Testing Currency Predictability Using An Evolutionary Neural Network Model

Andreou, A. and Georgakopoulos, E. and Likothanassis, S. and Zombanakis, George A.

University of Patras, University of Patras, University of Patras, Bank of Greece

12 May 1998

Testing Currency Predictability Using An Evolutionary Neural Network Model

Andreou A.[1] , Georgopoulos E.[1] , Likothanassis S.[1] and Zombanakis G.[2]

[1] *University of Patras, Department of Computer Engineering and Informatics, Patras 26500, Greece. E-mail addresses: aandreou@ceid.upatras.gr, georgops@ceid.upatras.gr , likothan@cti.gr.*
[2] *Bank of Greece, Research Dept., 21 Panepistimiou str., 10250, Athens, Greece, tel.:+1 3235809, fax: +1 3233025.E-mail address: bogllbr@ath . forthnet . gr.*

Abstract

Two alternative learning approaches of a MLP Neural Network architecture are employed to forecast foreign currencies against the Greek Drachma, a Back-Propagation with a hyperbolic tangent activation scheme and an evolutionary trained model. Four major currency data series, namely the U. S. Dollar, the British Pound, the French Franc and the Deutsche Mark, are used in this forecasting experiment. Extended simulations have shown a high predictive ability, which is significantly better when using the actual rates compared to using the logarithmic returns of each series. The genetic algorithm performs best on FF and DM, while the back-propagation on USD and BP.

1. Introduction

The problem of modelling a system from Economics and Finance field, based on high-frequency data observed at regular intervals, attracts an increasing interest among researchers over the last few years. The goal of such an effort, is to predict, as accurately as possible, the system's future behaviour, either in a short or a long predicting horizon. Examples of such systems are the stock and foreign currency rates (Papaioannou and Karytinos, 1995;, Andreou et. al., 1997, Adamopoulos et. al., 1997).

These systems, however, are usually "contaminated" with a certain level of noise, that makes the analysis a very difficult task. Thus, one must turn to models that either filter away noise, or "machines" that can learn how the system behaves, even in the presence of noise.

The technique followed in this paper, is based on the emerging technology of artificial neural networks. This technique belongs to a class of data driven approaches, as opposed to model driven approaches. The process of constructing such a machine based on available data, is addressed by certain general purpose algorithms. The problem is then reduced to the computation of the weights of a feedforward network, to accomplish a desired input-output mapping and can be viewed as a high dimensional, non-linear system identification problem. What is of more importance, is to find the suitable network architecture that matches better to the driving information and yet maintaining good performance.

Two neural network models have been developed, based on a multilayer feedforward architecture. The first uses the well known Back-Propagation learning algorithm with a hyperbolic tangent equation as its activation function, while the second constitutes a model whose parameters are genetically altered by a supervising algorithm.

The time series data used, consist of daily exchange rates of four major foreign currencies, namely the U.S. Dollar (USD), the British Pound (BP), the Deutsche Mark (DM) and the French Frank (FF), against the Greek Drachma. The rates are those determined during the daily "fixing" sessions in the Bank of Greece. The data cover an 13-year period, from the 1st of January 1985 to the 31st of December 1997, consisting of about 3100 daily observations. This amount of data is relatively small compared to the time series used in Natural Sciences, but large enough compared to other studies in Economics and Finance, in most of which the data length merely reach 2000 observations.

The models have been tested and evaluated using two different pre-processing methods of the initial data sets:

1. The raw data, that is the actual fixing rates of each currency, rescaled and adjusted to certain boundary levels.
2. The first differences of the natural logarithms of the raw data, aiming at obtaining returns in continuous time.

The rest of the paper is organized as follows : Section 2 touches upon some notions regarding the exchange-rate policy followed in Greece during the last two decades, while section 3 involves a brief introduction to the technical terms of the methods followed. Section 4 presents the application of neural networks, where the different topologies in each of the two models mentioned above are presented and evaluated. Finally, in section 5 we present a discussion on the economic valuation of the results, while section 6 contains the concluding remarks.

2. Economic Background

Forecast the drachma exchange rate can be an interesting task in more ways than one since it is not purely market-determined. Instead, the exchange rate has traditionally been one of the favourite policy instruments used by the authorities for attaining a variety of internal (e.g. price stability) or external (e.g. current-account balance) targets, thanks to its effectiveness and the absence of political cost which its use entails.

In fact, the exchange rate has been used as a major policy instrument in Greece since March 1975, when the drachma ceased to be pegged to the US dollar in terms of a fixed exchange rate. Using the exchange-rate policy, the authorities aimed at improving or even restoring the degree of competitiveness of Greek goods and services in the international markets, restricting the balance of payments deficit or even curtailing the inflation rate, depending on the circumstances. Since 1975 and until the early 90's, the drachma had been following a depreciating trend which, in parallel to the implementation of an expansive monetary policy, has contributed to preserving high inflation rates (Bank of Greece, 1984, p.31). The competitiveness gain thanks to the January 1983 15.5% drachma devaluation in effective exchange rate terms was to its largest part wasted since up and until August of the same year the drachma was linked to the US dollar which was considerably revalued in effective exchange rate terms, thus dragging the drachma rates high up again. The continuing depreciation path of the drachma vis-à-vis both the US dollar, as well as the major European currencies since 1984, reinforced the inflationary procedure and revived inflationary expectations.

As it has been realised by the authorities later on (Bank of Greece, 1986, p.22), this had been a major problem, bringing to attention the constraints of the exchange-rate policy, indicating that such measures can not be regarded by any means as a substitute for a cost-reducing or inflation-fighting policy. Indeed this policy option proved to be rather unfortunate, since high production costs resulted to a considerable erosion of the degree of competitiveness of the Greek products in the international markets. The 15% drachma devaluation in nominal exchange rate terms on the 11th of October 1985, seemed to be more promising in terms of effectiveness, compared to the 1983 devaluation, since the restoration of the degree of competitiveness in the international markets was pursued not just on the basis of the exchange rate policy alone, but, in addition, with the backing of a strict monetary policy and an effort to curtail the budget deficit. These measures aimed at delaying the rate of increase of nominal wages and at restricting the effect of the devaluation on labour cost and prices so as to preserve, or even to improve the degree of competitiveness of the Greek goods and services (Bank of Greece, 1986, p. 29). This stabilisation program, however, did not attribute much attention to the supply side, ignoring the need for structural reforms on the microeconomic level. It also underestimated the fact that since the import component of the Greek final products is high in terms of raw materials and semi-finished products, a devaluation or a depreciation is expected to lead to increases of the prices of these products, offsetting the benefits upon price competitiveness. Thus this program was abandoned long before it had attained its targets, this leading to a revival of the competitiveness deterioration and consequently to the burdening of the balance of payments. Thus, the rapid depreciation policy from 1988 onwards kept re-feeding inflation, since the business sector relied on the state support, regarding a cost subsidy in the form of a continuous drachma depreciation, as given. As a result the supply side neglected improving its productive efficiency and was reluctant to resist wage demands (Bank of Greece, 1991, p. 33).

The increasing current account deficits accompanied by the accelerating inflation rates due to the “easy” exchange-rate policy followed from 1988 onwards, led the authorities to realise the ineffectiveness of the devaluation or even the rapid depreciation solutions. This ineffectiveness has been a fact extensively supported in the literature (Brissimis and Leventakis,1989; Karadeloglou, 1990) and quite recently, in connection to export-promoting strategies (Karadeloglou et. al., 1998; Zombanakis, 1998). Thus, the authorities focused their attention on improving productivity, curtailing production costs and adjusting the supply side to changes in demand, since the early 90’s. They have thus decided to start by following a non-accommodating depreciation policy in the sense that the rate of depreciation does not fully cover the inflationary gap between Greece and its trading partners. Additional intervention measures have also been imposed in order to face the impact of seasonal or irregular factors on the drachma exchange rate.

