Iranian Economic Review, Vol.9, No.11, Fall 2004

Forecasting GDP Growth Using ANN Model With Genetic Algorithm

Saeed Moshiri*

Mohammad Reza Ghadimi**

Abstract

Applying nonlinear models to estimation and forecasting economic models are now becoming more common, thanks to advances in computing technology. Artificial Neural Networks (ANN) models, which are nonlinear local optimizer models, have proven successful in forecasting economic variables. Most ANN models applied in Economics use the gradient descent method as their learning algorithm. However, the performance of the ANN models can still be improved by using more flexible and general learning algorithm. In this paper, we develop an ANN model combined with Genetic Algorithm to forecast the Iranian GDP growth. In order to evaluate the performance of the model with other ANN and traditional econometric models, we compare the results of the model with other linear and nonlinear competing models such as ARMA, VAR, and ANN with gradient descent learning algorithm. We use the recently produced extended data on the Iranian GDP from 1937 to 2002. The results indicate that the GA can improve the forecasting performance of ANN model over other standard ANN and econometrics models.

Keywords: Forecasting, GDP Growth, Artificial Neural Networks, Genetic Algorithm, ARMA, VAR

1- Introduction

Applying nonlinear models to estimation and forecasting economic models are now becoming more common, thanks to advances in computing technology. Many studies using nonlinear models have shown that they have been able to

*- Assistant Professor, Faculty of Economics, University of Allameh Tabatabaie, Tehran, Iran, 15134, Email: smoshiri@mun.ca.

**- Ph.D. Candidate, Faculty of Economics, University of Allameh Tabatabaie, Tehran, Iran.

outperform linear models, particularly where the underlying data generation process is known to be nonlinear. For instance, Rothman (1998) uses six nonlinear models to forecast asymmetric business cycles using unemployment rates data and concludes the better results over the linear model. Stock and Watson (1998) compare four classes of linear and nonlinear models applied to forecasting 215 monthly US macroeconomic time series and conclude that the feedforward ANN model is able to forecast better than linear (AR, and EX) models and nonlinear (LSTAR) models in one month forecast horizon, but not in six and twelve month horizons. Zellner (2002) describes his nonlinear model-building experiments when working on projects such as marine conservation programs, dam construction on the Susquehanna River, and the Federal Reserve-MIT-PENN model of the US economy.

One of the problems with nonlinear models is the model specification. When we leave the linear world, we enter a large set of model specification options; there are many nonlinear functional forms to choose from. However, nonlinear ANN models are flexible and able to learn any patterns in the data and generate their future values with reasonable precision. In ANN models, one has to make two choices when specifying the model: the number of input and hidden neurons, and the number of hidden layers. The underlying economic theory or statistical features of the data can determine the number of input neurons, and certain formula or trial and error can specify the number of hidden layer neurons. As for the number of hidden layers, although the complexity of the models can increase by introducing extra layers, in most economic applications only one layer has proven sufficient, reducing the number of choices the modeler should make.

Most economic applications of ANN models have dealt the classification and forecasting variables in the financial markets. However, there have been a few works on macroeconomic applications. For instance, Masoomi et al (1994), Swanson & White (1997), Fu (1998), Tkacz (1999), Moshiri & Cameron (2000), and Moshiri and Brown (forthcoming) used ANN models to forecast variables such as GDP, unemployment rates, inflation, and interest rates. In many cases, they found ANN models outperformed the competing traditional linear and nonlinear models, particularly in short horizon forecasting.

Many economic applications of the ANN models use the feed forward ANN model with back propagation learning algorithm. However, the performance of back propagation ANN may be improved, when a better learning algorithm or a dynamic structure is applied. In this paper, we tend to forecast economic growth using the ANN model equipped with the Genetic Algorithm (GA). GA is an optimization algorithm adapted from genetic science and can be used for learning algorithm in the ANN model instead of the traditional gradient descent.

The organization of the paper is as follows. In section 2, a brief description of the ANN and GA will be presented. In section 3, the data and in section 4 the model will be explained followed by the results. Section 4 will conclude the paper.

