COMBINING GENETIC ALGORITHMS AND ARTIFICIAL NEURAL NETWORKS IN FORECASTING CONSUMER PRICE INDEX (CPI)

Dezdemona Gjylapi[1], Vladimir Kasemi[2]

[1]Department of Informatics Engineering,University "Pavaresia" Boulevard "Ismail Qemali", Vlore, Albania, dezdemonal@gmail.com

[2]Department of Naval and Mechanical Engineering, University "Ismail Qemali", Square "Pavaresia" Skele, Vlore, Albania, vladimirkasemi@yahoo.com

Abstract

Genetic algorithms (GAs) and artificial neural networks (ANNs) belong to the class of evolutionary computing algorithms.
ANN is a cognitive model approach widely used for machine learning. ANN is the emulation of the biological neural network, which is composed of many interconnected neurons. Due to their nonlinear model, ANNs can be used in many business applications. ANNs find wide usage in time series forecasting and one of their most promising ways is the combination with other intelligent techniques, such as GAs.
A GA is the idea of applying the principles of natural biological evolution in artificial systems. GAs belong to the class of optimization procedures and mainly applied in spaces that are too large and complex to be searched in a finite way. GAs have been used for ANN in two main ways: to optimize the network architecture and to train the weights of a fixed architecture.
The aim of this paper is the combination of ANNs with GAs to forecast the consumer price index (CPI), as a very important indicator to measure the inflation rate in the economy of a country. This forecast provides the monetary and fiscal policy makers with the advantages of preparing timetables aimed at macroeconomic stability and sustainable economic growth.
In this paper, has been used a multi-layer ANN model, which uses GA to learn, i.e. to train its weights in order to increase the efficiency of ANN. During the training phase, certain amount of previous values of CPI time series, taken from the Bank of Albania out the period January 2009 - December 2012, is given as input to the ANN and the network is trained to predict the next values of CPI time series. It has been found that ANNs combined with GAs are ideal for CPI forecasting.

Keywords**:** *neural network, genetic algorithm, forecasting, CPI*

1 - Introduction

Consumer price index (CPI) is a very important indicator to measure inflation in the economy of a country. This forecast provides the monetary and fiscal policy makers with the advantages of preparing timetables aimed at macroeconomic stability and sustainable economic growth.
The accurate forecast of Inflation has an impact in the forecast and calculation of the expected inflation rate which is closely related to future fluctuations in interest rates, the demand for money, the amount of savings and investments in a country, as well as many other significant macroeconomic indicators like GDP, unemployment rate, economic growth, etc. In this paper we use the CPI time series.
Time series can be financial data series, observed physical data series and mathematical data series. Time series forecasting takes a series of existing data to predict future values. The aim is to observe or design existing series to enable the accurate forecast of unknown future data values. (Foster, 1992)
The sample data used in this paper are taken from INSTAT (Institute of Statistics).
Time series forecasting has several important applications. One application is preventing undesirable events by forecasting the event, identifying the circumstances preceding the event, and taking corrective action so the event can be avoided. Another application is forecasting undesirable, yet unavoidable, events to preemptively lessen their impact. Finally, many people, primarily in the financial markets, would like to profit from time series forecasting. Nevertheless many products are available for financial forecasting. (Plummer, 2000)
One of them is the Consumer Price Index (CPI).

1.1 - Difficulties in forecasting time series

There are some difficulties that may arise in time series forecasting. Depending on the type of data series, a particular difficulty may or may not exist. (Bunn, 2000)
The first difficulty is the limited amount of data. For the series of observed data, the limited amount of data can be the main difficulty.
The second difficulty is the noise. There are two types of noisy data: 1) erroneous data and 2) components that obscure the basic form of the data series. (Foster, 1992)
The third difficulty is *non-stationarity*; data that does not have the same statistical quality (e.g, mean and variance) at any point in time.
The fourth difficulty is the choice of the forecasting technique. From the statistics in artificial intelligence, there are countless possibilites of choice for a technique. One of the simplest techniques is to look at a series of data related to past events and use matches to make a prediction. One of the most complex techniques is the training of a model on series and then using the model to make a prediction. Neural networks are examples of such technique (Sansom & Saha, 1999) .

2 - Basic concepts of an artificial neural network

Artificial Neural Network (ANN) is a cognitive-modeling methodology, widely used for machine learning. ANN is an emulation of a biological neural network, which is composed of many interconnected neurons. However, it uses only a very limited set of concepts of its biological counterpart.