During the past few years, the Bank of Greece, encouraged by the restrictive action of the non-accommodating exchange-rate policy upon inflation, has opted for an even tighter version, keeping the exchange rate of the drachma vis-à-vis the ECU, fixed. This has become known as the "hard- drachma policy", aiming at weakening both inflationary pressures and expectations concerning such pressures in the future. This policy has proven to be more than successful since, its drastic anti-inflationary impact was accompanied by a significant interest-rate reduction representing a relief for the budget deficit and a decrease of the capital cost of the business sector. Thanks, also, to the "hard-drachma", increases in servicing the foreign-currency denominated public debt have been avoided, the foreign-exchange risk has been restricted, while the cost of the imported raw materials for Greek export firms, the products of most of which bear a high import component, has been held constant.

Bearing in mind the benefits of the "hard-drachma policy", the position of the Central Bank on this matter has been adamant since deviating from such tactics would imply, among other things, that the authorities ignore international long-run policy targets (Bank of Greece, 1994) and would lead to the economy missing the last train for the Euro, in January 2001. The Governor of the Central Bank, in fact, during the annual monetary program announcement for 1997 refers to the exchange-rate policy as restricted by our commitment to curb inflationary pressures announcing the intention of the authorities to keep the drachma/ECU exchange rate fixed for 1997, adding, however, that this policy will be exercised "allowing for considerably higher flexibility than what was the case in the past" (Bank of Greece, 1997). The translation of this enigmatic statement is that the exchange rate may be fixed in year-end terms, but fluctuations within the year will not be predetermined, their direction of change and magnitude depending on the inflationary pressures and the market conditions. This flexibility is expected to discourage speculative action which has been the source of a considerable degree of instability in the domestic market during the recent past.

3. Technical Background

3.1 Methodology

As stated above, the paper implements two approaches of Multilayer Feedforward Neural Networks (MLFF), using in both models the hyperbolic tangent, given by the following equation:

$$\tanh(x) = \frac{1 - \exp(-b \cdot x)}{1 + \exp(-b \cdot x)} \tag{1}$$

The first model involves training using the well known Error Back-Propagation algorithm.

The second one bases the training of the neural network on an evolutionary algorithm (modified Genetic Algorithm), which is presented in a more detailed manner in the later subsections. In general, the evolutionary training approach is divided into two major steps (Yao): the first involves the representation scheme of the neural network, and the second one is the evolution itself driven by Genetic Algorithms.

3.2 The Modified Genetic Algorithm

3.2.1 Representation

The most convinient representation scheme is, from the perspective of GAs, binary strings. The problem with binary strings is that they don't exhibit great flexibility. That's why we adopt the linked list representation of the neural network. In the string representation approach there is need for a function that codes a network to a string and another one that decodes the string back to the network, in order to measure its performance. In the current approach no such functions have been used, in order to "go" from a neural network, to a floating point or binary string that represents that network. On the contrary, the modified genetic algorithm handles directly the neural networks that are represented as graphs using linked lists. The drawback of this approach is its cost in computer memory and computation time, since it has to deal with neural networks, structures much more complicated than strings. This disadvantage is overcame by its great flexibility: one can manipulate networks with different number of neurons (nodes) in the hidden region, add or delete hidden neurons (nodes), etc. This approach is a modified version of the method introduced for the case of MLP networks in Likothanassis et. al. (1997) and Andreou et. al. (1997).

3.2.2 The Evolutionary Algorithm

The modified Genetic Algorithm (Evolution Algorithm) works as follows:
It maintains a population of individuals, $P(t)=\{i_1^t, \ldots ,i_n^t\}$ for iteration t. In our problem each individual is a MLFF neural network. Each individual represents a potential solution to the problem, and is implemented as a linked list. Every solution, (individual) i_j^t, is evaluated to give a measure of its fitness. For this purpose, the fitness function uses the inverse Mean Relative Error of the network. A new population, (iteration t+1), is created by selecting the more fitted individuals according to their fitness, (select step). Some members of the population undergo transformations (alter step) by means of "genetic" operators to form the new solutions (individuals). There are unary transformations m_i (mutation type), which create new individuals by a small change in a single individual, (change the value of a random number of weights of the network, e.t.c.), and higher order transformations c_j (crossover type), which create new individuals by combining parts from two or more individuals. Thus, the evolutionary algorithm evolves the weights of the MLFF networks in order to teach them the input to output mapping. After some number of iterations the program converges. At this point it is expected that the best individual represents a near-optimum (reasonable) solution.
In the following lines we shall briefly describe the algorithm's steps.

Step1: Initialisation

Initialisation is the first step of the algorithm. An initial population of individuals is generated, where each individual is a complete neural network with fixed architecture. The size of the population is a parameter of the algorithm. The values for the connection weights of the individuals are generated randomly, with uniform distribution, in the interval [-1,1].

Step2: Selection

The selection is an essential operation in genetic algorithms, since it constructs a new population with respect to the probability distribution based on fitness values of the individuals that belong to the previous population. We have implemented a number of different selection processes, but in our experiments we have used the Elitism Selection Operator.

The fitness function used is the following:

$$\text{Fitness} = 1/(1 + \text{MRE}), \tag{2}$$

where MRE is the value of the Mean Relative Error cost function.

Step 3: Crossover

The crossover, or recombination, operator is a very important operator, that is applied to the new population. The crossover operator generally works as follows: it selects two parents and generates one or two offspring by recombining parts of them, for the case of neural networks with stable architectures, these parts can be layers, neurons or just weights. According to this, three different crossover operators were implemented. The first one creates a child by recombining whole layers from the two parents, while the second by recombining neurons with their connections (each child's neuron is chosen with equal probability either from parent1 or parent2) and the third works as the former one, with the only difference being that now it deals with connection weights and not neurons.

Step4: Mutation

The mutation process is proved to be a very significant operator that greatly influences the convergence of the algorithm. Two different mutation operators were implemented: the first operator, selects randomly a number of connection weights and changes their values to random ones in the interval [-1,1] (using a uniform probability distribution). The second mutation operator, is responsible for the fine tuning capabilities of the system. It selects randomly a number of connection weights and change their values to new ones as follows: Let us suppose that w is the old weight value then the new one is given by the formula:

$$w' = w + \Delta(t, ub-w), \quad \text{if a random digit is 0,} \tag{3}$$
$$w' = w - \Delta(t, w-lb), \quad \text{if a random digit is 1,} \tag{4}$$

where lb and ub are the lower and upper bounds of the weight values, which in our algorithm are -1 and 1 respectively, t is the generation number, and $\Delta(t,y)$ is a function that returns a value in the range [0, y], such that the probability of $\Delta(t,y)$, being close to 0, increases according to t . This property causes this operator to search the solution space initially uniformly (while t is small) and very locally at later stages. In our experiments the following function (Michalewicz, 1994) has been used:

$$\Delta\left(t, y\right) = y \cdot \left(1 - r^{\left(1-\frac{t}{T}\right)^{b}}\right), \qquad (5)$$

where r is a random number in [0,1], T is the maximal generation number, and b is a system parameter determining the degree of non-uniformity.

4. Empirical Results

In this section we present the some representative results obtained from the simulations runs. At fist, the classic MLP with a Back-Propagation learning algorithm is evaluated and the Genetically altered MLP follows. Performance was evaluated using two different types of measures, specifically the well known Correlation Coefficient (CC) and the Mean Relative Error (MRE). These measures were applied during the out-of-sample phase, that is, on a set of unseen data after completion of the networks learning phase. The CC provides for a qualitative measure of how close the series produced as a forecasted output of a network is to the original one, that is, whether it follows any ups or downs. MRE on the other hand shows the of accuracy of predictions, expressed on the actual value of the sample being predicted. Thus, we are able to estimate prediction error as a fraction of the actual value. MRE is given by the following equation:

$$\text{MRE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{o_i - d_i}{o_i}\right| \qquad (10)$$

where o_i is the output of the network and d_i the actual (desired) value.

4.1 MLP with Error Back-Propagation

4.1.1 Raw Data:

Table 1 summarizes the results of the simulations conducted on the raw data sets, where multiple topologies, different lengths of training and testing sets and numbers of iterations have been used. According to Table 1, prediction on all currencies performed quite well. The CC measure shows a very good to excellent follow-up of the actual series while accuracy, according to MRE, reaches a level of 86 to 90%. USD and BP accuracy of forecasts is slightly better than those of the FF and DM (see MRE). Figures 1 through 4 present graphically the best topology of each currency.