2- The ANN And GA

The ANN models are adaptive nonlinear models based on a natural neural system. A network of neurons is designed in three input, hidden, and output layers with adjustable weights connecting them. The network reads in the inputs, and then produces the output through a learning algorithm. The process is iterated and weights are adjusted until the produced output is close enough to the actual output, or the error is minimized. Rumelhart et al(1986) showed that an ANN model with sufficient hidden layer neurons is able to learn any kind of patterns in the data. There are two kinds of learning algorithm: gradient descent method and global search method. The back-propagation, Newton, Quasi-Newton, Levenberg-Marquant methods can be classified as gradient descent method. The GA is considered as a global search method.

The ANN model is very useful, particularly when the underlying data generation process is nonlinear and complex, or functional form is not clear. However, the learning process, or parameter adjustment, in an ANN model can take time and may end up in a trap, i.e., a local minimum, particularly when there are multiple parameters. To avoid such potential problems, the GA can be used in the learning process instead of gradient descent method. The GA, which is borrowed from the biology, mimics the way evolution finds a solution to a given problem. The GA searches for all possible solutions, the parameters that minimize the forecast errors, and finds the best one based on the criteria given to the model.

A typical GA uses three operators: crossover, mutation, and selection (Holland, 1975). The GA starts with a large population of candidate solution vectors that are subjected to some selection criteria, such as survival of the fittest, and through the operations above searches for the improved solutions. In another word, the GA optimizes the tradeoff between sticking with the solutions performed well in the past and searching for new solutions. The GA works as follows.

The objective function is to minimize the sum of squared errors: differences between the actual and the predicted dependent variable. Therefore, the problem is finding the parameters (weights), which are adjusted by the genetic operations, so that the error function is minimized. The process is carried out in eight steps. First, a set of initial population of weight vectors with the size of p is randomly created. Second, two pairs from the initial population are randomly selected, and the best of each pair is taken as a parent. They are winners by fitness criteria, such as estimated accuracy of prediction, or probability of survival. Third, the crossover operation is applied to the parents to create children in the following way. Using a cut-point, also chosen randomly, the elements of the two weight vectors are replaced with each other, creating two new weight vectors, called children. Fourth step is the mutation operation through which a random shock with a small probability is applied to each element of the two children. Fifth, the four weight vectors, two parents and two children, engage in a tournament through which a vector that contributes the most to achieving the objective will survive and the rest will extinguish. Sixth, the process will iterate so that the new generation of weight vectors with the size of the initial population, p, are created. In step seven, using fitness criteria, all the members of the new generation with the members of the old generation are compared and the best is selected to pass through the next round. And finally, the process continues for many generations until the system converges where there is no meaningful difference between the old and the new generations. In fact, under the GA, elitism is applied, implying that the best weight vectors are generated after mixing the population and selection of the elements based on the best-fit criteria. The selection criterion in tournaments is how they minimize the squared error function in the ANN. More detailed explanation of the GA is presented in the appendix.

Among the three major operations in the GA, the crossover operation is the most crucial one for obtaining global outcomes. It mixes the partial information contained in the strings of the population. The mutation operation enhances the search space by introducing new strings in the generation. Since applying mutation operation too frequently will cause a loss of highly fit strings in the population, an optimal mutation probability is set to 1/k, where k is the number of coefficient weights (parameters) in the ANN.

3- The Data

We have used an extensive annual data set recently made available by the Iranian Central Bank (KhavariNejad, 2002). This data set expands the Iranian macroeconomic series beyond 1959 to 1936, creating time series for the period 1939-2002. The use of this new data set alleviates the problem of the small data set usually used in the macroeconomic applications of the ANN models, particularly where there is no seasonal macroeconomic series available.

4- The Forecasting Models

We use three sets of econometric models; namely structural, univariate time series, and multivariate time series; along with two sets of ANN and GANN (Genetic Algrothim ANN) to estimate and forecast the Iranian economic growth. The specification of the models is as follows.

1- Growth Equation Model

Following the endogenous growth model literature, we try variables from two groups of neoclassical and institutional factors to explain the economic growth. They are investment ratio, labor/population, human capital, openness, financial variables, inflation, government, and political conditions. All the data are tested for the presence of the unit root to avoid the spurious regression problem. The ADF test results show that the series are all generated by stationary processes, and therefore, they can be used in their level forms. We start with a very general specification and then reduce the size of the model by dropping off the insignificant variables. The following regression produced the best result.

$$\underset{t}{RGDP} = \underset{(2.54)}{0.03} + \underset{(1.94)}{0.24}\, DIGDP + \underset{(1.80)}{0.09}\, RH + \underset{(4.23)}{0.16}\, RXM - \underset{(-1.61)}{0.06}\, DP - \underset{(-2.02)}{0.04}\, D + e$$

Adjusted R2 = 0.60 , DW= 2.59 , F=18.41

Where RGDP is the per capita real GDP growth, DIGDP investment ratio, RH growth in human capital approximated by the labor with high education, RXM openness approximated by the sum of exports and imports ratios, DP inflation, and D dummy variable which is 1 for years corresponding with world war II, revolution, and war with Iraq, i.e., 1320, 1357, and 1380, and zero for the rest. All coefficients are significant with right signs and the regression equation has been able to explain about 60 percent of the growth variation over the 63 years of observations.

2- Univariate Time Series Growth Model

Using the Box-Jenkins methodology, the ARMA (1,1) produced the best result to estimate the Iranian economic growth. The results are as follows.

$$RGDP_t = 0.98\ RGDP_{t-1} - 0.88\ e_{t-1} + u_t$$
$$t \qquad (34.46) \qquad (-11.74)$$

Adjusted R2 = 0.60, DW= 2.59, F=18.41, LM = 3.07 Jarque-Bra=5.37

The model above produces the least AIC and SBC, and the diagnostic test results as shown by the LM and J-B tests indicate that the residuals are independent and normally distributed.

3- Multivariate Time Series Growth Model

We set up a VAR model to incorporate the dynamics in the relationship among the variables affecting growth. The variables are the same as those used in the structural regression equation above and the number of lags is determined by the AIC and SBC criteria. Since the AIC and SBC results are indicative of a VAR with different orders, we use out-of-sample forecasting errors criteria to decide on the number of lags. VAR(4) produces lower RMSE and MAE of forecasts, and therefore, is chosen as the best multivariate time series for estimating and forecasting the growth model.

4- ANN Model

A set of ANN models with back-propagation and gradient descent learning algorithm are applied to estimate and forecast the Iranian economic growth.

Although the ANN models can learn any patterns in the data, as Swanson & White (1997), Chatfield (1993), and Moshiri & Cameron (2000) show, the best results are produced when the input variables are selected based on the economic theory or the statistical features of the data. In this paper, we set up a variety of ANN models based on different input variables using the three econometric models presented above. Specifically, we use three ANN models as follows: a) Structural ANN, which uses the variables specified in the structural growth model. b) ARMANN, which uses ARMA specification. And c) VARANN, which uses variables specified in VAR as explained above.

In the ARMANN specification, we need moving average terms as part of input vector. One way to introduce this type of dynamics into ANN model is to allow a feedback from the output by introducing estimated error terms as inputs. We use the Jordan ANN model, which is a type of recurrent ANN, to capture the dynamics of the data (Kuan & White, 1994, Moshiri et al, 1999). VARANN's structure is based on a system of equations and it is, therefore, slightly different from the ARMANN. In the VARANN specification, the number of input layer nods is 22, consisting of five endogenous variables with four lags each plus intercept and a dummy variable, and the number of output layer nods is five.

Following the convention in the economic applications, only one hidden layer is used in the network. The number of hidden layer nodes is usually selected by trail and error, but in this paper, we use GA to search for the best number of hidden nods. In fact, the GA searches for the optimal number of hidden nodes, momentum rate, and the learning rate parameters, which minimize the forecasting error. The numbers of hidden layer nods selected by the GA for the three ANN models above are one, three, and four, respectively. The gradient descent learning algorithm is used in all versions of the ANN models applied in this section.

5- GANN Model

The GANN is an ANN model specified above which uses GA, instead of gradient descent, as its learning algorithm. The GA as explained earlier selects the best connection weights matrix in the ANN model by applying the best-fit criteria using the three crossover, mutation, and selection operations. The parameters of the GA are set as follows: the population size 50, cut-point probability 0.9, and probability for the mutation process 0.01. The optimization operation is repeated 100 times to achieve the best results.

5- The Results

To evaluate the forecasting results, the data set is divided into two parts: one for estimation, or learning in the ANN terminology, and another for forecasting, or generalization in ANN. The estimation and the forecasting periods are 1936-1986, and 1987-2002, respectively. The forecasting method in the structural set up is static and in the time series is static and dynamic, i.e., the forecast values, not the actual values, are used when generating new forecasts. We evaluate one-period-ahead out-of-sample forecasting results produced by the models above by five criteria. They are Root Mean Square Errors (RMSE), Mean Absolute Errors (MAE), Mean Absolute Percentage Errors (MAPE), Theil Inequality Criteria (TIC), and Confusion Rate (CR). The first three measure some functions of forecast errors, the TIC compares the RMSE of the model with that obtained by a naïve model, and the CR, which is the ratio of incorrect forecasts in direction to total number of forecasts, measures the precision of the turning points forecasts.

There are three groups of the models, namely, structural, univariate time series, and multivariate time series. Within each group, there are ANN and GANN models according to their group's specifications. In total, therefore, there are nine models for estimation and forecasting. A summary of the static forecasting results generated by the models above is presented in Table 1 and the dynamic forecasting results in Table 2. As the results show, the GANN consistently outperforms the other models in the structural specification. It produces almost similar outcomes with ANN model, but outperforms the ARMANN: the ANN model with ARMA specification. Finally, the forecasting results are much better for the GANN Model than those produced by other two models when VAR specification is applied. Overall, we can decide that the GANN model is able to outperform other competing models from the Econometric and even the ANN model with BPN and gradient descent learning algorithm.

Although the forecasting results shown in Table 1 indicate that the GA and ANN models clearly outperform other competing models, the differences in the forecasting errors should be tested for their statistical significance. In this paper, we apply the test developed by Diebold-Mariano (1995) and its revised version by Harvey, Leybourne, and Newbold (1997), to check for statistically meaningful discrepancies among the forecast errors generated by the GA, ANN and other econometric models. The Diebold-Mariano test statistic is as follows.

$$s = \bar{d} \Big/ \sqrt{\phi}$$

Table (1): The Static Forecasting Results By Structural, Time Series, ANN and GANN Models

Model	RMSE	MAE	MAPE	TIC	CR
Structural	0.035	0.031	281.75	0.6	0.714
ANN_Structural	0.023	0.019	176.68	0.49	**0.571**
GANN_Structrual	**0.022**	**0.018**	**171.474**	**0.46**	**0.571**
ARMA	0.015	0.013	32.22	0.199	**0**
ANN_ARMA	0.04	**0.011**	37.13	**0.157**	0.142
GANN_ARMA	0.015	0.013	**31.48**	0.181	0.142
VAR	0.050	0.041	467.61	0.58	0.571
ANN_VAR	0.028	0.026	369.01	0.46	0.571
GANN_VAR	**0.016**	**0.013**	**167.2**	0.40	**0.285**

Structural, ARMA, and VAR are structural growth equations, univariate, and multivariate time series models presented in section 4, respectively. ANN_ and GANN_ are the ANN and GANN models with the structural, ARMA, and VAR specifications, respectively.

Table (2): The Dynamic Forecasting Results By The Time Series, ANN and GANN Models

Model	RMSE	MAE	MAPE	TIC	CR
ARMA	0.016	0.014	31.77	0.213	**0**
ANN_ARMA	**0.014**	**0.011**	37.13	**0.157**	0.142
GANN_ARMA	0.015	0.012	**35.56**	0.179	0.142
VAR	0.075	0.069	878.99	0.68	0.57
ANN_VAR	0.030	0.028	391.80	**0.48**	0.571
GANN_VAR	**0.028**	**0.023**	**264.63**	0.50	**0.428**

ARMA, and VAR are structural growth equations, univariate, and multivariate time series models presented in section 4, respectively. ANN_ and GANN_ are the ANN and GANN models with the structural, ARMA, and VAR specifications, respectively.

Where $\bar{d}$ is the mean of the difference between squared errors of two models, and ϕ is asymptotic variance of d which can be estimated by autocovariances of d as follows.

$$\hat{\phi} = \frac{1}{t_f}\left(\gamma_0 + 2\sum_{i=1}^{k-1}\gamma_i\right)$$

Where t_f is the forecasting period, γ_i the ith autocovariance estimation of d, and k is forecasting horizon. Diebold-Mariano shows that, under the null hypothesis of equal forecast errors, s has standard normal distribution with mean zero and variance one in large samples. For small samples, Harvey et al introduce the following revision of the test statistic.

$$s^* = s\left(\frac{t_f + 1 - 2k + \frac{k}{t_f}(k-1)}{t_f}\right)^{1/2}$$

s^* has t distribution with degrees of freedom t_f-1. We test the forecast errors generated by each econometric models, namely regression equation, ARMA, and VAR models against the ANN and GANN models, respectively. The results are summarized in Table 3.

