Using Intelligent Optimization Methods to Improve the Group Method of Data Handling in Time Series Prediction

Maysam Abbod and Karishma Deshpande

School of Engineering and Design,
Brunel University, West London, UK Uxbridge, UK. UB8 3PH
Maysam.Abbod@Brunel.ac.uk

Abstract. In this paper we show how the performance of the basic algorithm of the Group Method of Data Handling (GMDH) can be improved using Genetic Algorithms (GA) and Particle Swarm Optimization (PSO). The new improved GMDH is then used to predict currency exchange rates: the US Dollar to the Euros. The performance of the hybrid GMDHs are compared with that of the conventional GMDH. Two performance measures, the root mean squared error and the mean absolute percentage errors show that the hybrid GMDH algorithm gives more accurate predictions than the conventional GMDH algorithm.

Key words: GMDH; GA; PSO; time series; prediction; finance.

1 Introduction

Forecasting future trends of many observable phenomena remains of great interest to a wide circle of people. This requirement has maintained a high rate of activity in various research fields dedicated to temporal prediction methodologies. Two such important application domains are financial markets and environmental systems. Predicting such systems has been attempted for decades but it remains such a challenging task for a wide array of modeling paradigms.

The foreign exchange market is a large business with a large turnover in which trading takes place round the clock and all over the world. Consequently, financial time series prediction has become a very popular and ever growing business. It can be classified as a real world system characterized by the presence of non-linear relations. Modeling real world systems is a demanding task where many factors must be taken into account. The quantity and quality of the data points, the presence of external circumstances such as political issues and inflation rate make the modeling procedure a very difficult task. A survey of the different methods available for modeling non-linear systems is given in Billings [1].

Some researchers tried auto-regressive methods to predict foreign currency exchange rates [2]. Episcopos and Davis [6] used the GARCH and ARIMA statistical

methods to identify the model. These methods have not always produced good results which have urged scientists to explore other more effective methods.

During the last two decades adaptive modeling techniques, like neural networks and genetic algorithms, have been extensively used in the field of economics. A list of examples using such methods is given in Deboeck [4] and Chen [3]. An extension to the genetic algorithm paradigm [9] is the versatile genetic programming method introduced by Koza [11]. Scientists were quick to use this method in many aspects of mathematical modeling.

An alternative non-linear modeling method, which was introduced in the late sixties, is the Group Method of Data Handling [11]. Since its introduction researchers from all over the world who used the GMDH in modeling were astonished by its prediction accuracy. Many of these applications were in the field of time series prediction. Parks et al [14] used the GMDH to find a model for the British economy. Ikeda et al [10] used a sequential GMDH algorithm in river flow prediction. Hayashi and Tanaka [8] used a fuzzy GMDH algorithm to predict the production of computers in Japan. Robinson and Mort [15] used an optimized GMDH algorithm for predicting foreign currency exchange rate.

In this paper we describe how the GMDH can be used in conjunction with Genetic Algorithms (GA) and with Particle Swarm Optimization (PSO) to improve the prediction accuracy of the standard GMDH algorithm when it is applied to time series in the form of financial data.

2 The Data

In this paper both the hybrid GMDHs and the conventional GMDH algorithms were used to make one step ahead prediction of the exchange rate from US Dollars to Euros (USD2EURO). Values from 29 September, 2004 to 5 October 2007 were obtained from the website `www.oanda.com`, a total of 1102 points (Fig. 1). The first 1000 points were used in the training and checking of the GMDH algorithm. The last 102 points were unseen by the algorithm throughout its computation. The performance of the algorithm was evaluated on the last 102 points of the data.