An ANN may have one or more layers of neurons. They may be fully or partially connected. Each link between two nodes has a weight, which summarizes the "knowledge" of the system. The processing of the existing cases with inputs and expected results, will adjust these weights based on the difference between the actual and the expected results. Because of the nonlinear ANN model, they can be used in many business applications.

The whole process of ANN learning has three consecutive steps:
1. Calculation of the interim results
2. Comparison of the results with the desired objectives
3. Adjustment of the weights and then start over the process

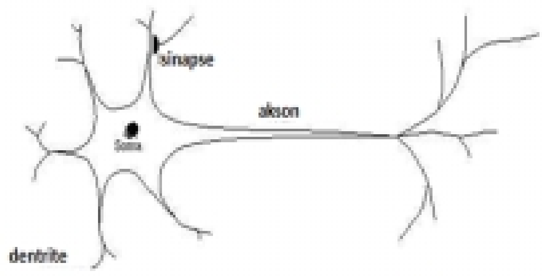

Figure 1. Biological neuron

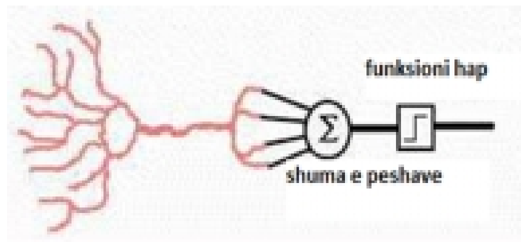

Figure 2. Artificial neuron (perceptron)

An ANN consists of a number of simple and interconnected processors, also called neurons, analogous to biological neurons in the brain. Each neuron receives a number of input signals through its connections, however, it gives no more than one output signal. The output signal is transmitted through the neuron's exit line. (Negnevitsky, 2005)
The mathematical model of a biological neuron is called *Perceptron*. While dendrites of the biological neurons receive electrical signals from the axons of other neurons, to the perceptron these electrical signals are presented as numerical values. At synapses between dendrites and axons, electrical signals are modeled in different amounts. This is also modeled on the perceptron by multiplying any input value by a value called *weight*.
A biological neuron generates a signal only when the total strength of the input signals exceed a certain threshold. This phenomenon is modeled on a perceptron by calculating the weighted sum of the inputs to represent the total strength of the input signals, and applying a step function on the sum to determine its output. As in biological neural networks, these outputs are fed to other perceptrons. (Roberts, 2000)

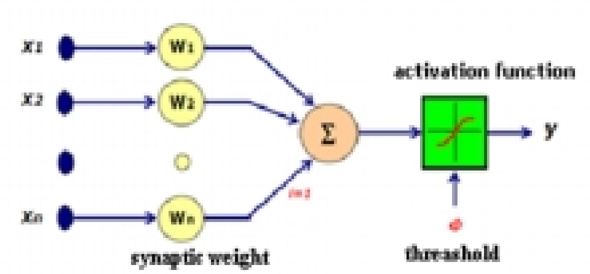

Figure 3. Elements of a perceptron

The perceptron takes some signals from its incoming links, calculates an activation level and sends it as an output signal through the output connections. The entry signal can be raw data or an output of other neurons. The perceptron calculates the weighted sum of the entry signals and compares the result with θ - the threshold value. If the neuron uses the following activation function:

$$X = \sum_{i=1}^{n} x_i w_i \qquad\qquad Y = \begin{cases} +1 & if \;\; X >=^{-} \\ -1 & if \;\; X <^{-} \end{cases}$$

Where:

X – neuron's weighted network entry

x_i - the value of entry *i*

w_i - the weight of entry *i*

n – neuron's entries number

Y – neuron's output.

If the input of the network is less than the threshold, the neuron output is -1; if the network input is greater or equal to the threshold, the neuron is activated and his output becomes +1.

An activation function performs a mathematical operation on the output signal. The activation functions that are commonly used are: the linear function, the hyperbolic tangent function, the limit value function, and the sigmoid function.

3- The activation function

The activation functions are chosen based on the problems' types solved by the network. Over the years, researchers dealt with a number of functions to convert inputs into outputs.

The activation function used in our application is the bipolar sigmoid function where the coefficient α is determined by the user. Figure 6 shows different bipolar sigmoid functions depending on the coefficient α.

