International Journal of Business, Human and Social Sciences
ISSN: 2517-9411
Vol:4, No:4, 2010

Application of Feed-Forward Neural Networks Autoregressive Models with Genetic Algorithm in Gross Domestic Product Prediction

E. Giovanis

Abstract**—**In this paper we present a Feed-Foward Neural Networks Autoregressive (FFNN-AR) model with genetic algorithms training optimization in order to predict the gross domestic product growth of six countries. Specifically we propose a kind of weighted regression, which can be used for econometric purposes, where the initial inputs are multiplied by the neural networks final optimum weights from input-hidden layer of the training process. The forecasts are compared with those of the ordinary autoregressive model and we conclude that the proposed regression's forecasting results outperform significant those of autoregressive model. Moreover this technique can be used in Autoregressive-Moving Average models, with and without exogenous inputs, as also the training process with genetics algorithms optimization can be replaced by the error back-propagation algorithm.

Keywords**—**Autoregressive model, Feed-Forward neural networks, Genetic Algorithms, Gross Domestic Product

I. Introduction

EMPIRICAL analysis in macroeconomics as well as in financial economics is largely based on times series. The existence of unexpected shocks or innovations to the economy plus measurement errors, strongly suggest that economic variables are stochastic. This approach allows the model builder to use statistical inference in constructing and testing equations that characterize relationships between economic variables. A forecast might be judged successful if it is close to the outcome but that judgment may also depend on how close it is measured. Depending upon the degree of forecast uncertainty, forecasts may range from being highly informative to utterly useless for the tasks at hand. A measure of forecast uncertainty provides an assessment of the expected or predicted uncertainty of the forecast errors which helps to qualify the forecasts themselves and to give a picture of the expected range of likely outcomes.

Aryal and Yao-Wu [1] applied a MLP network with 3 hidden layers to forecast the Chinese construction industry and they compare the forecasting performance of the MLP networks with that of ARIMA. and they found that the RMSE of the MLP estimation is 49 percent lower than the ARIMA counterpart. Li *et al*. [2] applied an AR model containing Autoregressive Moving Average and Generalized Autoregressive Conditional Heteroscedastic process combined with Generalize Regression Neural Network (GRNN) suggesting that the forecasting performance is improved considerably in comparison with simple Generalized Autoregressive Conditional Heteroscedastic processes. Swanson and White [3]-[4] applied neural networks to forecast nine seasonally adjusted US macroeconomic time series and they found generally neural networks outperform the linear models, while Tkacz [5] has found that neural networks produce lower forecasting errors for the yearly growth rate of the real Canadian GDP relative with the linear and univariate models.

In this paper we compare the forecasting performance of Autoregressive (AR) and Feed-Forward Neural Networks Autoregressive (FFNN-AR) models in the case of Gross Domestic Product growth rate. The optimization training is done based on genetic algorithms. A problem with backpropagation optimisation is that it can be trapped in a local minimum of a nonlinear objective function, because it is derivative based algorithm. Genetic algorithms are derivative-free, stochastic optimisation methods, and therefore less likely to get trapped. They can be used to optimise both structure and parameters in neural networks. The structure of the paper has as follows. In section II we present the methodology of stationarity and unit root tests, as well as the estimating and forecasting procedure for both Autoregressive and FFNN Autoregressive models. In section III the frequency and the type of data are described. In section IV the estimated and forecasting results are reported, while in the last section the concluding remarks of this study are presented.

II. METHODOLOGY

A. Unit Root and Stationary Tests

It is possible that the variables are not stationary in the levels, but probably are in the first or second differences. To

Eleftherios Giovanis is with the Royal Holloway University of London, department of Economics, Egham, Surrey TW20 0EX, UK, e-mail: giovanis@freemail.gr, Eleftherios.Giovanis.2010@live.rhul.ac.uk

be specific we confirm this assumption by applying Augmented Dickey-Fuller-ADF [6] and KPSS stationary test [7]. The ADF test is defined from the following relation:

$$\Delta y_t = \mu + \gamma y_{t-1} + \phi_1 \Delta y_{t-1} + ... + \phi_p \Delta y_{t-p} + \beta t + \varepsilon_t \quad (1)$$

, where y_t is the variable we examine each time. In the right hand of (1) the lags of the dependent variable are added with order of lags equal with p. Additionally, (1) includes the constant or drift μ and trend parameter β. The disturbance term is defined as ε_t. In the next step we test the hypotheses:

H_0: φ=1, β=0 => $y_t \sim I(0)$ with drift

against the alternative

H_1: |φ|<1 => $y_t \sim I(1)$ with deterministic time trend

The KPSS statistic is then defined as:

$$KPSS = T^{-2} \sum_{t=1}^{T} s^2{}_t / \hat{\sigma}^2(p) \cdot \quad (2)$$

, where T is the number of sample and $\hat{\sigma}^2(p)$ is the long-run variance of ε_t and can be constructed from the residuals ε_t as:

$$\hat{\sigma}^2(p) = \frac{1}{T}\sum_{t=1}^{T}\varepsilon_t^2 + \frac{2}{T}\sum_{t=1}^{p} w_j(p) + \sum_{t=j+1}^{T}\varepsilon_t \varepsilon_{t-j}, \quad (3)$$

, where p is the truncation lag, $w_j(p)$ is an optional weighting function that corresponds to the choice of a special window [8]. Under the null hypothesis of level stationary,

$$KPSS \to \int_0^1 V_1(r)^2 dx \quad (4)$$

, where $V_1(x)$ is a standard Brownian bridge: $V_1(r) = B(r) - rB(1)$ and $B(r)$ is a Brownian motion (Wiener process) on $r \in [0, 1]$. Because (4) is refereed in testing only on the intercept and not in the trend and as we are testing with both intercept and trend we have the second-level Brownian bridge $V_2(x)$ and it is:

$$KPSS \to \int_0^1 V_2(r)^2 dx \quad (5)$$

, where $V_2(x)$ is given by:

$$V_2(r) = W(r) + (2r - 3r^2)W_1 + (-6r + 6r^2)\int_0^1 W_s(s)ds \quad (6)$$

B. Autoregressive (AR) Models

We consider a series $y_1, y_2, \ldots, y_n$. An autoregressive model of order p denoted AR(p), states that y_t is the linear function of the previous p values of the series plus an error term:

$$y_t = \phi_0 + \phi_1 y_{t-1} + \phi_2 y_{t-2} + ... + \phi_p y_{t-p} + \varepsilon_t \quad (7)$$

, where $\varphi_1, \varphi_2, \ldots, \varphi_p$ are weights that we have to define or determine, and ε_t denotes the residuals which are normally distributed with zero mean and variance σ^2 [9]. Various procedures have been suggested for determining the appropriate lag length in a dynamic model such as based on information criteria Akaike, Schwartz and Hannan-Quinn or based on the t-student statistics indicating that the last added lagged dependent variable is significant. Specifically we choose Akaike criterion which is defined as:

$$AIC(p) = \ln \frac{e'e}{T} + \frac{2p}{T} \quad (8)$$

, where e denotes the residuals, T is the sample and p indicates the lag number. We examine Akaike criterion up to 5 lags. Conditioned on the full set of information available up to time i and on forecasts of the exogenous variables, the one-period-ahead forecast of y_t would be

$$\hat{y}_{t+1|t} = \hat{\phi}_{0\,t+1|t} + \hat{\phi}_1 y_t + \hat{\phi}_2 y_{t-1} + ... + \hat{\phi}_p y_{t-p+1} + \hat{\varepsilon}_{t+1|t} \quad (9)$$

B. Feed-Forward Neural Networks Autoregressive (FFNN-AR) Models with Genetic Algorithms

The Feed-Forward Neural Networks model is a widely used approach known for its speed and accuracy. In Fig. 1 we present a feed-forward neural network with an input layer of m_0 nodes for $n=1 \ldots m_0$, one hidden layer and a single output layer. The input layer includes the input variables, which in the case we examine are the lags of the dependent variable of (1) and specifically the Gross Domestic Product of each country. The hidden layer consists of hidden neurons or units placed in parallel. Each neuron in the hidden layer performs a weighted summation of the weights which then passes an activation function. The output layer of the neural network is formed by another weighted summation of the outputs of the neurons in the hidden layer [10].