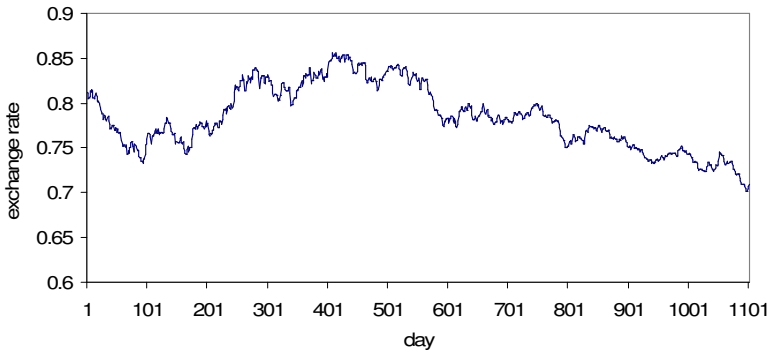


Fig. 1. USD2EURO from 29 Sept, 2004 to 5 Oct, 2007.

3 The Group Method of Data Handling (GMDH)

GMDH method was developed by Ivakhnenko [11] as a rival to the method of stochastic approximation. The proposed algorithm is based on a multilayer structure using, for each pair of variables, a second order polynomial of the form:

$$Y = A_O + A_1X_I + A_2X_J + A_3X_IX_J + A_4X_I^2 + A_5X_J^2 \quad (1)$$

where x_i and x_j are input variables and y is the corresponding output value. The data points are divided into training and checking sets. The coefficients of the polynomial are found by regression on the training set and its output is then evaluated and tested for suitability using the data points in the checking set. An external criterion, usually the mean squared error (mse), is then used to select the polynomials that are allowed to proceed to the next layer. The output of the selected polynomials becomes the new input values at the next layer. The whole procedure is repeated until the lowest *mse* is no longer smaller than that of the previous layer. The model of the data can be computed by tracing back the path of the polynomials that corresponded to the lowest *mse* in each layer.

4 Intelligent systems Optimization

4.1 Genetic Algorithms

GAs are exploratory search and optimization methods that were devised on the principles of natural evolution and population genetics [9]. Unlike other optimization techniques, a GA does not require mathematical descriptions of the optimization problem, but instead relies on a cost-function, in order to assess the fitness of a particular solution to the problem in question. Possible solution candidates are represented by a population of individuals (generation) and each individual is encoded as a binary string, referred to as a chromosome containing a well-defined number of alleles (1's and 0's). Initially, a population of individuals is generated and the fittest individuals are chosen by ranking them according to *a priori*-defined fitness-function, which is evaluated for each member of this population. In order to create another better population from the initial one, a mating process is carried out among the fittest individuals in the previous generation, since the relative fitness of each individual is used as a criterion for choice. Hence, the selected individuals are randomly combined in pairs to produce two off-springs by *crossing over* parts of their chromosomes at a randomly chosen position of the string. These new off-springs represent a better solution to the problem. In order to provide extra excitation to the process of generation, randomly chosen bits in the strings are inverted (0's to 1's and 1's to 0's). This mechanism is known as *mutation* and helps to speed up convergence and prevents the population from being predominated by the same individuals. All in all, it ensures that the solution set is never naught. A compromise, however, should be reached between too much or too little excitation by choosing a small probability of mutation.

4.2 Practical Swarm Optimization

Particle Swarm Optimization is a global minimization technique for dealing with problems in which a best solution can be represented using position and velocity components. All particles remember the best position they have seen, and communicate this position to the other members of the swarm. The particles will adjust their own positions and velocity based on this information. The communication can be common to the whole swarm, or be divided into local neighborhoods of particles [12].

5 Results and Discussions

The GMDH network, as mentioned earlier, uses a quadratic polynomial as the transfer function in each layer. A question arises: what if the relationship between the input and the output is not best described by a quadratic polynomial? This leads to another question: Could a better global optimization method replace the regression technique to fit the quadratic polynomial and generate a function that described the input-output relationship more accurately, leading to an improvement in the prediction accuracy of the GMDH algorithm?

The GA- and PSO-GMDH algorithm is similar to the conventional GMDH algorithm except that the transfer function polynomials are fitted using better optimization algorithms, namely PSO and GA. The optimization algorithm is applied to the data while the GMDH iterates through in order to find the best function that maps the input to the output. It is important to note that the GA and PSO are applied separately in order to find an exact mapping between the input and the output at different iteration stages.