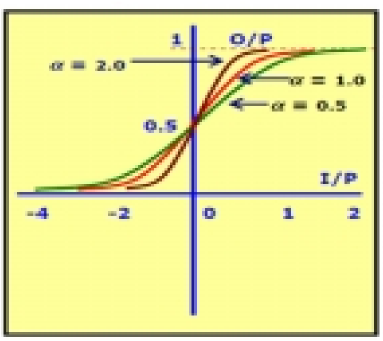

Figure 4. Bipolar sigmoid function

$$Y = f(X) = \frac{1}{1 + e^{-\alpha X}} - 1, \quad 0 \le f(X) \le 1$$

$$f'(X) = \frac{2\alpha e^{-\alpha X}}{(1 + e^{-\alpha X})^2} = \frac{\alpha(1 - f(X)^2}{2}$$

4- Architecture of the artificial neural network used

We have based the time series forecasting on a neural network architecture with 3 layers where the number of neurons in the first layer – the input layer - is defined by the user through the input parameter number, the second layer, which is a hidden layer, has the same number of neurons as the input layer and the third one - the output layer - has just one neuron as shown in Figure 4.

The network that we use is fully connected, so each neuron of one layer is connected to all neurons of the previous and subsequent layers.

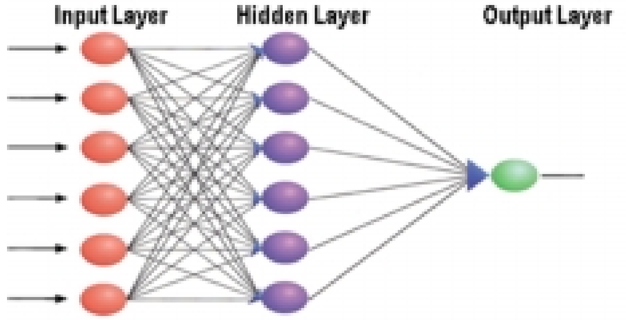

Figure 5. Architecture of NN used

We tested the neural network trained with the genetic algorithms using three consecutive selection methods: elitism, ranking and roulette wheel method, to see which of them would result more efficient in the neural network training. GANN has been compared with a neural network trained with the back propagation method to check the effectiveness of the GA in the training of neural networks versus traditional method that is BP.

The data regarding the Consumer Price Index that we have used for the model, were obtained from INSTAT for the period January 2007 to August 2012.

The results are shown in the Section 6.

5- Training a neural network with genetic algorithms

A genetic algorithm is the idea of applying the biological principles of natural evolution in the artificial systems. A genetic algorithm is an iterative procedure that includes a population of individuals, each of whom is represented by a finite string of symbols, known as genes, which encode a possible solution in the space of a given problem. This space, here referred to as the *research space*, summarizes all the possible solutions to the problem concerned.

Genetic algorithms are applied mainly in the spaces that are too large to be searched in a finite way. In neural network training the individuals are weight vectors. Their elements are generated randomly and coded as real numbers between -1 and 1, and represent the weight of each connection between neurons.

The objective of GA is the minimization of the difference between the real CPI adn the CPI forecated by the NN.

There are different selection methods to select the parents, which are going to be subject of the genetic operators, crossover and mutation, to create the next generation of weight vectors who are thought to be better neural networks for CPI forecasting. In our paper were tested three selection methods:

a. Elite Selection - first copies the best chromosome (or a few best chromosomes) to new the population. The rest is done in classical way. Elitism can very rapidly increase performance of GA, because it prevents losing the best found solution.
b. Rank Selection- first ranks the population and then every chromosome receives fitness from this ranking. The worst will have fitness 1, second worst 2 etc. and the best will have fitness N (number of chromosomes in population). After this all the chromosomes have a chance to be selected. But this method can lead to slower convergence, because the best chromosomes do not differ so much from other ones.

c. Roulette Wheel Selection -Parents are selected according to their fitness. The better the chromosomes are, the more chances to be selected they have. (Obitko, 1998)

In Figure 5 is given a block schema featuring a simple GA NN training.

