Introduction
One of the most exciting challenges in human researching is future forecasting. The need of the knowledge of the future does not come only from the natural human curiousness, but also from the necessity to improve the current technologies and methods.

The term prediction, which often substitutes the term forecasting, is very wide. It compromises methodologies for weather forecast [1], [2], [3], [4] financial data prediction [5], [6], [7], predicting the biological characteristics [8], technological parameters [9], horse racing results [10], energy grid behavior [11], etc. However, it can be said that the prediction is always based on the model of the process to be predicted.

Generally, the two main approaches to modeling are possible – white box modeling and black box modeling. The first method uses a priori knowledge about the predicted system. Typical white box model is mathematical model based on physical and chemical laws. The black box modeling is based on the identification data that allows update initial predictor to obtain proper results. Typical black box models are artificial neural networks.

There have been published many applications of artificial neural networks in prediction [12], [13], [14]. Nevertheless, the choice of proper artificial network and its settings is usually not trivial. There are many parameters that influence the quality of prediction. One of them is the number of input neurons (in the so called zero layer), in other words the length of the input vector of the predictor. Generally, the size of the input vector relates to frequency of the signal. Some authors mention phrase memory which refers to the correlation of a prediction back to n previous intervals of time [15], [16]. Though, an exact methodology that would provide strict rule for designing zero layer of specific artificial neural network type does not exist.

The motivation of this contribution comes from our previous research and the paper studies the influence of the input vector size for six selected artificial neural network structures.

This paper is organized as follows. Sect. 2 provides some background on time series to be predicted. Section 3 describes the design and implementations of the tested artificial neural network structures. Sect. 4 presents simulations and results of the testing. Section 5 consists of some concluding remarks.

2 Testing time series
The data set A from the Santa Fe Competition [17] was selected as the signal to be predicted (see Fig. 1). This time series was generated by NH₃ laser and it is good example of realistic data. It was selected because due its competition usage it became a common way of predictor testing and benchmarking.

Recent Researches in Circuits and Systems


441
The measurements were made on an 81.5-micron 14NH3 cw (FIR) laser, pumped optically by the P(13) line of an N2O laser via the vibrational aQ(8,7) NH3 transition. The intensity data was recorded by a LeCroy oscilloscope. No further processing happened. The experimental signal to noise ratio was about 300 which means slightly under the half bit uncertainty of the analog to digital conversion [18].

3 Tested artificial neural networks

There are inexhaustible variety of types and structures of artificial neural networks, but not all of them are capable to model and predict time series. However, there are some popular artificial neural networks that are mostly used. One of the most popular is multilayered feed-forward neural network (MFFNN). These networks are very versatile and have been already applied in many applications in the prediction task. Even the special case of MFFNN - simple adaptive linear network (ADALINE) - can model and predict various systems. When the temporally/sequentially extended dependencies over unspecified (and potentially infinite) intervals should be modeled [19], the recurrent neural networks are often applied. It is worth of noticing the radial basis function neural networks (RBFNNs). RBF networks are popular due to its very fast training [20]. Obviously, the list of artificial neural networks would be long, but it is necessary to mention also functional networks [21], Kohonen networks [22], [23] and probabilistic fuzzy neural network [24].

In this paper, there were chosen following structures of artificial neural networks to be tested: two variants of multilayered feed-forward neural network, because of its wide usage, adaptive neural network due to its simplicity, Elman neural network as the representative of the recurrent neural networks, two structures radial basis function neural network, because it provides simple training with good prediction performance and adaptive neural network due to its simplicity.

3.1 Multilayered feed-forward neural networks

Multilayered feed-forward neural networks have neurons structured in layers and the information flows only in one direction (from input to output). Typically, all neurons in specific layer have same transfer function, while variety of transfer functions is used. It has been proved [25], [26] that two-layered MFFNN can approximate any function with certain accuracy while non-polynomial transfer function in the hidden layer is used. In this paper two variants of MFFNN were tested.

The first tested structure had five neurons with hyperbolic tangent transfer function in the hidden layer and one neuron with linear transfer function in the output layer. This network will be denoted as $\text{mffntp}$ in the following text. The second structure had same hidden layer, but one neuron in the output layer utilized hyperbolic tangent as a transfer function. This network will be denoted as $\text{mffntt}$.

Both networks were trained using Levenberg-Marquart algorithm built in Matlab Neural Network Toolbox.

3.2 Adaptive linear neural network

Adaptive linear neural networks can be regarded as a special (simple) case of multilayered feed-forward neural networks. They have typically only one layer with linear transfer function. Despite its simplicity they have many applications [27], [28], [29]. Very often (in case of need of one output value) adaptive linear networks have only one neuron. This methodology is accomplished in the paper and such structure will be called adaline hereinafter.