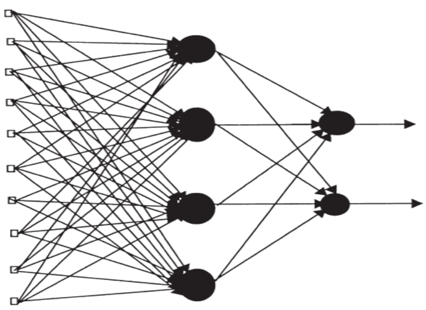

Fig. 1. A Feed-Forward neural networks with one hidden layer and one output layer

The cost function which is minimized is defined as:

$$e_k(n) = d_k(n) - y_k(n) \tag{10}$$

, where $e_k(n)$ is the error signal, $y_k(n)$ is the neural network output signal and $d_k(n)$ is the desired target, which is the real value of the Gross domestic Product growth rate. The purpose of the neural network learning process is to apply corrective adjustments to the synaptic weight of neuron k in order to make the output $y_k(n)$ to come closer to the desired response $d_k(n)$ in a step-by-step manner. The minimization of the cost function is:

$$f(n) = \frac{1}{2} e_k(n) \tag{11}$$

We denote the $w_{kj}(n)$ as the value of the synaptic weight w_{kj} of neuron k excited by element $x_j(n)$ on the signal input vector $x_j(n)$ at time step n, where input vector contains the independent variables we examine. We test three transfer functions, from input to hidden layer, the logistic, hyperbolic tangent and linear On the other hand the linear transfer function, from hidden to output layer, is used in all three tests. The logistic, hyperbolic tangent and linear transfer functions are defined respectively by expressions (12)-(14)

$$f(x) = \frac{1}{1+e^{-x}} \tag{12}$$

$$f(x) = \frac{2}{1+e^{-2x}} - 1 \tag{13}$$

$$f(x) = x \tag{14}$$

The tests show that all functions give satisfying forecasts. We randomly choose the logistic function and therefore we present only the results of the last function.

The Genetic Algorithm (GA) works with a fixed-size population of possible solutions for a problem, called individuals, which are evolving in time. A GA uses 3 principal genetic operators: selection, crossover, and mutation. During each step, called a generation, in the reproduction process, the individuals in the current generation are evaluated by a fitness function, which is a measure of how well the individual solution solves the problem. Then each individual solution is reproduced in proportion to its fitness; higher fitness means a greater chance to participate in mating, which is the cross-over stage, and to produce an offspring. A small number of newborn offspring i.e., new solutions, undergo the action of the mutation operator. After many generations, only those individuals –solutions with the best genetics or from the point of view of the fitness function survive. The best individual provides an optimum or near optimum solution to the problem. Genetic algorithms are essentially probabilistic. In contrast, traditional optimization methods are essentially deterministic. The probabilistic nature of evolutionary computations allows them to explore areas in the search space that may appear improbable at first glance. Bad solutions-individuals are not thrown out from the population. Instead, they have some finite probability of mating and of giving future generations some genetic features that could be very useful in creating true elite offspring. Thus, the GA avoids local optima and can find a true global solution to the problem [11]-[12]. We minimize (11) in order to find the optimum parameters. The steps for genetic algorithm are [11]-[12]:

1. Start with a randomly generated population of n chromosomes, which are the candidate solutions. . It should be noticed that we could obtain binary encoding incorporate the rules. We preferred to take real numbers encoding and the random population is generated based on the range of the actual values, because the results are superior than those we could have if we had taken binary encoding. Moreover in neural networks real number encoding is more appropriate because we are trying to find the optimum weights [11]-[12].
2. Calculate the fitness $f(x)$ of each chromosome x in the population
3. Repeat the following steps until n offspring have been created:
   a. select a pair of parents chromosomes of the current population and compute the probability of selection being an increasing function of fitness. In this case we take the roulette wheel selection algorithm. Also the selection process is one with replacement meaning that the same chromosome can be selected more than once to become a parent.
   b. The next step is the crossover. We use one-point crossover process with probability p_c cross over the pair at a chosen point. If no crossover takes place we form two offspring that are exact copies of their respective parents.
   c. Mutate the two offspring with probability p_m and place the resulting chromosomes in the new population. We use uniform mutation.
4. Replace the current population with the new population.
5. Go to step 2.