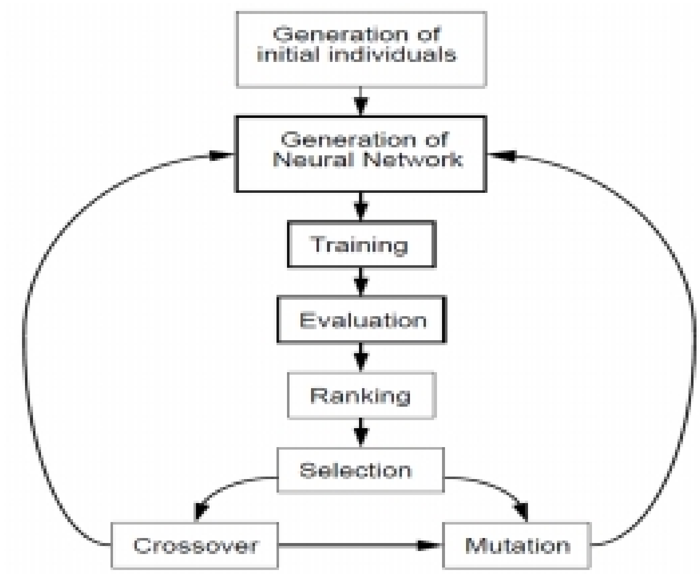

Figure 6. Using GA to train NN

6- Result

We used C# to build our application. The CPI forecasting was done by neural networks trained with genetic algorithms using different selection methods.

As shown in Figure 7, all the selection methods lead to the same accuracy in forecasting CPI time series. The neural network trained with the genetic algorithm, regardless of the selection method of the individuals, it forecasts the CPI with a very good accuracy.

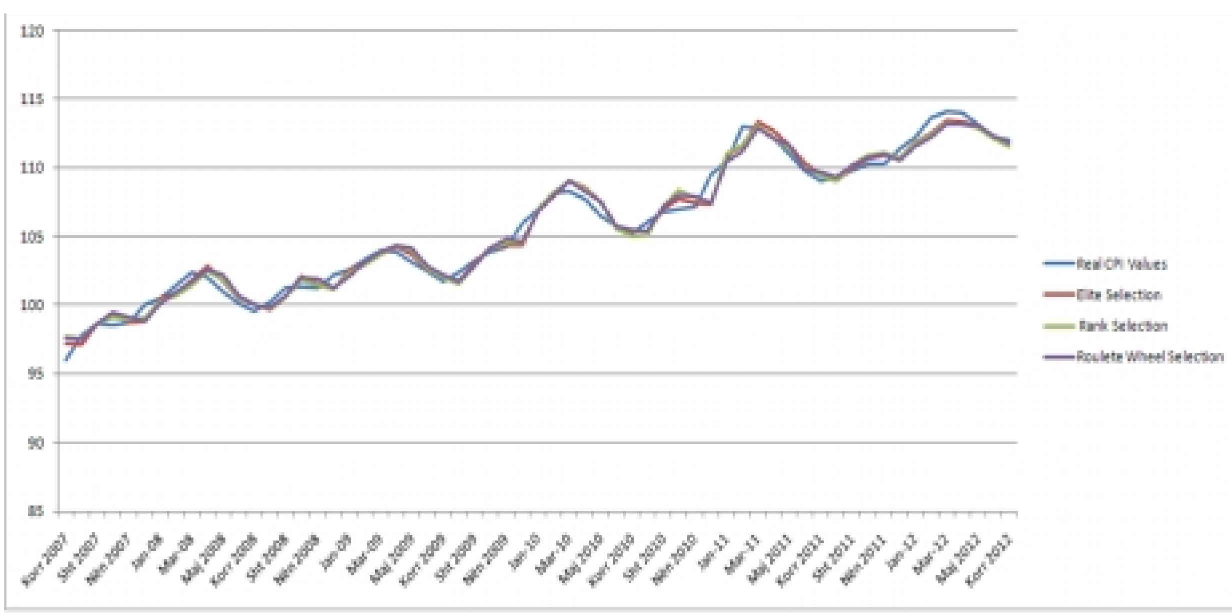

Figure 7. The results of different selection operators

From the tests performed on the training of the neural network with genetic algorithms and back propagation, the CPI forecast resulted with a much better accuracy in the first case than in the second, and this difference is noticeable, as shown in Figure 8.