Table 1: Empirical results using MLP with Back-Propagation on the raw data set

Currency	Topology*	Epochs	Training Patterns	Testing Patterns	CC	MRE
USD	5-0-1	900	500	200	0.959	0.081
	5-4-1	900	500	200	0.953	0.379
	5-10-1	900	500	200	0.955	0.361
BP	5-0-1	900	1000	200	0.840	0.046
	5-4-1	900	1000	200	0.831	0.032
	5-10-1	900	1000	200	0.828	0.031
FF	5-0-1	625	800	200	0.980	0.155
	5-4-1	625	800	200	0.975	0.170
	5-10-1	625	800	200	0.975	0.171
DM	5-0-1	900	700	200	0.960	0.164
	5-4-1	625	700	200	0.956	0.183
	5-10-1	625	700	200	0.950	0.181

* "n-p-m" stands for n input nodes, p nodes in the hidden layer and m output nodes

In order to achieve best forecasting performance, we varied the number of patterns used in the training set and the training period (epochs) too. As Table 1 shows, the diversification of these two factors holds not only for different currencies, but for different topologies used to predict the same currency as well.

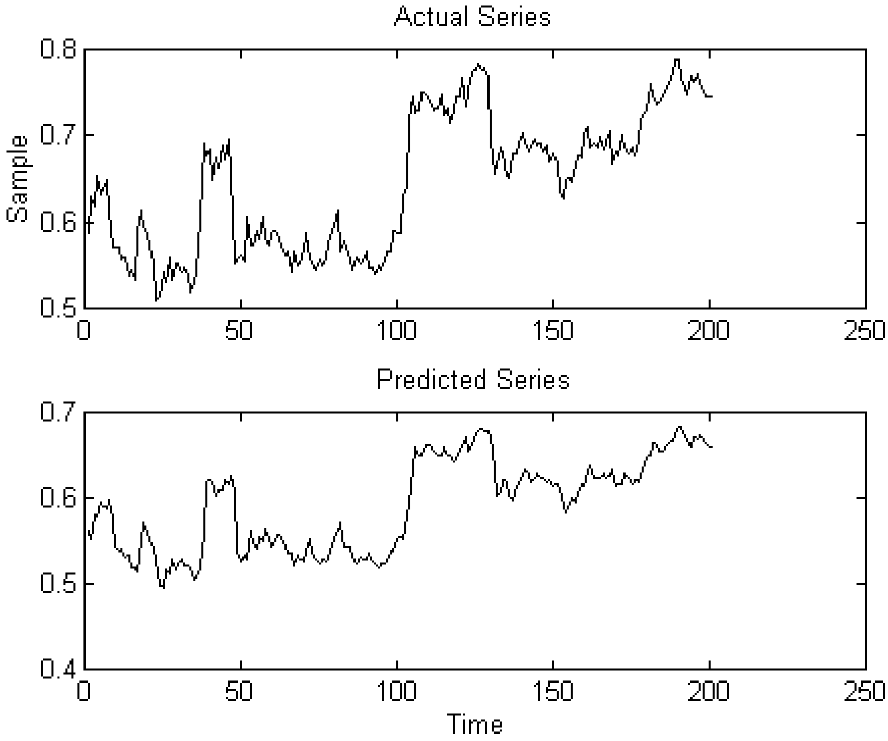

Figure 1: USD/GRD raw data of an MLP with Back-Propagation and a 5-0-1 topology

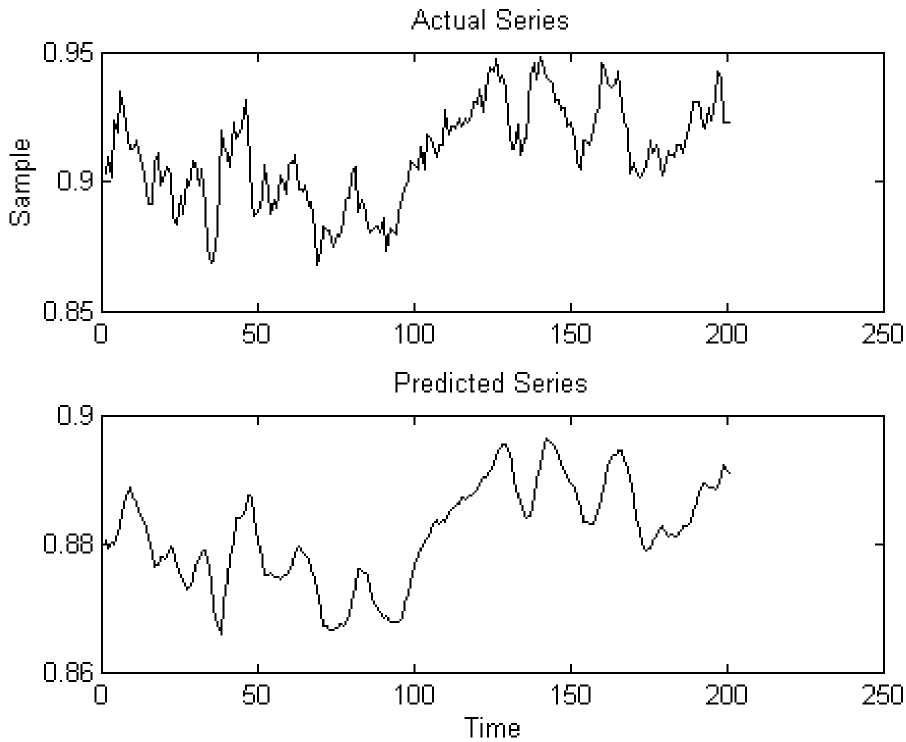

Figure 2: BP/GRD raw data of an MLP with Back-Propagation and a 5-10-1 topology

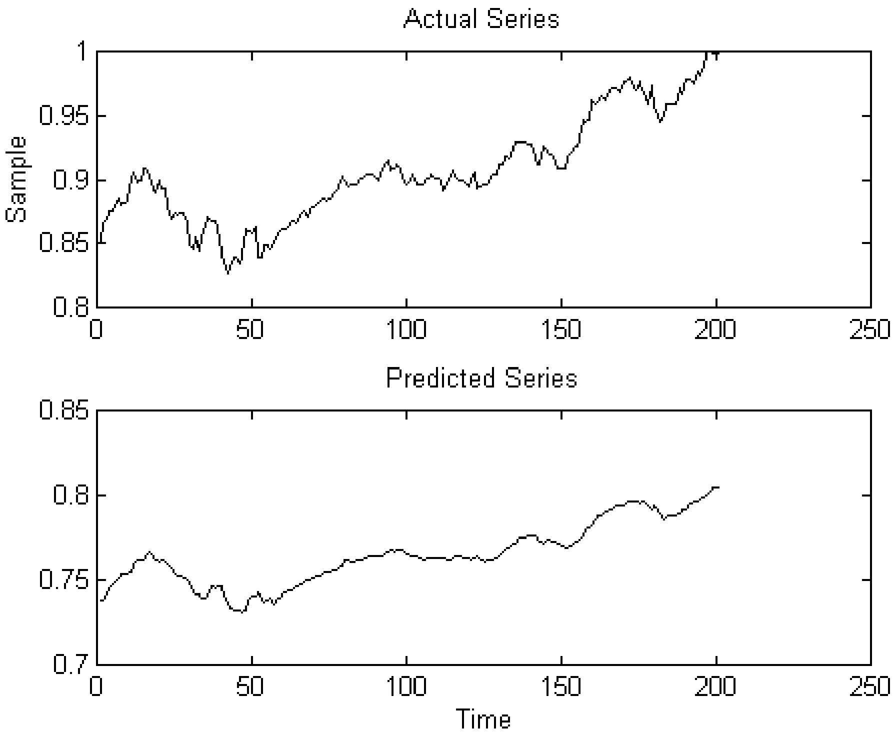

Figure 3: FF/GRD raw data of an MLP with Back-Propagation and 5-0-1 topology

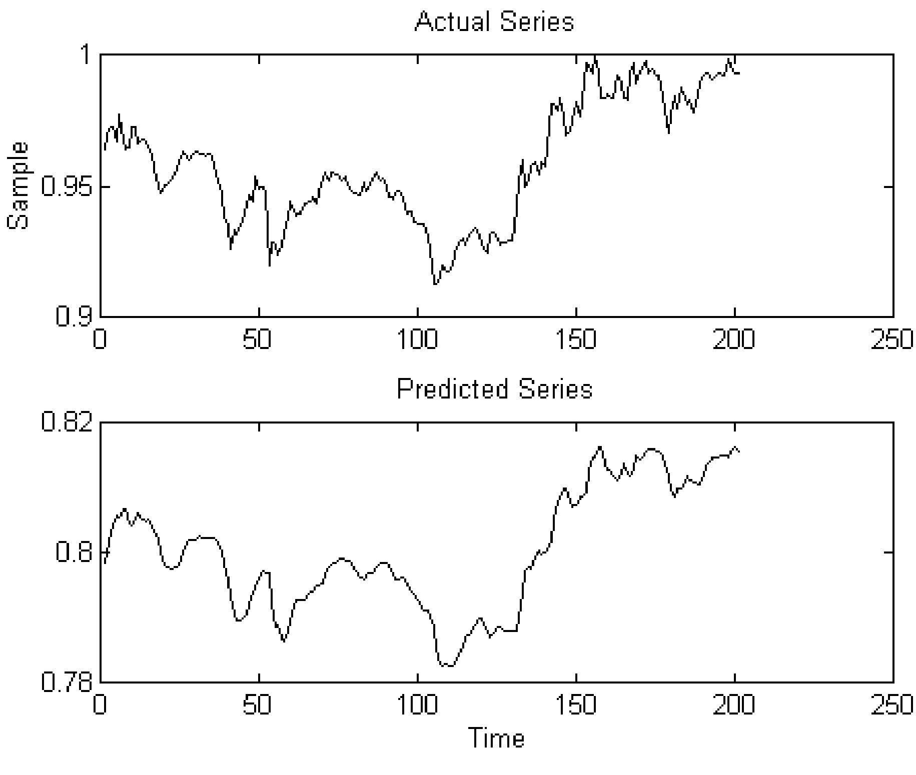

Figure 4: DM/GRD raw data of an MLP with Back-Propagation and a 5-0-1 topology

4.1.2. Logarithmic Data

Table 2 presents the empirical results obtained when using the logarithmic set of data of each currency. It is quite obvious that irrespective of the fact that MRE errors stay very low (except in the DM case), the CC's prove the inability of networks to produce a predicted series such as to follow any ups or downs of the original one, or in a more optimistic sense, it follows with a substantial upward or downward displacement. This is confirmed through Figures 5 to 8, where the predicted series "catches" the actual one in some scattered parts only, with a discernible shifting. It is also worth noticing that when it comes to DM the accuracy of forecasting is very poor, indicating a diversification from the other three currencies.

Table 2: Empirical results using MLP with Back-Propagation on the logarithmic data set

Currency	Topology*	Epochs	Training Patterns	Testing Patterns	CC	MRE
USD	5-0-1	625	900	200	0.001	0.077
	5-4-1	900	900	200	0.022	0.077
	5-10-1	900	900	200	0.001	0.077
BP	5-0-1	900	700	200	0.027	0.028
	5-4-1	900	700	200	0.013	0.028
	5-10-1	900	700	200	0.009	0.028
FF	5-0-1	625	900	200	0.054	0.028
	5-4-1	625	900	200	0.063	0.027
	5-10-1	625	900	200	0.012	0.027
DM	5-0-1	625	700	200	0.074	1.232
	5-4-1	900	700	200	0.018	1.250
	5-10-1	900	700	200	0.063	1.316

* "n-p-m" stands for n input nodes, p nodes in the hidden layer and m output nodes

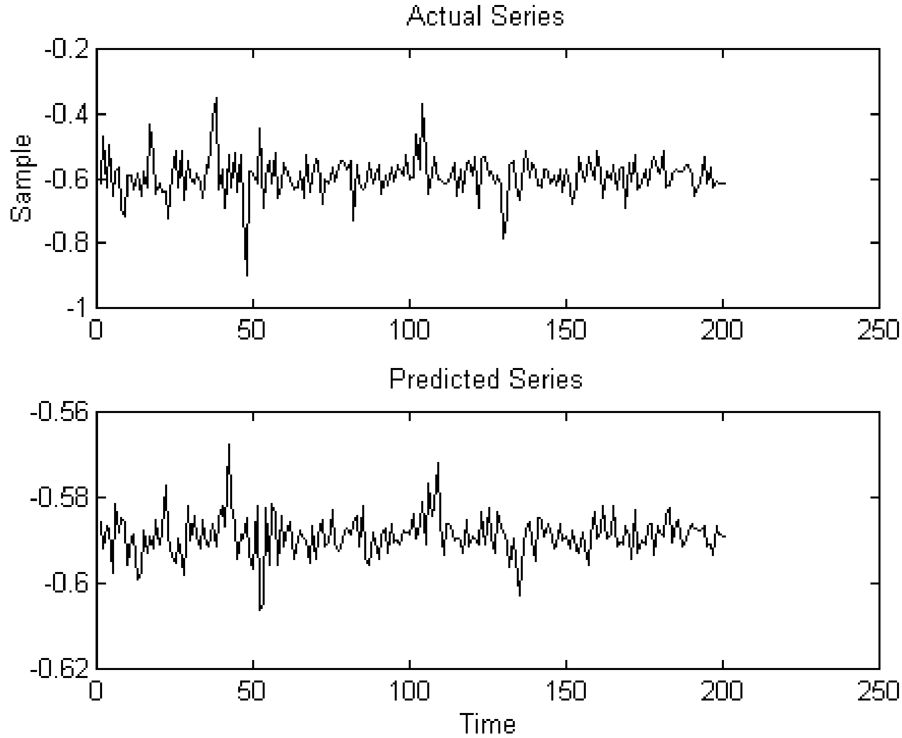

Figure 5: USD/GRD raw data of an MLP with Back-Propagation and a 5-4-1 topology

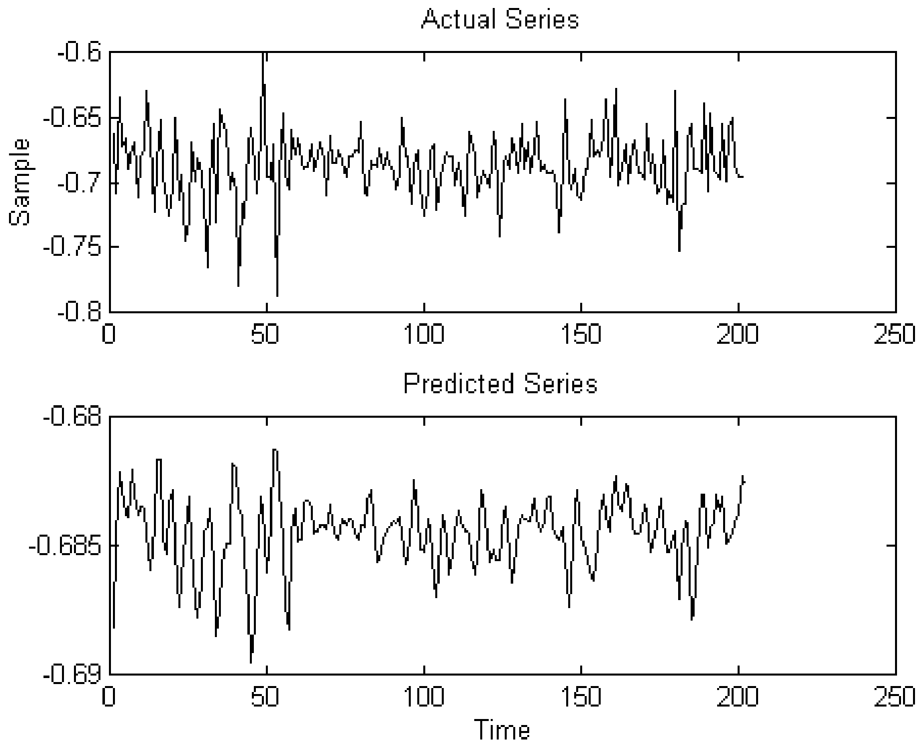

Figure 6: BP/GRD raw data of an MLP with Back-Propagation and a 5-0-1 topology

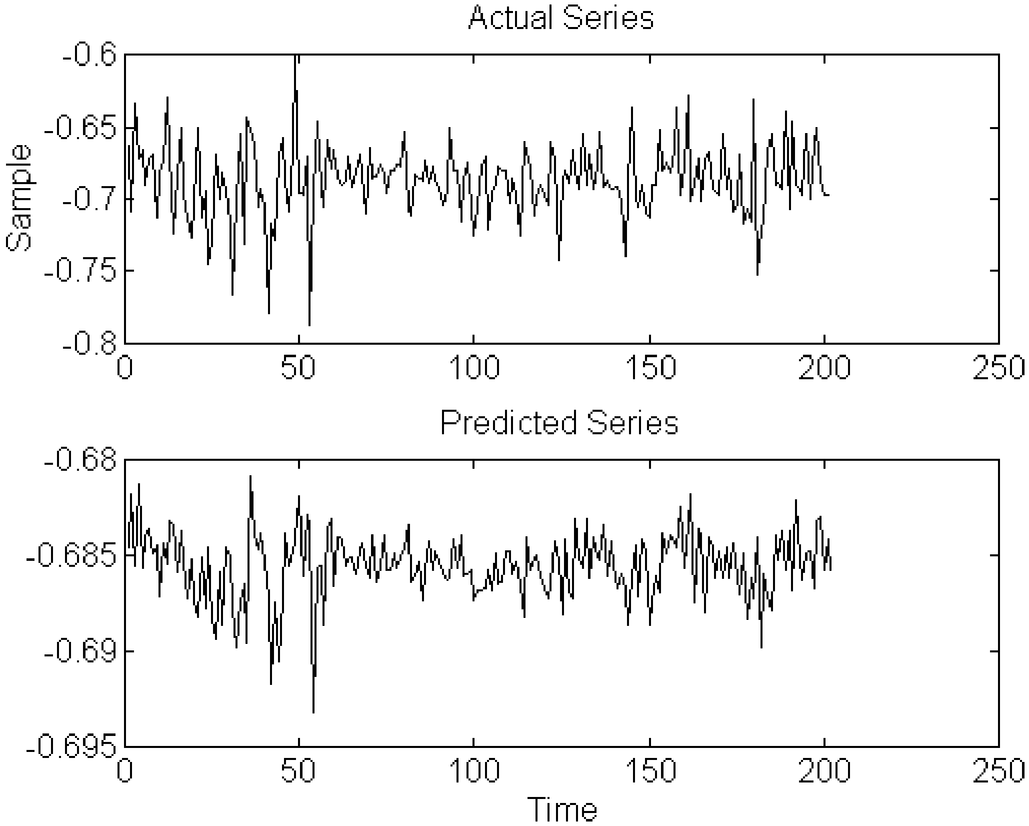

Figure 7: FF/GRD raw data of an MLP with Back-Propagation and a 5-4-1 topology

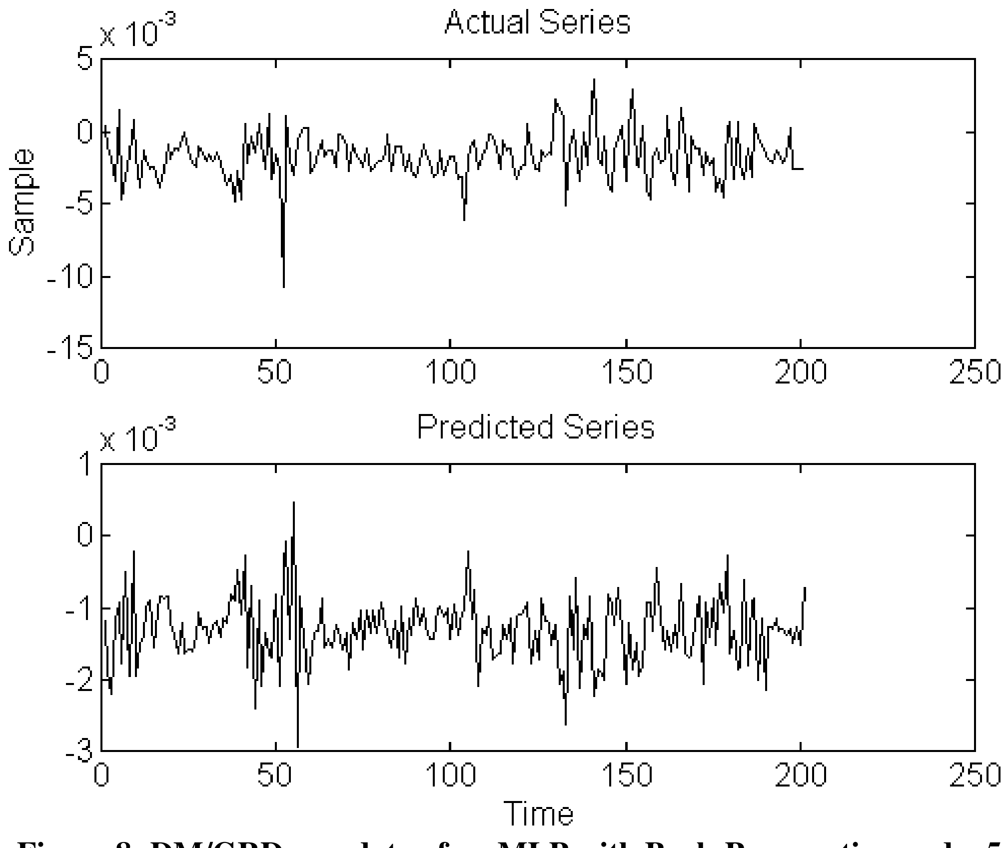

Figure 8: DM/GRD raw data of an MLP with Back-Propagation and a 5-0-1 topology

4.2 MLP with a Genetic Algorithm

All the numerical experiments conducted using the genetically trained neural network model, were executed using the same values for all the genetic parameters: Population Size =50 individuals, number of Generations =2000, number of training patterns =300 and number of testing patterns =100. Table 3 shows the results regarding the performance of the algorithm for a number of different architectures.

4.2.1 Raw Data

Simulation runs on the raw data sets using the Genetically altered MLP are shown in Table 3. As in the case of hyperbolic tangent MLP, the CC's reach almost 100%, while MRE's prove a very good to excellent accuracy of forecasts. The best prediction performance was yielded by DM and FF, with BP being slightly inferior. Figures 9 to 12 present a graphical view of these results. Direct comparison of these results to those depicted in Table 1, prove a slight inferiority. It should be noticed however that the training set used by the genetic algorithm is smaller than the one used by the back-propagation, due to some implementation limitations, discussed at the end of this section.

Table 3: Empirical results using MLP with a Genetic Algorithm on the raw data set

Currency	Topology*	CC	MRE
USD	5-0-1	0.983	1.593
	5-4-1	0.949	0.835
BP	5-0-1	0.986	0.110
	5-4-1	0.962	0.087
FF	5-0-1	0.998	0.051
	5-4-1	0.994	0.350
DM	5-0-1	0.998	0.059
	5-4-1	0.993	0.019

* "n-p-m" stands for n input nodes, p nodes in the hidden layer and m output nodes