Finally we propose the following neural network regression:

$$y = \sum_{i=1}^{p} (\phi_1 y_{t-i} + \mathrm{b}) w^{(A)}{}_{kj} \tag{15}$$

, where for $i=1,2...p$ is the number of lags as in the case of AR models, $w^{(A)}{}_{kj}$ are the optimized weights from hidden to input layer and b is the bias which is a vector of ones as in the case of the ordinary least squares method. So we regress the initial dependent variable y, which denotes the GDP growth rates, on the weighted inputs, forming a kind a weighted regression. The forecasting performance of Autoregressive (AR) models and Feed-forward neural networks Autoregressive (FFNN-AR) models in both in-sample and out-of- sample periods is counted based on the Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) described respectively by (16) and (17).

$$MAE = \frac{1}{n}\sum_{i=1}^{n} | y_i - \hat{y}_i | \tag{16}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (y_i - \hat{y}_i)^2} \tag{17}$$

III. DATA

The data are in quarterly frequency and are referred in Gross Domestic Product (GDP) growth rates for quarter-by-quarter. The period examined is 1991-2009 for Canada, France, Italy, Japan, UK and USA. Moreover the period 1991-2006 is obtained as the in-sample for AR model or as the train period for the FFNN-AR model, while period 2007-2009 is taken as the out-of-sample period. Moreover, we apply a four-step ahead period forecasting. Specifically in the one-step ahead prediction both models present a very similar and high performance. The purpose is to extend the step forecasting period because it is much more useful. Firstly, we estimate the forecasts for 2007, then we replace the forecasting values with the actual the forecasts for 2008. The same procedure is followed for year 2009. It should be noticed that more than four step periods ahead the forecasting performance of the models becomes poor.

IV. EMPIRICAL RESULTS

In Table I we present the results of ADF and KPSS tests. The results are ambiguous among the two tests as for example we accept the hypothesis that the Canadian GDP is stationary in levels, according to ADF test, but it is stationary in first differences based on KPSS test. Generally according to ADF test and for α=0.01 all time series we examine are *I(1)*, with the exception that of Canadian GDP, while the GDP in Japan and UK are are *I(0)* in α=0.05 and the GDP of Italy and USA are *I(0)* in α=0.10. Based on KPSS and again for α=0.01 only GDP of Italy and Japan are *I(0)*.

In Tables II and III the Autoregressive (AR) and Feed-Forward Neural Networks Autoregressive (FFNN-AR) regression estimations are reported The maximum number of genetic iterations is set up at 50, the crossover rate p_c at 0.2 and the mutation rate p_m at 0.01.

TABLE I
ADF UNIT ROOT AND KPSS STATIONARY TESTS

Countries	ADF test		KPSS test	
	In Levels	In First differences	In Levels	In First differences
Canada	-4.29		0.1542	0.0928
France	-2.039	-6.25	0.1270	0.0342
Italy	-3.26	-6.40	0.0697	
Japan	-3.56	-4.07	0.0988	
UK	-3.81	-6.70	0.1797	0.0739
USA	-3.17	-6.45	0.1499	0.0685
Critical values for ADF[1]	-4.110 for α=0.01	Critical values for KPSS[2]	0.216 for α=0.01	
	-3.482 for α=0.05		0.146 for α=0.05	
	-3.169 for α=0.10		0.119 for α=0.10	

1 MacKinnon [13], 2 Kwiatkowski *et al.*,[7]

TABLE II
. AUTOREGRESSIVE (AR) MODEL ESTIMATIONS

	Estimated parameters				
	φ_1	φ_2	φ_3	φ_4	φ_5
Canada	0.9114	0.1542	-0.1429	-0.4483	0.5124
	(0.1072)	(0.1471)	(0.1470)	(0.1461)	(0.1064)
	[8.501]*	[1.047]	[-0.971]	[-3.068]*	[4.816]*
France	1.3021	-0.1914	-0.1955		
	(0.1231)	(0.2002)	(0.1243)		
	[10.577]*	[-0.956]	[-1.572]		
Italy	1.3204	-0.3020	-0.1743		
	(0.1231)	(0.2002)	(0.1243)		
	[11.197]*	[-1.550]	[-1.379]		
Japan	1.0447	-0.1479	-0.1001		
	(0.1220)	(0.1758)	(0.1219)		
	[8.565]*	[-0.841]	[-0.821]		
UK	1.631	-0.600	-0.1376	-0.2674	0.3503
	(0.1216)	(0.2293)	(0.2416)	(0.2514)	(0.1503)
	[13.423]*	[-2.620]**	[-0.571]	[-1.063]	[2.330]**
USA	1.316	-0.2154	-0.1892	-0.3396	0.4044
	(0.1186)	(0.1958)	(0.1980)	(0.1991)	(0.1233)
	[11.099]*	[-1.099]	[-0.955]	[-1.70]***	[3.280]*
	Diagnostic tests				
	F-statistic	R^2_{adj}	Q-stat (2)	Standard Error of Estimate	
Canada	16.601	0.4713	4.719	0.8688	
	{0.000}		{0.4510}		
France	202.969	0.8487	5.795	0.6506	
	{0.000}		{0.3744}		
Italy	225.929	0.8620	5.083	0.7726	
	{0.000}		{0.4230}		
Japan	88.907	0.7095	7.151	1.2168	
	{0.000}		{0.1281}		
UK	234.785	0.9304	2.677	0.5540	
	{0.000}		{0.6132}		
USA	113.928	0.8658	0.535	0.6852	
	{0.000}		{0.9700}		