For the creation and training of adaptive linear neural networks were used newlin and adapt functions from Matlab Neural Network Toolbox. Nevertheless, the standard approach to network training as it was applied for other types of ANN was not effective. The adaptive linear neural networks had to be adapted recursively using sliding adaptation method.
3.3 Recurrent neural networks
Elman neural networks were selected as a representative of large group of recurrent neural networks. Typical Elman network has one hidden layer with delayed feedback. In this article the hidden layer contained ten neurons with hyperbolic tangent transfer function and the output layer of the Elman neural network used linear transfer function (this structure is below denoted as enn). The backpropagation algorithm was used for the enn training.

3.4 Radial basis function neural networks
Typical RBFNN contains two layers, while the hidden layer utilizes radial basis transfer function and output layer employs linear transfer function.

Radial basis function neural networks are popular for their fast training. Unfortunately, this advantage is balanced with higher memory requirements because the network has as many RBF neurons as many training vectors are used (Matlab Neural Network Toolbox function newrbe). This network is below denoted as rbf.

However, in the Matlab Neural Network Toolbox is available the second approach to RBFNNs – the function newrb. This methodology is much more computational demanding but it is the only solution how to model with large amount of training data. Such structure will be in the following text indicated as rbfu. Due to long training times the maximum number of RBF neurons was set to 500.

4 Simulations and results
The tested number of inputs from which the prediction is to be performed is as follows: 2, 3, 4, 5, 6, 7, 8, 9, 10, 50, 100. The original signal was processed into training matrices according to number of values in the input vector. For each combination of the tested artificial network type and the input vector size were created 100 simulations and average values were used for the consequent evaluation and comparison.

For better comparison two prediction quality criterions were defined. The first criterion ABS describes total sum of absolute values of prediction errors relative to number predictions whilst the second criterion function SQR characterizes total sum of squares of prediction errors relative to number predictions.

\[
\text{ABS} = \frac{\sum_{i=1}^{N} |t(i) - p(i)|}{N} \tag{1}
\]

\[
\text{SQR} = \frac{\sum_{i=1}^{N} (t(i) - p(i))^2}{N} \tag{2}
\]

Where \(N\) is number of predictions (length of predicted signal), \(t\) stand for target (original) signal, \(p\) denotes predicted signal and \(i\) is number of the prediction.

![Fig. 2 – Prediction using mffnntp](image)

![Fig. 3 – Prediction using mffnnmtt](image)

The ABS criterion gives same importance to all errors. On the other hand SQR emphasizes higher errors and lower prediction errors are suppressed.

As can be seen from figures 2-7 the best results provided rbf. This is caused by the fact that rbf networks have so many neurons. In other words
each neuron in the hidden layer works as an exclusive predictor for one input vector.

It is worth of noticing that \( \text{rbfu} \) performed reasonable good results with markedly less neurons (maximally 500).

On the contrary, the worst results were obtained using \( \text{enn} \). Elman neural network have apparently problems to model this kind of signal. The second worst was \( \text{adaline} \), if predictions for 50 and 100 input values were omitted. The criterions \( \text{ABS} \) and \( \text{SQR} \) for two largest input vectors were too high to display them in the figure 4.

Multilayered feed-forward neural networks lie in the middle of results chart. Their prediction accuracy verges to \( \text{rbfu} \) as far as \( \text{ABS} \) criterion is concerned. However, as can be seen from \( \text{SQR} \) course in figures 2, 3 and 7, the \( \text{mffntn} \) and \( \text{mffntn} \) produce significantly more high prediction errors.

The influence of the size should be assessed separately for \( \text{rbf} \) and the rest of tested structures. The radial basis function neural network \( \text{rbf} \) due its exact design converges to zero prediction error with increasing length of input vector. The prediction error for the rest of tested artificial neural networks generally slightly decreases in the beginning but after certain interval rises approximately around 10 values in the input vector. It can be concluded that the optimal length of input vector lies between 3 and 10 with the exception of \( \text{rbf} \).

\section{5 Conclusions}

The paper showed the case study of non-artificial time series prediction using various artificial neural network structures. In the limited space was not possible to present all results of the benchmarking and testing. Nevertheless, the presented simulations showed dependencies of prediction accuracy on the number of values in input vector.

It can be concluded that selection of optimal input vector size is not trivial and it depends on the selected artificial neural network type and the
predicted system / signal. As was presented hereinbefore, it was typical for most of the tested networks that the prediction error decreases in the beginning, and then, after reaching minimum, the prediction error rises up. Therefore, the optimal number of inputs usually lies between 3 and 10. Exception to this behavior is the rbf network that due to its “exact training” has descending prediction error trend.

As future research, results presented in this paper will be compared with neural networks optimized by Self-Organizing Migration Algorithm (SOMA). This evolutionary algorithm seems promising as it has many successful implementations on multiobjective optimization problems, e.g. [30 - 33].

Acknowledgment

This paper is supported by the European Regional Development Fund under the project CEBIA-Tech No. CZ.1.05/2.1.00/03.0089.

References:


