

# Moldova Republic's Gross Domestic Product Prevision Using Artificial Neural Network Techniques

Ilie Constantin

Ilie Margareta

*„Ovidius” University of Constanta,  
ilie.ctin@yahoo.com*

## Abstract

Considering the present economical status, it is necessary to efficiently manage the Gross Domestic Product (GDP) especially for developing countries. Thus, it will be advantageous to have some knowledge about future trends or specific values of the GDP in accordingly to the dynamics of certain economics indices such as Investment in fixed capital, Imports from other countries etc. The present paper contains the basic research to determine the possibility of using artificial intelligence for the prevision of future GDP. In order to determine the GDP of Moldova Republic the following data were considered: Imports from other countries, Investment in fixed capital, Retail trade and Industrial production. A feed forward artificial neural network (ANN), with 10 hidden neurons, was trained and tested. After 590 iterations a maximum training absolute error of 0.008983 was obtained. Also the absolute validation error was 0.012664 and the network error was 0.000248. The final testing errors belongs to the [0.00; 2.47] interval of absolute values and to the [0.00; 0.23] interval of relative values. The results offer the bases for the future researches.

**Key words:** Gross Domestic Product, artificial intelligence, prevision, Imports, Retail trade;

**JEL Classification:** C82

## 1. Introduction

The Artificial Neural Networks are used in several domains such as the card fraud detection, the prevention of stock exchange, the effects of decision making etc. Their capabilities emerge from their possibilities to simulate nonlinear systems and processes and give tendencies and previsions in fields (like economics) where, currently, the variables are more nonlinear and unpredictable.

Predicting the GDP can be very useful for decision making in order to initiate and develop decision in order to influence the regulation of GDP.

The survey of input and output data correspond to the monthly 1995-2009 timetable. For the processing and construction of the feedforward ANN the Alyuda NeuroIntelligence was used [1].

## 2. Experimental research

The research was developed regarding to the phases of artificial neural network use:

- Data analysis;
- Preprocessing data;
- Neural Network design;
- Neural Network training;
- Training testing - Final query.

The testing and final query will be presented in the Results and Discussions section.

### *Data Analysis*

The data analysis is very important because it prepares the grounds for the entire following procedures and any error in the data values will be perpetuate in all the next phases and will affect the results (“garbage

in, garbage out”). The values analyzed are presented in table 1.

From the basic data set, that contains 7 numeric columns, 3 of those columns were disabled: Agricultural production, Cargo, Consumer price indices, as the result of the influence that those data had over the output. The remaining 4 numeric columns: Industrial production, Investment in fixed capital, Imports from other countries and Retail trade (In% to previous year) were accepted as input data.

The entire data set consisted from 28 series from which 3 were used for the testing process. The remaining 25 data series were organized as follows: 22 records to Training set (88%), 2 records to Validation set (8%) and 1 record to Test set (4%).

Table 1: Data values analyzed [4]. (Example)

Producției industriale	Investiții în capital fix	Importurile din alte țări	Comerț cu amănuntul	PIB-ul
96.00	84.00	268.00	112.0	98.60
85.00	110.00	176.00	88.00	93.50
88.00	78.00	114.00	73.00	96.60
116.0	107.00	274.00	118.0	106.6

### Preprocessing data

In order to get an efficient and easier training, in several occasions, methods or techniques of data values preprocessing are used. This is applicable in our case and the method was: Scaling Numeric Columns. The result is in table 2.

Numeric columns are automatically scaled during data preprocessing. By default numeric values are scaled using the following formula [1]:

$$SF = \frac{SR_{max} - SR_{min}}{X_{max} - X_{min}} \quad (1)$$

$$X_p = SR_{min} + (X - X_{min}) \cdot SF \quad (2)$$

where:

$X$  - actual value of a numeric column

$X_{min}$  - minimum actual value of the column

$X_{max}$  - maximum actual value of the column

$SR_{min}$  - lower scaling range limit

$SR_{max}$  - upper scaling range limit

$SF$  - scaling factor

$X_p$  - preprocessed value

Table 2: Data values preprocessed.  
(Example)

Producției industriale	Investiții în capital fix	Importurile din alte țări	Comerț cu amănuntul	PIB-ul
-0.29	-0.73	0.92	0.27	98.60
-1.00	0.42	-0.22	-0.50	93.50
-0.80	-1.00	-1.00	-1.00	96.60
1.00	0.28	1.00	0.47	106.60
-0.29	-0.73	0.92	0.27	98.60

Preprocessing report:

Input columns scaling range: [-1..1];

Output column(s) scaling range: [11.84..17.96].

### Neural network design

Considering the values, the numbers and the type that defines the data, the feedforward artificial neural network was chosen to be used. The number of hidden neurons and layers was obtained through the search of several architectures and due to the past experience. So the final architecture used is presented in figure 1.

The architecture search report was:

- Architecture was selected manually  
The architecture selected for training is [4-7-1]- had the best fitness;
- Fitness criteria: Inverse Training error;
- Hidden layers activation function: Hyperbolic tangent;

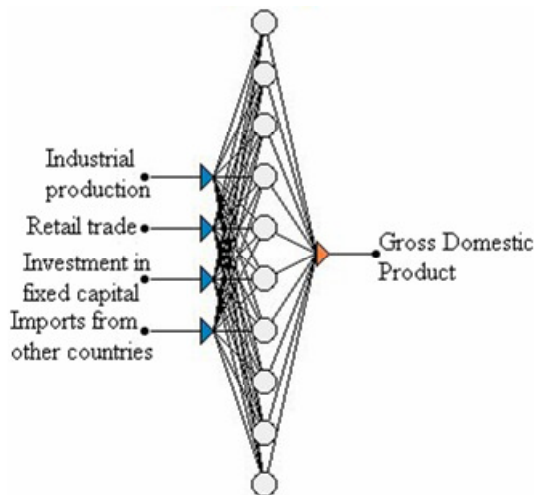


Figure 1: ANN architecture [4-10-1].

- Output data: Gross Domestic Product (GDP) [ln% to previous year];
- Output error: Sum-of-squares;
- Error function: Sum-of-squares;
- Activation function for the output layer: Linear.

The usual training algorithm for the feedforward neural network is back propagation algorithm, in this case the Levenberg-Marquardt algorithm.

Levenberg-Marquardt is an advanced non-linear optimization algorithm. It is the fastest algorithm available for multi-layer perceptrons. However, it has the following restrictions:

- It can only be used on networks with a single output unit.
- It can only be used with small networks (a few hundred weights) because its memory requirements are proportional to the square of the number of weights in the network.
- It is only defined for the sum squared error function and therefore it is only appropriate for regression problems.

The results of the training session can be visualized in figure 2 as the network error.

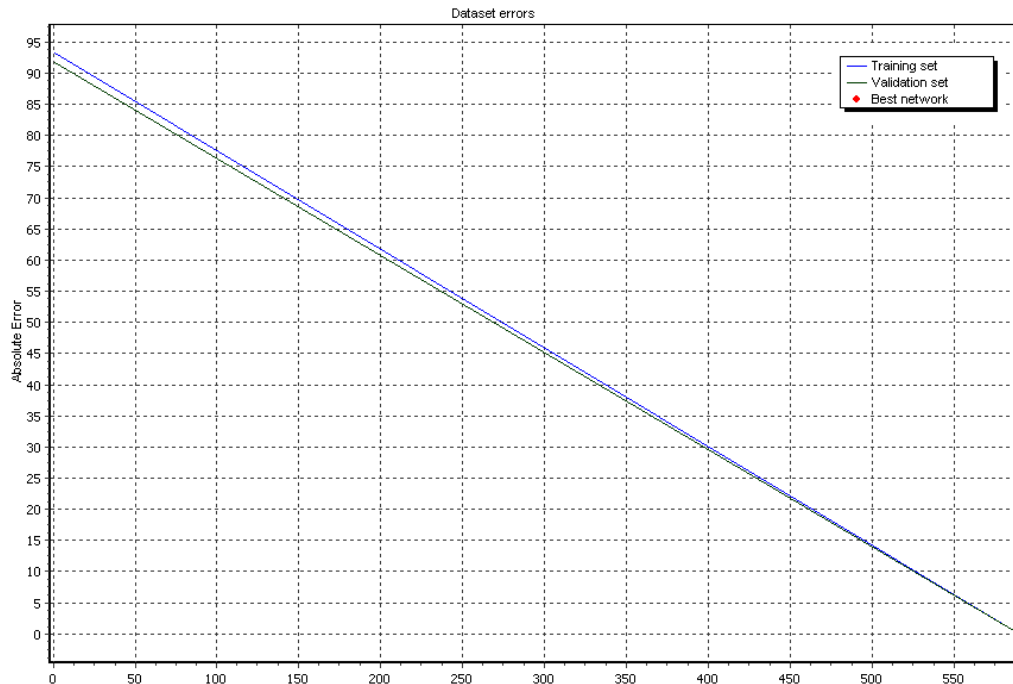


Figure 2. Network errors. Error improvement vs. iteration number.