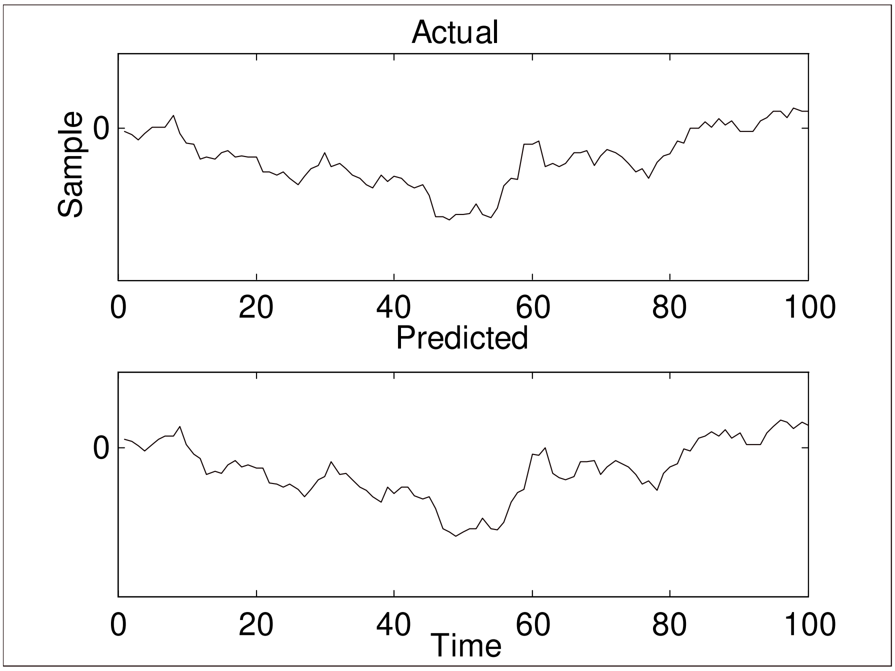

Figure 9: USD/GRD raw data of an MLP with a genetic algorithm and a 5-4-1 topology

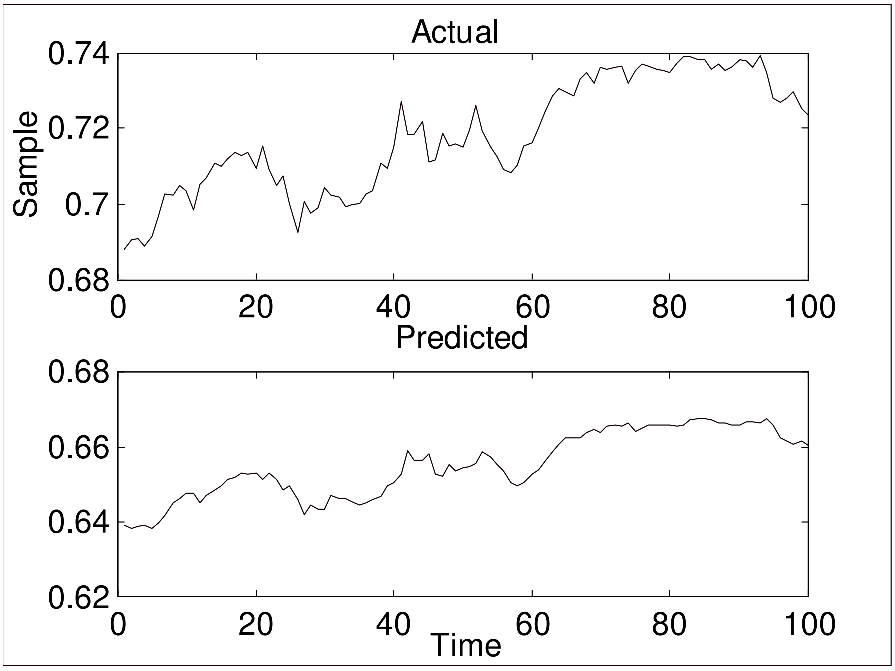

Figure 10: BP/GRD raw data of an MLP with a genetic algorithm and a 5-4-1 topology

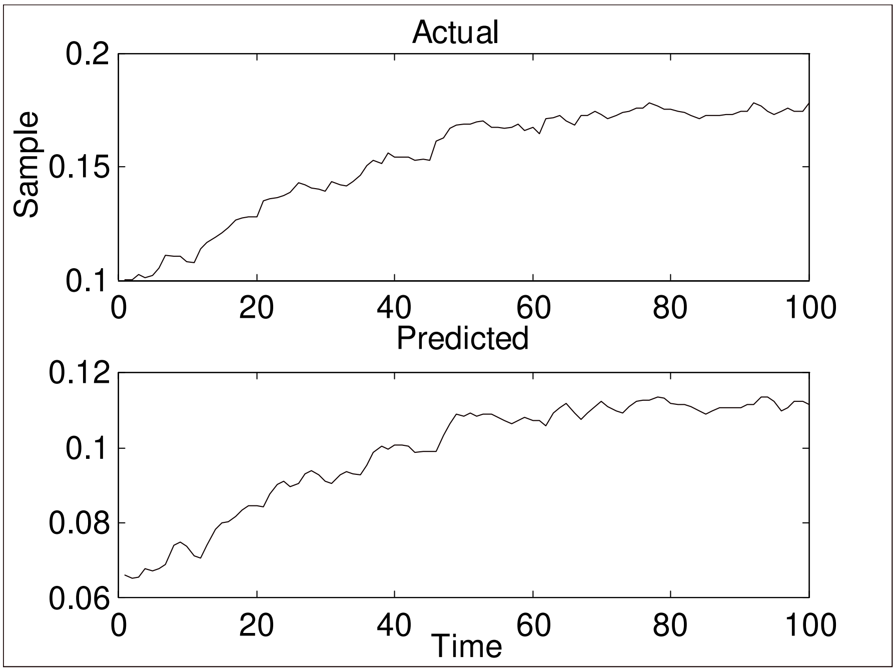

Figure 11: FF/GRD raw data of an MLP with a genetic algorithm and a 5-0-1 topology

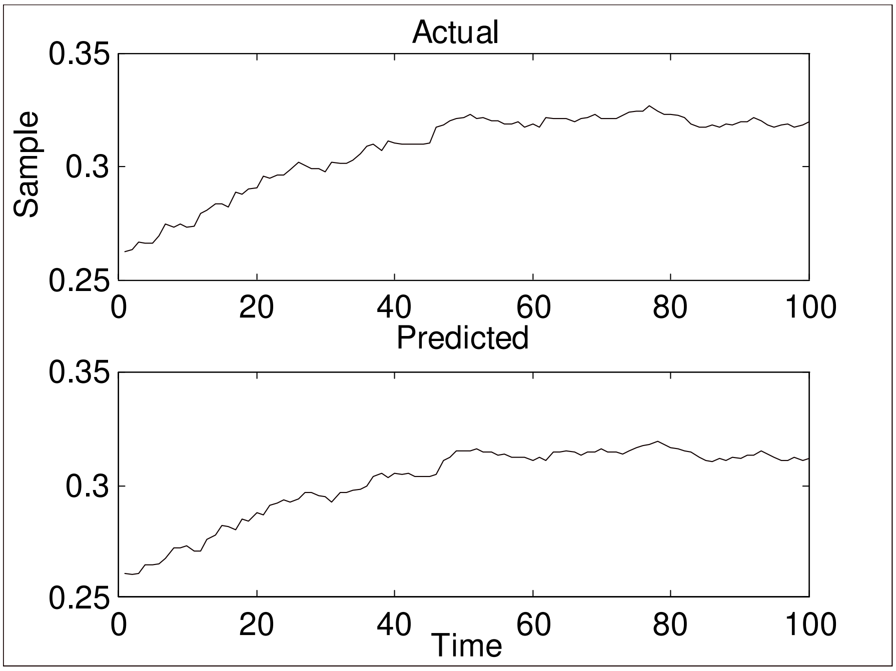

Figure 12: DM/GRD raw data of an MLP with a genetic algorithm and a 5-4-1 topology

4.2.2. Logarithmic Data

Table 4 presents the results on the logarithmic set of data of each currency. The picture here is similar to that of the hyperbolic tangent when using the same data set. It is quite obvious here too that despite the fact that MRE errors stay very low (except in the DM case), the CC's prove the failure of the predicted series to follow up the original one in either upward or downward "spikes", or sample values scaling. This is confirmed through Figures 13 to 16, where the predicted series "catch" a few upward or downward jumps of the actual series and are shifted by a certain factor. It is also noticeable that when it comes to DM the accuracy of forecasting is very poor, indicating a diversification from the other three currencies, just like in the back-propagation case.