Table (3): Harvey et al (1997) Test Results for Forecast Error Comparisons

Models	**s***
Structural & ANN	2.35
Structural& GANN	1.28
ARMA & ANN	0.33
ARMA & GANN	0.57
VAR & ANN	3.35
VAR & GANN	2.16

The critical values for t-student distribution with degrees of freedom 6 are 1.48, 2.01, and 3.14 for 10%, 5%, and 1% confidence level, respectively. Therefore, ANN's forecasting error differences with the structural and the VAR models are statistically significant at 5 percent, but not with ARMA. Likewise, the GANN has statistically different forecast errors when compared with the structural model at about 12 percent, and the VAR model at 5 percent, but not with ARMA. Since the forecasting error functions are very close to each other in the case of ARMA, ANN, and GANN models, the test results above should not be surprising.

To put the forecasting models in real test, we forecast the Iranian economic growth for the first two years of the third economic plan, i.e., 2001-2, using our model, and compare the results with those published by the Management and Planning Organization (MPO) macro model. The results are presented in Table 4. As the figures show, the economic growth has slowed down for the two forecasting years. The 3rd plan along with the ARMA, and VAR models have missed this drop in the economic growth, but the structural, and two ANN models have produced forecasts consistent with the actual numbers, although the former has overestimated and the latter underestimated the growth rates. According to RMSE and MAE criteria, ANN with different specifications have

Table (4): Forecast Error Comparisons among Competing Models for the First Two Years of the 3rd Iranian Economic Plan

	2001	2002	RMSE	MAE
Actual Growth	5.1	4.5		
3rd plan	4.5	5.5	0.0082	0.0079
Structural	6.4	5	0.011	0.009
Structural ANN	5	5.3	0.0057	0.0044
ARMA	3.4	3.5	0.014	0.013
ANN_ARMA	4.4	4.2	0.0053	0.0048
VAR	5.6	8	0.0246	0.0196
ANN_VAR	6.5	5.1	**0.0104**	**0.0096**

been able to produce lower forecast errors on average, with the ANN with VAR specification the lowest. These results can be taken as another evidence for better performance of the ANN models when compared with the traditional forecasting models.

6- Conclusion

Forecasting economic series is a big challenge as there are many factors affecting the variable and underlying relationships among the variables are complex. Advances in computing technology in recent years have allowed scientists, including economists, to make more use of non-linear and complex models to explore the non-explained parts of economic relationships. One problem with applying nonlinear models to economic modeling and forecasting is model specification; when we distance ourselves from the linear world, we enter a complex world with many functional forms to choose from. Finding and working with flexible functional forms, which can be applied universally, is therefore very crucial. The ANN models are among nonlinear models that are universal in learning any kind of patterns in the data and optimize the objective function locally, not globally as the traditional parametric linear regression equations do. The ANN models have shown that they have been able to perform well in forecasting many different economic variables such as stock prices, exchange rates, and inflation. There are some drawbacks in ANN models, such as being trapped in the local minimum rather than the global one. To enhance the performance of the ANN model, a variety of ANN models, such as recurrent ANN and radial basis function ANN, have been developed and applied in economic literature in the past decade. The Genetic Algorithm can also be used to empower the learning ability of ANN models. The GA, which is rooted in biology, searches for the best-fit solutions using crossover, mutation, and selection operations, and is particularly useful when the search space of possible solutions to a given problem is high-dimensional, where the systematic search algorithm is not feasible due to sheer number of combinations to test (McNelis, 1997).

There have been many applications of ANN models in the economic literature, but just a few of GA. In this paper, we investigate the forecasting performance of economic growth among the structural, univariate and multivariate time series models along with the ANN and an ANN combined with GA, using the data on the Iranian economic growth rates from 1936 to 2002. The out-of-sample one-period-ahead static and dynamic forecasting results indicate that the ANN, particularly when combined with GA, is able to outperform other econometric models.

One of the problems with the macroeconomic applications of adaptive models such as ANN and GA is the shortage of data, particularly in developing countries where data with high frequency is not regularly produced. In Iran, the macroeconomic data set is annual and starts at 1956. This gives us at most 44 observations, which are not adequate with ANN standards. In this paper, we use, for the first time, an extended macroeconomic data set which has expanded the data span back to 1936, generating 68 observations. The other option is to estimate seasonal data using the annual data set. Although this approach would generate more data, its use for estimation and forecasting is not appealing; we impose a certain relationship to the data, then try to model, and forecast it. In general, the more data one works with, the more confidence s/he will have with the results, other things being equal. Although our annual data set still is not very large, it is large enough for applying the nonlinear models such as ANN to a macroeconomic series and making comparisons among the alternative models.