Standard errors in parentheses, t-statistics in brackets, p-values in {}, * denotes significance in α=0.01, ** denotes significance in α=0.05, *** denotes significance in α=0.10, Q-stat is the Ljung-Box test on squared standardized residuals with 2 lags

TABLE III
. FEED-FORWARD NEURAL NETWORKS AUTOREGRESSIVE (FFNN-AR) MODEL ESTIMATIONS

	Estimated parameters				
	φ_1	φ_2	φ_3	φ_4	φ_5
Canada	1.5409	0.3760	-2.0242	1.1981	-0.5953
	(0.3196)	(0.2424)	(0.3879)	(0.1709)	(0.1679)
	[4.821]*	[1.551]	[-5.218]*	[7.011]	[-3.54]*
France	0.0068	0.1468	-0.4315		
	(0.0381)	(0.0132)	(0.0354)		
	[0.179]	[11.100]*	[-12.17]*		
Italy	0.1916	-0.2206	-0.2167		
	(0.0324)	(0.0266)	(0.0175)		
	[5.908]*	[-8.295]*	[-12.35]*		
Japan	0.2127	0.0027	-0.0621		
	(0.0412)	(0.0103)	(0.0661)		
	[5.164]*	[0.265]	[-0.940]		
UK	-0.9284	-1.8622	0.1860	0.2107	0.2391
	(0.1832)	(0.3616)	(0.5344)	(0.0477)	(0.0273)
	[-5.067]*	[-5.149]*	[3.348]*	[4.412]*	[8.739]*
USA	-0.2843	0.0258	-0.3103	0.2753	0.3012
	(0.1274)	(0.0738)	(0.0363)	(0.1509)	(0.0668)
	[-2.23]**	[0.3496]	[-8.549]*	[1.82]***	[4.51]*
	Diagnostic tests				
	F-statistic	R^2_{adj}	Q-stat (2)	Standard Error of Estimate	
Canada	16.983	0.4810	1.7209	0.8644	
	{0.000}		{0.9321}		
France	218.906	0.8599	4.802	0.6237	
	{0.000}		{0.4889}		
Italy	212.073	0.8560	5.610	0.7666	
	{0.000}		{0.3205}		
Japan	92.243	0.7199	7.435	1.1956	
	{0.000}		{0.1146}		
UK	214.612	0.9253	2.677	0.5495	
	{0.000}		{0.6132}		
USA	119.111	0.8726	0.384	0.6625	
	{0.000}		{0.9837}		

Standard errors in parentheses, t-statistics in brackets, p-values in {}, * denotes significance in α=0.01, ** denotes significance in α=0.05, *** denotes significance in α=0.10, Q-stat is the Ljung-Box test on squared standardized residuals with 2 lags

We observe in Tables II and III that the hypothesis of autocorrelation in residuals existence is rejected.

In Tables IV and V we present the MAE and RMSE measures for the in-sample and the out-of-sample period respectively of the estimated models. Only in one case AR model outperforms the FFNN-AR model in the in-sample period and more specifically in the case of UK, while in the remained cases we examine, RMSE and MAE are very close among the two models. On the other hand in the out-of-sample period which is of greatest interest, FFNN-AR with genetic optimization outperforms significant the AR model. This can be shown in Fig. 2-7, where the forecasts generated of the two models examined versus the actual values of GDP for the out-of-sample period are reported. This indicates that FFNN-AR with genetic algorithms optimization is a good alterative choice in time series modelling.

