LABORATORY WORK №1

LINEAR SINGLE LAYER PERCEPTRON FOR TIME SERIES FORECASTING

Objective: To study the training and functioning of a linear SLP for time series prediction.

Theoretical materials

The ability of neural networks (after training) to generalize and prolong the learning outcomes permits to building various predictive systems on their basis.

Let the time series $x(t)$ be given in the interval $t=\overline{1,m}$. Then the task of forecasting is to find the continuation of the time series in an unknown interval, that is, it is necessary to determine $x(m+1)$, $x(m+2)$ etc. (Fig. 2.2).

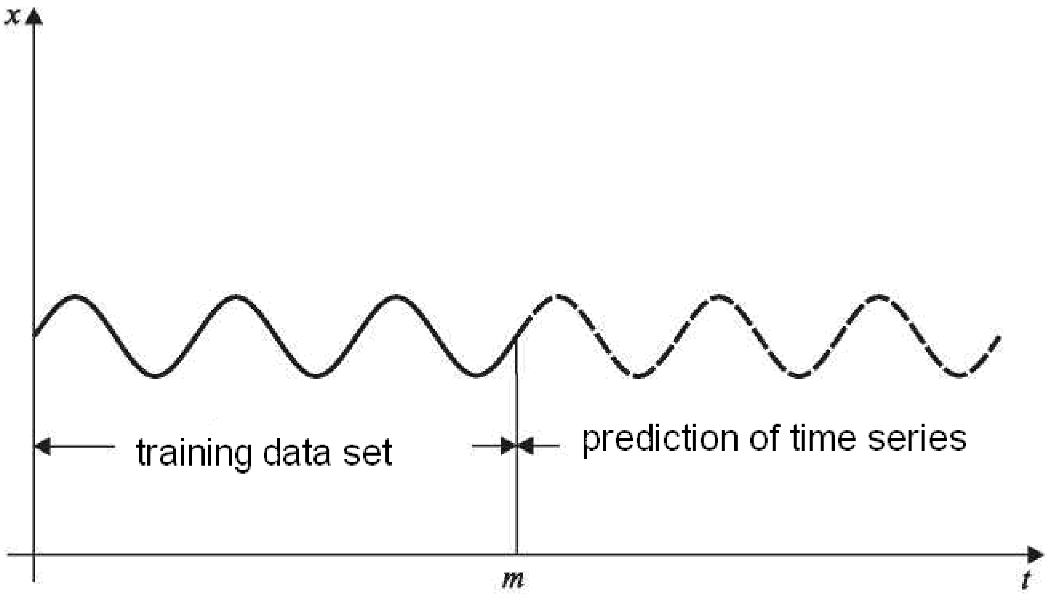

Fig. 2.2. Time series predicting

The "sliding window" approach is used for forecasting. It is characterized by a window length p equal to the number of elements in a time series simultaneously fed to the neural network. This determines the structure of the neural network, which consists of p input and one output neurons. The set of known values of the time series forms a training data set. The dimension of training data (number of training samples) is defined by the following way:

$$L = m - p$$

This model corresponds to the linear autoregression model and is described by the expression

$$\overline{x(t)} = \sum_{k=1}^{p} \omega_k x(t-p+k-1) - T,$$

where $\omega_k, k=\overline{1,p}$ – weights; T- threshold; $\overline{x(t)}$ – assessment of the value of time series $x(t)$ at the time t.

The prediction error is defined as

$$e(t) = \overline{x(t)} - x(t).$$

The training patterns of a neural network can be represented as a matrix, the rows of which characterize the vectors supplied to the input of the network:

$$X = \begin{bmatrix} x(1) & x(2) & \dots & x(p) \\ x(2) & x(3) & \dots & x(p+1) \\ \dots & \dots & \dots & \dots \\ x(m-p) & x(m-p+1) & \dots & x(m-1) \end{bmatrix}.$$

This is equivalent to moving the window along a time series with step equaled one.

Thus, a set of known elements of the time series is used to train the neural network for forecasting. The delta rule is used for learning:

$$\omega_1(t+1) = \omega_1(t) - \alpha(y \quad - e \quad)x_1 \ ,$$

$$\omega_2(t+1) = \omega_2(t) - \alpha(y \quad - e \quad)x_2 \ ,$$

$$\omega_p(t+1) = \omega_p(t) - \alpha(y \quad - e \quad)x_p \ ,$$

$$T \ (t+1) = T \ (t) + \alpha(y \quad - e \quad),$$

After training, the network should predict the time series for a next period of time.

Training Algorithm:

1. Choose the learning rate $\alpha(0 < \alpha < 1)$ and desired total mean square error E_m, which we want to reach during training of neural network.

2. Start with random weights for the connections.

3. Select an input vector X from the set of training samples.

4. Modify all connections in accordance with delta rule (1.24) and (1.25).

5. Go back to 3.

5. Algorithm is continued until weights will cease to change or total mean square error will be less than desired error $E_s \leq E_m$, or the total mean square error E_s does not decrease or decrease slightly.

Total mean square error (Całkowity błąd kwadratowy):

$$E_s = \sum_{k=1}^{L} E(k) = \frac{1}{2} \sum_{k=1}^{L} \sum_{j=1}^{m} (y_j{}^k - e_j^k)^2,$$

Where L is number of training samples, y_j^k and e_j^k – real and desired output of j-th unit for k-th sample.

Remark:

For better neural network training, the input data can be transformed into a smaller range of values.

Let the input be in the range $[x_{\min}, x_{\max}]$. It is necessary to map them into a range $[a,b]$, i.e.

$$x \in [x_{\min}, x_{\max}] \to [a,b].$$

In this case, each component of the input sample is transformed as follows:

$$\overline{x_i^k} = \frac{(x_i^k - x_{\min})(b-a)}{(x_{\max} - x_{\min})} + a.$$

For instance if $[x_{\min}, x_{\max}] \to [0,1]$ then

$$\overline{x_i^k} = \frac{x_i^k - x_{\min}}{x_{\max} - x_{\min}}.$$

TASK:

DEVELOP SLP USING ANY PROGRAMMING LANGUAGE FOR PREDICTION OF THE FOLLOWING FUNCTION

$$y = a\sin(bx) + d \quad .$$

Training should be performed using 30, 50 and 100 points, respectively. Testing should be performed using 20 points. X changes in increments of 0.1. The learning rate is chosen by the student independently. It is necessary also to use adaptive learning rate and to compare the results.

Report content:

1. Learning outcomes: graph of time series, a table with the following columns: reference value, real value, deviation; the graph of the error change depending on the epoch.
2. Prediction results: table with the following columns: reference value, real value, deviation.
3. Comparison of results

Variants

№ of variant	a	B	d	Number of input units
1	9	1	0.2	5
2	2	6	0.3	3
3	4	5	0.5	5
4	1	10	0.6	7

5	2	9	0.1	5
6	3	8	0.2	5
7	4	7	0.8	7
8	5	6	0.3	7
9	6	5	0.4	9
10	1	3	0.1	5
11	2	12	0	7
12	3	14	0	5
13	4	2	0.5	3
14	7	3	0.6	4
15	6	4	0.7	6
16	5	5	0.8	7
17	2	6	0.9	5