Appendix

The basic idea in the GA is to find the fit individual in the population. In our context, the GA searches for the coefficient vector that is the global optimum solution. The basic GA process is as follows. You start with an initial randomly chosen population. Then determine the fitness of each individual in the population. If the population converged, you stop the process and present the solution. Otherwise, reproduce new individuals from the old ones and after certain operations create the next generations. You keep repeating the process until you reach the objective. The details of the GA process are discussed below.

The objective in the GA is to minimize the sum of squared errors subject to the parameters generated by the algorithm. That is,

$$\text{Min.}\sum_{i=0}^{n}(y_i - \hat{y}_i)^2 \qquad \text{subject to} \quad \hat{y} = f(x \mid \theta)$$

Where $\hat{y}$ is the model-generated output, and θ is the coefficient parameters which minimize the error function. The GA has eight major steps as follows.

1. Create an initial population P of coefficient vectors at generation 1, $[\theta 1, \theta 2, \ldots, \theta p]$, where P is an even number, and θi is a K by 1

vector. Thus, the size of the initial population is P by K. The initial population can be created by random draws from the standard normal distribution or imposing certain restrictions on the sign or the range of the parameters.

2. Choose two pairs of the initial population at random with replacement. Evaluate the fitness of these four vectors according to the objective function, and select two winners (vectors) with the best fitness (lowest sum of squared errors). Call the two best-fit vectors "parents".
3. Create two new vectors from the parents; call them "children", by applying crossover operation. The simplest form of the crossover operation is the "single-point" crossover through which the two parent vectors are cut at integer I and the coefficients to the right of this cut point are swapped. I am randomly chosen from the set (1, K-1). For more type of crossover operations, see Duffy and McNeils (2001). For instance, if K=4, and I=2, the crossover operation will create two children (C1 and C2) out of two parents (P1 and P2) by swapping the last two elements of the parent vectors as shown below.

$$\begin{bmatrix} \theta_{11} \\ \theta_{12} \\ --- \\ \theta_{13} \\ \theta_{14} \end{bmatrix}_{P1} \begin{bmatrix} \theta_{21} \\ \theta_{22} \\ --- \\ \theta_{23} \\ \theta_{24} \end{bmatrix}_{P2} \Rightarrow \begin{bmatrix} \theta_{11} \\ \theta_{12} \\ --- \\ \theta_{23} \\ \theta_{24} \end{bmatrix}_{C1} \begin{bmatrix} \theta_{21} \\ \theta_{22} \\ --- \\ \theta_{13} \\ \theta_{14} \end{bmatrix}_{C2}$$

4. Apply mutation operation to each element of the two children vectors. Under this operation, each element of the children vectors are subject to a shock with a small probability, $\mu > 0$. The probability is usually given by μ=0.15 + 0.33/G, where G is the size of the generation. The mutation operation result is as follows (Michalewicz (1996)).

$$\hat{\theta} = \begin{cases} \hat{\theta} + s[1 - r_2^{(1-t/T)^b}] & if\ r_1 > 0.5 \\ \hat{\theta} - s[1 - r_2^{(1-t/T)^b}] & if\ r_1 < 0.5 \end{cases}$$

Where r1, r2 are two real numbers from [0,1] drawn randomly and s is random number from a standard normal distribution, t the generation number, T is the maximum number of generations, and b is a parameter that governs the degree to which the mutation operator is non-uniform.

5. Run a tournament between the four parent and children vectors (P1, P2, C1, and C2). The two best fit vectors, with lowest sum of squared errors, will survive and therefore pass to the next generation, while the other two will extinguish.
6. Repeat the process above, with parents returning to the population pool for possible selection again, until the next generation is populated with P coefficient vectors.
7. Evaluate the members of the current generation with the new generation according to the fitness criterion, and choose the best one for the next generation. This process is called "Elitism".
8. Create new generations with P population and evaluate the convergence by the behavior of the best member of each generation according to the fitness criterion. If there is little change in the fitness evaluation of the best member of each passing generation for about 50 generations, you can assume that the genetic search has converged to an optimum.