Table 4: Empirical results using MLP with a Genetic Algorithm on the logarithmic data set

Currency	Topology*	CC	MRE
USD	5-0-1	0.028	0.109
	5-4-1	0.046	0.111
BP	5-0-1	0.076	0.241
	5-4-1	0.162	0.241
FF	5-0-1	0.060	0.014
	5-4-1	0.139	0.015
DM	5-0-1	0.043	0.231
	5-4-1	0.046	0.231

* "n-p-m" stands for n input nodes, p nodes in the hidden layer and m output nodes

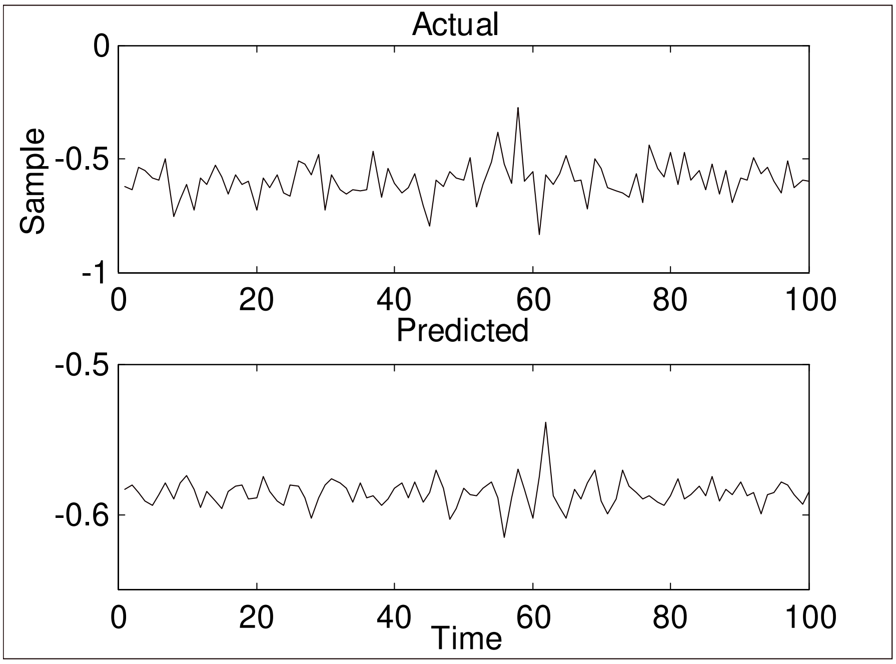

Figure 13: USD/GRD logarithmic data of an MLP with a genetic algorithm and a 5-4-1 topology

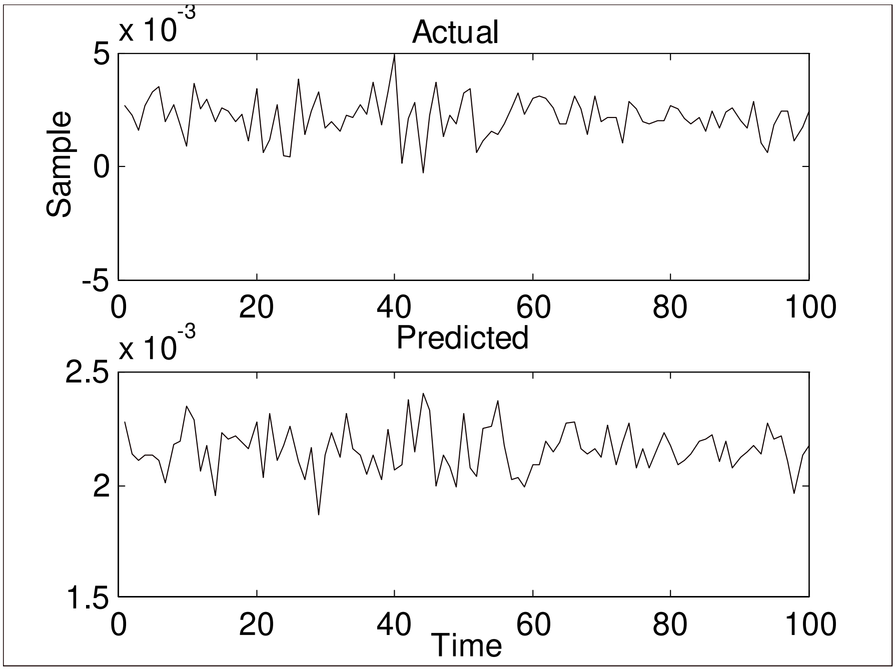

Figure 14: BP/GRD logarithmic data of an MLP with a genetic algorithm and a 5-4-1 topology

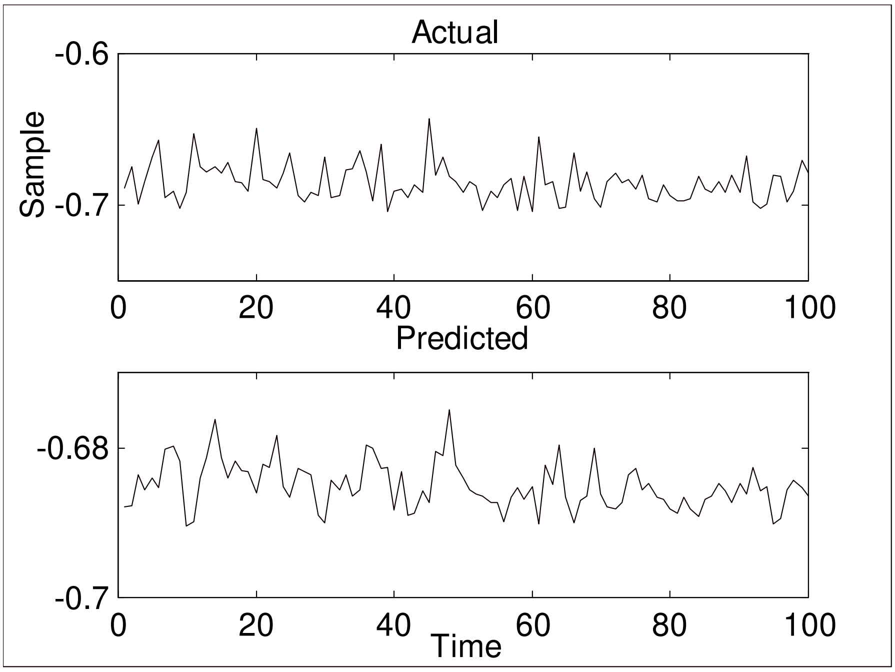

Figure 15: FF/GRD logarithmic data of an MLP with a genetic algorithm and a 5-4-1 topology

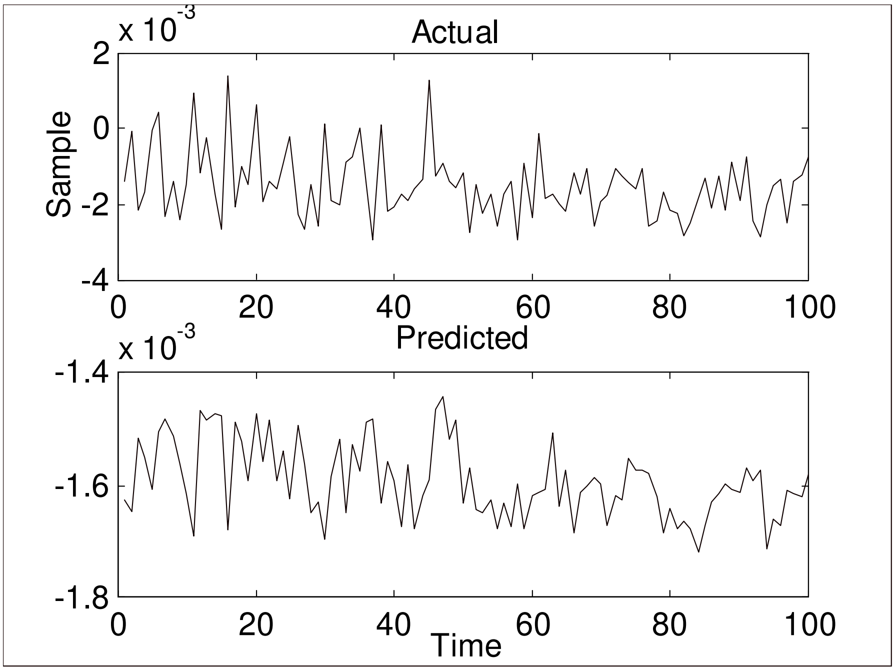

Figure 16: DM/GRD logarithmic data of an MLP with a genetic algorithm and a 5-4-1 topology

4.3 Remarks on Simulations

At this point we would like to justify some differences observed in performance between the Back-Propagation (BP) and the Genetic Algorithm (GA) learning techniques as follows:

1. There is a significant difference in the number of training patterns between the BP and GA. The use of only 300 training patterns for the case of GA is explained by the fact that the GA evolves simultaneously a big number of neural networks while the BP works with only one network. Thus, GA has much more computer memory needs than the simple BP and the use of a big training set has a great influence on the GA's convergence speed. Currently we conduct experiments with bigger training sets.
2. GA was executed using a relatively small population size (50 members) and a quite small number of generations (2000), because of hardware limitations (experiments have been performed on a PC).
3. Apart from the above, the GA used is a very general one, and further modifications and improvements can be made to the genetic operators and parameters.
4. An important thing one should notice is the fact that although this GA is of a general kind its forecasting performance proves to be similar to the performance of the BP (an enhanced version of BP), and in some cases it outperforms.