The training conditions were:

- Quick propagation coefficient: 0.5;
- Learning rate: 0.1;
- Number of iterations: 590;
- Stop training conditions: Desired error achieved value: max AE=0.0001, track on training set.

Where: *Quick propagation coefficient* - Quick propagation coefficient is additional training parameter for Quick propagation algorithm. This parameter is used to control magnitude of weights increase in some cases.

*Learning rate* - A control parameter used by several learning algorithms, which affects the changing of weights. The bigger learning rates cause bigger weight changes during each iteration.

*AE - Absolute error*. An error value that indicates the "quality" of a neural network training. Calculated by subtracting the current output values with the target output values of the neural network. The smaller the network error is, the better the network had been trained.

*Iterations* - A single complete presentation of the training set to the neural network during the training process. After 590 iterations a maximum training absolute error of 0.008983 was obtained. Also the absolute validation error was 0.012664 and the network error was 0.000248.

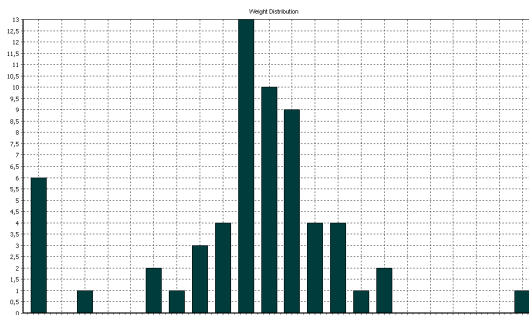


Figure 3. Network' weights after training session.

In figure 3 it is shown the weight distribution for the neural network used, after the training phase.

### 3. Results and Discussions

In order to see that the training was successful and the ANN was ready to prevision the sales values from the input data: Industrial production, Investment in fixed capital, Imports from other countries and Retail trade (In% to previous year), it was necessary to take a final testing. This the final query that uses data unrevealed to the ANN until then. If the ANN offers results values very closed to the real data, then the training was successful and it can be use in the future prevision. In order to validate the research results an interval of difference between the real and simulated data was imposed: [-3; 3] absolute value and [-5; 5] % relative values. The comparison between the real data and data simulated by the ANN is presented in table 3.

Table 3: Gross Domestic Product (GDP)  
[In% to previous year].

Real data	Simulated data	Absolute value difference	Relative value difference [%]
102.10	102.10	0.00	0.00
107.20	104.73	2.47	2.30
93.50	93.42	0.08	0.09
	MIN	0.00	0.00
	MAX	2.47	2.30

As can be seen in table 3, the results of the testing not only that comply with the imposed condition but the error are smaller than those expected.

#### 4. Conclusions

Having in mind the results presented above we can say that this first phase of the research was a success.

The next phases imply adding new input data that influences the Gross Domestic Product and also the effects that those data have over the major economic dynamics. Also, future research must be modified for the use by the small and medium business, in order to modify the acquisition, stock and flow management for the products and/or raw materials.

#### 5. References

- [1] <http://www.alyuda.com/neural-networks-software.htm>.
- [2] C. Ilie, C. Neagu, Bucuresti, Doctoral thesis, (2010).
- [3] L. Barbes, C. NEAGU, L. V. Melnic, C. Ilie, M. VELICU, *The use of Artificial Neural Network (ANN) for Prevision of Some Airborne Pollutants Concentration in Urban Areas*, Revista de Chimie, Bucuresti, 2009.
- [4] <http://www.cisststat.com/>
- [5] C. Neagu. C. Ioniță, “*Rețele Neuronale. Teorie și aplicații în modelarea proceselor și sistemelor de producție*”, București, 2004.