References

1- Chatfield, Chris (1993): "Neural Networks: Forecasting Breakthrough or Passing Fad?", International Journal of Forecasting 9, p. 1-3.

2- Diebold, F.X. and Mariano, R. S. (1995): "Comparing Predictive Accuracy", Journal of Business and Economic Statistics 13, p. 253-263.

3- Duffy, John and McNelis, P. D. (2001): Approximating and Simulating the Stochastic Growth model: Parameterized Expectations, Neural Networks, and the Genetic Algorithm. Journal of Economic Dynamics & Control, 25, 1273-1303.

4- Fu, Jiarong (1998): "A Neural Network Forecast of Economic Growth and Recession", the Journal of Economics XXIV, No. 1, p. 51-66.

5- Harvey, D.I., Leybourne S.J. and Newbold, P. (1997): "Tests for Forecast Encompassing", Journal of Business and Economic Statistics 16, p. 254-259.

6- Herrera, F., Lozano, M. and Verdegay, J. L. (1998): "Tackling Real-Coded Genetic Algorithms: Operators and Tools for Behavioral Analysis, Artificial Intelligence Review 12, p. 265-319.

7- Holden, K. , Peel, D. A. and Thompson, J. L. (1990): Economic forecasting: An Introduction. Cambridge: Cambridge University Press.

8- Holland, J. (1975): Adaptation in Natural and Artificial System. University of Michigan Press, Ann Arbor, Reprinted in 1992, MIT Press, Cambridge, MA.

9- Kuan, Chung-Ming and White ,Halbert (1994): "Artificial Neural Networks: An Econometric Perspective", Econometric Reviews 13, p. 1-91.

10- Maasoumi, E., Khotanzad, A. and Abaye, A. (1994): "Artificial Neural Networks for Some Macroeconomic Series: A First Report", Econometric Reviews 13, p. 105-122.

11- McNelis, Paul D. (1997): Neural Networks and Genetic Algorithms: Tools for Forecasting and Risk Analysis in Financial Markets. Department of Economics, Georgetown University.

12- Michalewicz, Z. (1996): Genetic Algorithms + Data Structures = Evolution Programs. Third Edition, Springer-Verlag, New York.

13- Mitchell, M. (1996): An Introduction to Genetic Algorithms. MIT Press, Cambridge, MA.

14- Moshiri Saeed, and Laura Brown (forthcoming), "Unemployment Variation over the Business Cycles: A Comparison among Forecasting Models", Journal of Forecasting.

15- Moshiri, Saeed and Norman Cameron, (2000): "Neural Network Versus Econometric Models in Forecasting Inflation", Journal of Forecasting 19, p. 201-217.

16- Moshiri, Saeed, Norman Cameron, and David Scuse, (1999): "Static, Dynamic, and Hybrid Neural Networks in Forecasting Inflation", Computational Economics 14, p. 219-235.

17- Perron, p. (1989): "The Great Crash, the Oil Price Shock and the Unit Root Hypothesis", Econometrica 57, p. 1361-1401.

18- Rumelhart, D. E., Hinton, G. E. and Williams, R. J. (1986): "Learning Internal Representations by Error Propagation", Parallel Distributed Processing 1, p. 318-62.

19- Stock, James H. and Watson, Mark W. (1998): "A comparison of Linear and Nonlinear Univariate Models for Forecasting Macroeconomic Time Series", National Bureau of Economic Research Working Paper 6607.

20- Swanson, Norman R. and White, Halbert (1995): "A Model-Selection Approach to Assessing the Information in the Term Structure Using Linear Models and Artificial Neural Networks", Journal of Business and Economic Statistics, Vol. 13, No. 3, p. 265-275.

21- Swanson, Norman R. and White, Halbert (1997): "A Model Selection Approach to Real-Time Macroeconomic Forecasting Using Linear Models and Artificial Neural Networks", Review of Economics and Statistics 79, p. 540-550.

22- Tkacz, Greg (1999): "Neural Network Forecasts of Canadian GDP Growth Using Financial Variables", Bank of Canada, mimeo.

23- Verkooijen, William (1996): "A Neural Network Approach to Long-Run Exchange Rate Prediction", Computational Economics 9, p. 51-65.

24- Zellner, Arnold (2002), My Experiences with Nonlinear Dynamic Models in Economics, Studies in Nonlinear Dynamics & Econometric, Volume 6, Issue2.