5. Economic Evaluation of Results

The technical analysis followed in this paper has yielded results which are to a considerable extent anticipated and lead to plausible conclusions as regards the predictability of the behaviour of the drachma fluctuations versus the four currencies involved, namely the US dollar, the British Pound, the Deutsche Mark and the French Franc. Both algorithms perform successfully as regards the prediction of the exchange rate levels of all currencies, which is not the case for the logarithmic version of the data despite its satisfactory MREs, given that the correlation coefficients obtained are very low. At the risk of being diverted to technical discussions we can not help recalling that traditional econometric techniques often show preference for logarithmic functions when it comes to equation specifications, frequently without any theoretical justification. This is a complication which the use of the neural networks helps to avoid since it seems that it performs much better in the case of the levels of the exchange rates.

The prediction performance in the case of the MLP with Back-Propagation is very successful for all four drachma rates. The same applies for the case of the MLP using a Genetic Algorithm, with the exception of the Drachma / US Dollar rate.. This seems reasonable to support because the US Dollar exhibits more pronounced fluctuations in the international markets since it is subjected to exchange-rate discipline to a much lesser extent, unlike the rest of the currencies, which are ECU participants, two of them being ERM members. On the other hand, the drachma exchange rate policy is to a large extent and has, for a considerably long time period, been determined on the basis of the ECU as indicated earlier on in this paper. It provides for increased discipline for the drachma fluctuations versus all ECU participant currencies, which include all currencies involved in the analysis except the US dollar. This leads to expecting the predictability of the rate of the drachma versus the US dollar to be reduced compared to that of the drachma rates vis-à-vis the rest three currencies due to the absence of any discipline framework in this case. Thus, the results obtained seem very reasonable, particularly in the case of the Genetic Algorithm: Very good prediction results for the three ECU participants, and even better ones for the DM and the FF which are, in addition, ERM members, a fact which introduces additional discipline in the relevant exchange rates, the fluctuations of which are bound by the ERM band. The Pound, on the contrary has participated in the ERM only for a short time period until its membership suspension on the 16th of September 1992, a period too short to provide for disciplined behaviour in the sample period chosen.

The poor performance of the drachma / dollar rates, however, was not the case with the Back-Propagation algorithm, which, has provided for a remarkably low MRE on just one occasion. This exceptional case may be the result of a wide variety of factors most important of which is the varying number of training patterns in this case as opposed to the constant number of training patterns in the case of the Genetic Algorithm, something which is a decisive determinant of the prediction performance of the network.

6. Conclusions

This paper has employed two advanced artificial learning approaches to an MLP Neural Network architecture, a Back-Propagation and a Genetic Algorithm, based on a hyperbolic tangent activation scheme, in order to forecast the drachma exchange rates versus four major currencies, namely the US Dollar, the Deutsche Mark, the French Frank and the British Pound.

In general, despite the difficulties encountered due to the nature of the Greek exchange rate policy and the sample period in use which includes major noise elements, like a drachma devaluation, the results have been proved very satisfactory, since they have provided for successful exchange rate predictions in all four cases with both algorithms.

Specifically, two pre-processed data sets have been used: The first includes the raw data, that is, the fixing rates of each currency rescaled, and the second involves with the logarithmic returns. Both the methods have exhibited very high quantitative and qualitatively accuracy of forecasting as regards the former, where results have provided for very small MRE's on one hand (accurate forecasts) and high Correlation Coefficients on the other (very close follow-up of the actual series). The latter case resulted to poor performance for both methods, regarding mostly the CC, where while accuracy was good in some cases, the predicted series were either shifted upwardly or downwardly compared to the actual ones, or failed to follow-up for the entire forecasting period.

The conclusion, therefore, is that the promising results obtained for the raw data have been as expected and the explanation lies with the degree of discipline of behaviour that characterises the fluctuations of each of the currencies involved with respect to the drachma. In three out of four cases, with the exception of the dollar case, the drachma factor introduces a considerable degree of consistency in its behaviour thanks to its exchange rate versus the ECU which is a policy guideline. The increased predictability in the case of the DM and the FF comes as a result of their ERM membership, which, unlike the case of the Pound, provides for fluctuation bands and consequently for increased predictability.

The pending issue of the variation in the number of the training patterns indicates the need for further research on the topic, concentrating on modifying the algorithms used so as to indicate the number of training and testing patterns that yield the best prediction results. Our efforts will concentrate on developing a genetic algorithm that will supervise the selection of the most appropriate length for both the training and the testing sets, so as to improve forecasting regarding this highly sensitive performance factor.

7. Acknowledgement

The authors would like to thank John Gourdoulis, currently a student at the Department of Computer Engineering & Informatics, University of Patras, for his help in performing the simulations.

8. References

- Adamopoulos, A., Andreou, A., Georgopoulos E., and Likothanassis, S. (1997) "Currency forecasting using recurrent RBF networks optimized by genetic algorithms". Proceedings of the Fifth International Conference on Computational Finance, London Business School, London, 1997.
- Andreou, A., Georgopoulos, E., Likothanassis, S. and Polidoropoulos, P. (1997) "Is the Greek foreign exchange-rate market predictable? A comparative study using chaotic dynamics and neural networks". Proceedings of the Fourth International Conference on Forecasting Financial Markets, Banque Nationale de Paris and Imperial College, London.
- Bank of Greece (1984) "Report of the Governor", Athens.
- Bank of Greece (1986) "Report of the Governor", Athens.
- Bank of Greece (1991) "Report of the Governor", Athens.
- Bank of Greece (1994) "Report of the Governor", Athens.
- Bank of Greece (1997) Annual Monetary Program Announcement, Athens.
- Brissimis S. N. and J. A. Leventakis (1989) "The Effectiveness of Devaluation: A General Equilibrium Assessment with Reference to Greece", Journal of Policy Modelling, 11(2), pp.247-271.
- Karadeloglou P. (1990) "On the Existence of an Inverse J-curve", Greek Economic Review, V.12,2, p.p. 285-305..
- Karadeloglou P., C. Papazoglou, G. A. Zombanakis (1998) "Is the Exchange Rate An

Effective Anti-Inflationary Policy Instrument?", Economia (Forthcoming).

- Likothanassis, S. D., Georgopoulos, E. and Fotakis, D. (1997) "Optimizing the Structure of Neural Networks using Evolution Techniques", Proceedings of the 5th Intern. Conf. on HPC in Engineering, Spain, 4-7 July.
- Michalewicz, Z. (1994) "Genetic Algorithms + Data Structures = Evolution Programs", Second Ext. Edition, Springer Verlag.
- Papaioannou, G. and Karytinos, A. (1995) Nonlinear time series analysis of the stock exchange: The case of an emerging market. International Journal of Bifurcation and Chaos 5, 1557-1584.
- Yao, X., "A Review of Evolutionary Artificial Neural Networks", International Journal of Intelligent Systems, vol. 8, pp. 539 – 567.
- Zombanakis G. A. (1998) "Is The Greek Exporters' Price Policy Asymmetric?", Greek Economic Review (Forthcoming).