TABLE IV
FORECASTING PERFORMANCE OF AR AND FFNN-AR MODELS FOR IN-SAMPLE PERIOD

	AR		**FFNN-AR**	
	MAE	RMSE	MAE	RMSE
Canada	0.6715	0.8376	0.6653	0.8303
France	0.5201	0.6371	0.4729	0.5604
Italy	0.5808	0.7566	0.5348	0.6346
Japan	0.9243	1.4199	0.9014	1.1704
UK	0.4196	0.5341	0.4251	0.5388
USA	0.5481	0.6606	0.5381	0.6538

TABLE V
FORECASTING PERFORMANCE OF AR AND FFNN-AR MODELS FOR OUT-OF-SAMPLE PERIOD

	AR		**FFNN-AR**	
	MAE	RMSE	MAE	RMSE
Canada	1.8316	2.1814	1.1970	1.4836
France	2.1385	2.5840	1.4401	1.7210
Italy	2.3555	2.9334	1.8876	2.1612
Japan	2.5912	3.1301	2.2356	2.7698
UK	2.2230	2.8528	1.7563	2.2013
USA	2.0581	2.5179	1.7242	2.1711

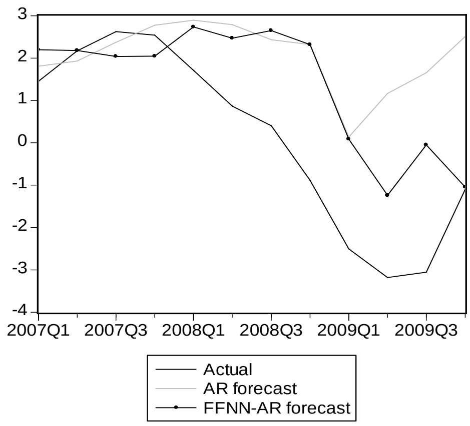

Fig. 2 Out-of-sample period forecasts with AR and FFNN-AR models for Canada

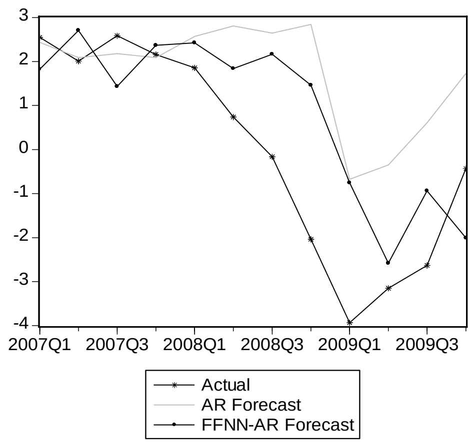

Fig. 3 Out-of-sample period forecasts with AR and FFNN-AR models for France

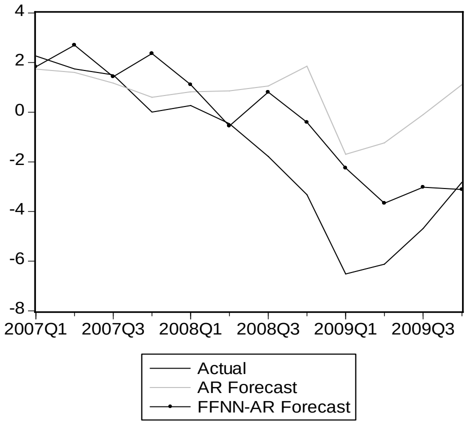

Fig. 4 Out-of-sample period forecasts with AR and FFNN-AR models for Italy

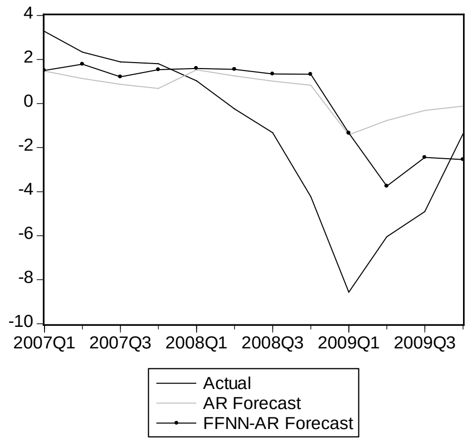

Fig. 5 Out-of-sample period forecasts with AR and FFNN-AR models for Japan

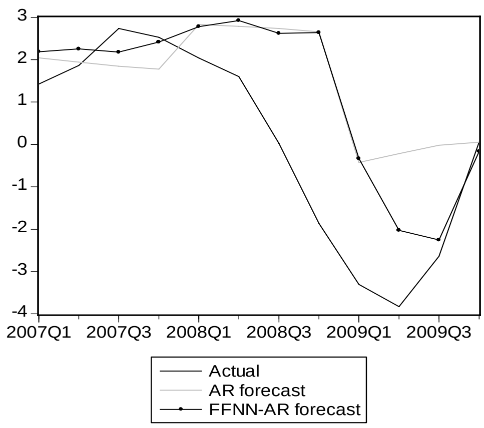

Fig. 6 Out-of-sample period forecasts with AR and FFNN-AR models for UK

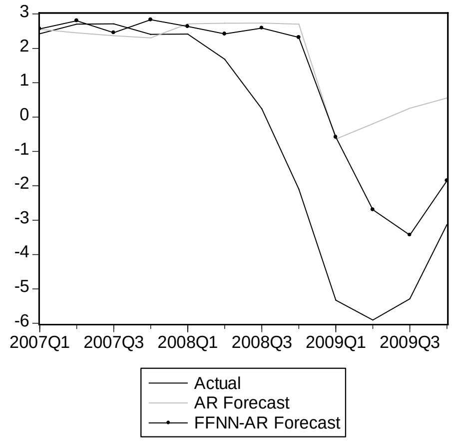

Fig. 7 Out-of-sample period forecasts with AR and FFNN-AR models for USA

V. CONCLUSIONS

The main conclusion of this paper is that the forecasting performance of the Feed-Forward Neural Networks Autoregressive (FFNN-AR) model is a little superior or very close in the in-sample period comparing with the simple Autoregressive (AR) model. On the other hand we have shown that FFNN-AR with genetic algorithms outperforms significant the Autoregressive model. The purpose of the paper was to compare only Feed-Forward Neural Networks Autoregressive (FFNN-AR) and Autoregressive (AR) model, while Feed-Forward Neural Networks can be extended as well as into Feed-Forward Neural Network Moving Average (FFNN-MA), or Autoregressive Moving Average (FFNN - ARMA) or even Smoothing Transition Autoregressive (FFNN-STAR) model, opening a new research field in econometric literature.