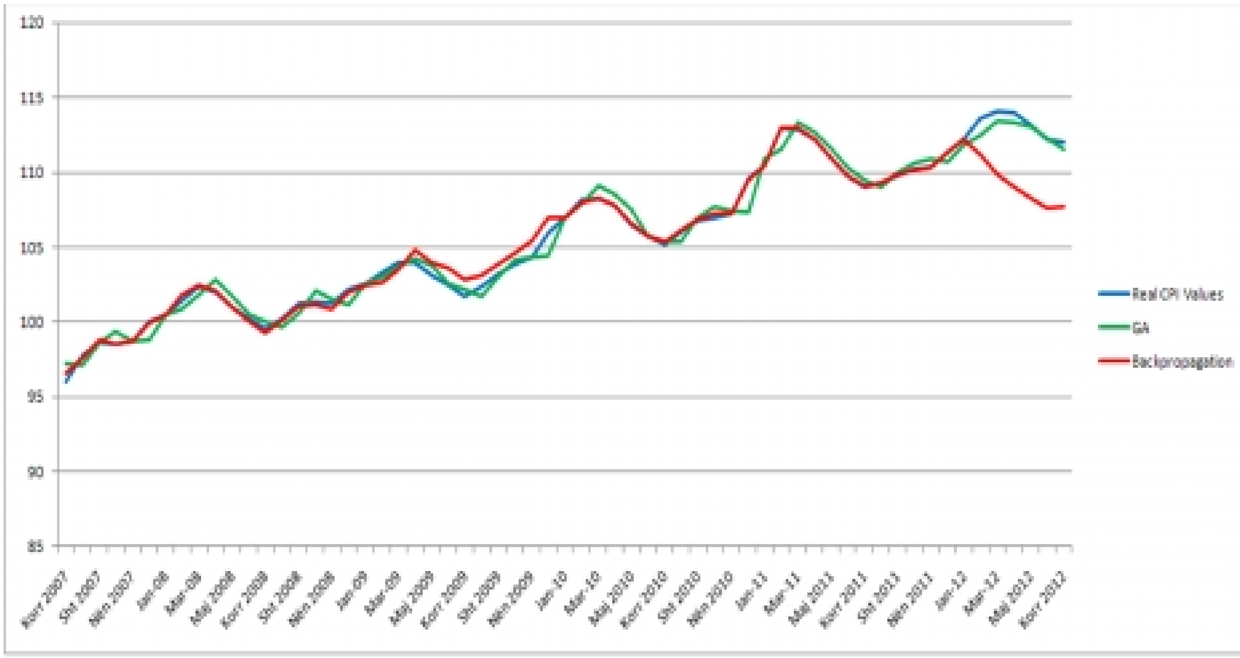

Figure 8. Results of NN with GA and NN with backpropagation

7- Conclusions

The neural networks are suitable for time series forecasting mainly due to learning from examples only, with no need to add any additional information that may bring more confusion than help in the forecasting. "Neural networks are able to generalize the problem" (Caudill & Butler, 1990) and are resistant to noise. On the other hand, it is generally not possible to determine exactly what a neural network learns, and it is also difficult to assess the potential forecasting error. However, neural networks are often successfully used in time series forecasting.

They are ideal especially when we have no other description of the observed series. Based on the results of our testing, we can say that the genetic algorithms are very efficient in training the neural networks for the CPI forecast. Furthermore, based on the results obtained, we can say that the CPI forecast is more accurate in the case of the artificial neural networks trained with the genetic algorithms rather than the neural network trained with the backpropagation method.

8- References

Bunn, D. W. (2000). Forecasting loads and prices in competitive power markets. *Proceedings of the IEEE, Vol. 88 (2)*, (pp. 163-169).

Caudill, M., & Butler, C. (1990). *Naturally Intelligent Systems.* Cambridge, Massachusetts.: MIT Press.

Foster, W. R. (1992). Neural network forecasting of short, noisy time series. *Computers Chem. Engg., Vol. 16, No. 4,*, (pp. 293-297).

Negnevitsky, M. (2005). *Artificial Intelligence: A Guide to Intelligent Systems.* Pearson Education.

Obitko, M. (1998, September). Retrieved from Introduction to Genetic Algorithms - Tutorial with Interactive Java Applets: http://www.obitko.com/tutorials/genetic-algorithms/selection.php

Plummer, E. A. (2000, July). Time series forecasting with feed-forward neural networks: guidelines and limitations.

Roberts, E. (2000). Retrieved from Neural Networks - Sophomore College : http://www-cs-faculty.stanford.edu/~eroberts/courses/soco/projects/neural-networks/Neuron/index.html

Rojas, R. (1996). A short proof of the posterior probability property of classifier neural networks. *Neural Computation* , 41-43.

Sansom, D., & Saha, T. (1999). Neural networks for forecasting electricity pool price in a deregulated electricity supply industry. *Australasian Universities Power Engineering Conference*, (pp. 214-219,). Darwin,Australia.