REFERENCES

[1] R.D. Aryal and W. Yao-Wu, "Neural Network Forecasting of the Production Level of Chinese Construction Industry", *Journal of Comparative International Management* , vol. 29, pp. 319-33, 2003
[2] W. Li, Y. Luo, Q. Zhu, J. Liu and J. Le, "Applications of AR*-GRNN model for financial time series forecasting", *Neural Computing and Applications*, vol. 17, no. 5/6, pp. 441-448, 2008
[3] N.R Swanson and H. White, "A model selection approach to real time macroeconomic forecasting using linear models and artificial neural networks", *Review of Economics and Statistics*, vol. 79, pp. 540-650, 1997a
[4] N.R Swanson and H. White, "Forecasting economic time series using adaptive versus non-adaptive and linear versus nonlinear econometric models", *International Journal of Forecasting*, vol. 13, pp. 439-461, 1997b
[5] G. Tkacz, "Neural network forecasting of Canadian GDP growth", *International Journal of Forecasting*, vol. 17, pp. 57-69, 2001
[6] D. A. Dickey, and W. A. Fuller, "Distribution of the Estimators for Autoregressive Time Series with a Unit Root", *Journal of the American Statistical Association*. vol. 74, pp. 427–431, 1979
[7] D. Kwiatkowski, P. C. B. Phillips, P. Schmidt and Y. Shin, "Testing the Null Hypothesis of Stationarity against the Alternative of a Unit Root", *Journal of Econometrics,* vol. 54, pp. 159-178, 1992

[8] M. S. Bartlett, "Periodogram analysis and continuous spectra," *Biometrika,* vol. 37, pp. 1-16, 1950
[9] W.H Greene, *Econometric Analysis*, Sixth Edition, Prentice Hall, New Jersey, 2008, pp. 560-579
[10] Haykin, S. (1999), *NEURAL NETWORKS: A Comprehensive Foundation*, Second Edition Pearson education, Prentice Hall, Delhi, India, 1999, pp. 33-47, 73-76
[11] Z. Michalewicz, *Genetic Algorithms + Data Structures= Evolution Programs*, Springer, 1995, pp. 33-36
[12] M. Mitchell, *An Introduction to Genetic Algorithms*, MIT Press Cambridge, Massachusetts, London, England, 1996, pp. 12-13, 58-59, 118
[13] J. G. MacKinnon, "Numerical Distribution Functions for Unit Root and Cointegration Tests", *Journal of Applied Econometrics*, vol. 11, pp. 601-618, 1996