Two performance measures were used in assessing the accuracy of the conventional GMDH and the hybrid GMDH: the mean absolute percentage error, denoted MAPE, (equation 2) and the widely used root mean squared error, RMSE, (equation 3).

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\frac{abs(Y_i - Z_i)}{Y_i}\times 100\% \tag{2}$$

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_i - Z_i)^2} \tag{3}$$

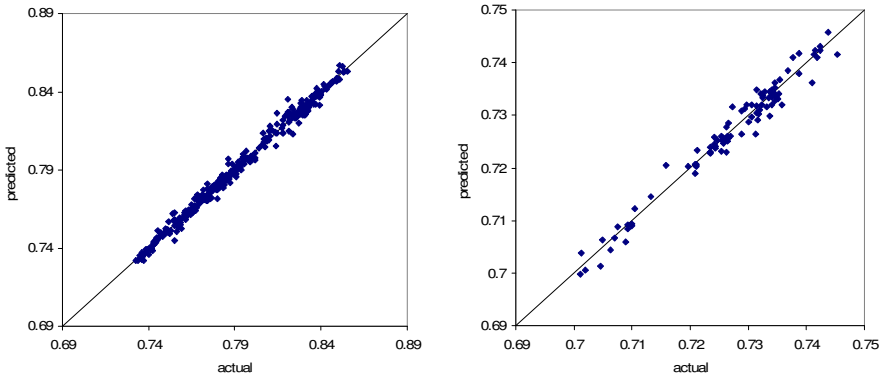
where n is the number of variables, Y is the actual output and Z is the predicted output.

The PSO algorithm was set to a population size of 100, while the inertial cognitive and social constants are as follows [5]: w = 0.7298; c1 = 1.49618; c2 = 1.49618

The GA algorithm was set as a binary code of 12 bits for each of the parameters of the polynomial. The polynomial has 6 parameters so that makes the chromosome 12×6=72 bits long. The 12 bit binary number maps to a search space of -2 to +2 units in decimals. The 12 bits settings allows 4096 steps for a search space of 4 units which

is a step size of 4/4096= 0.0000976. The GA was set with a mutation rate of 0.1 and a single point crossover at a rate of 0.9.

The results are classified into two types, the training phase and the testing phase. The former phase involves 1000 data points which are used to fit the polynomials at different GMDH iterations. The later stage is to test the GMDH system to data that the system has never experienced before, in this case 102 data points were used. Figs. 2a and 2b shows the predicted against the actual data for a standard GMDH for training and testing respectively. While Figs. 3a and 3b show the same figures but for the PSO-GMDH, and finally Figs. 4a and 4b shows the results of the GA-GMDH version.

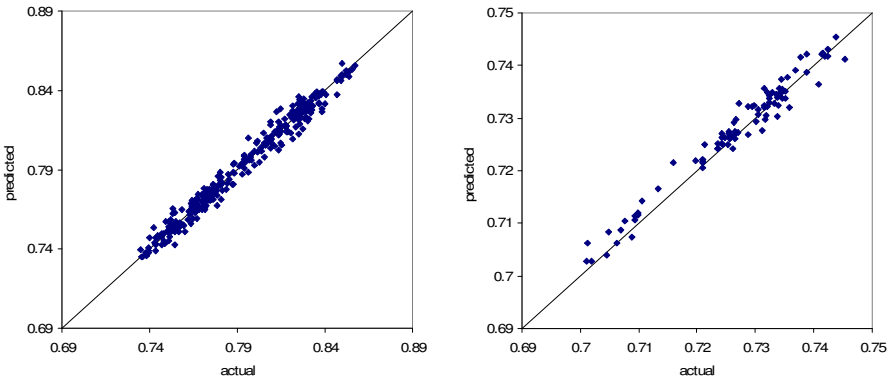


(a) training data (b) testing data

Fig. 2. Graphs of GMDH predicted against actual USD2EURO.

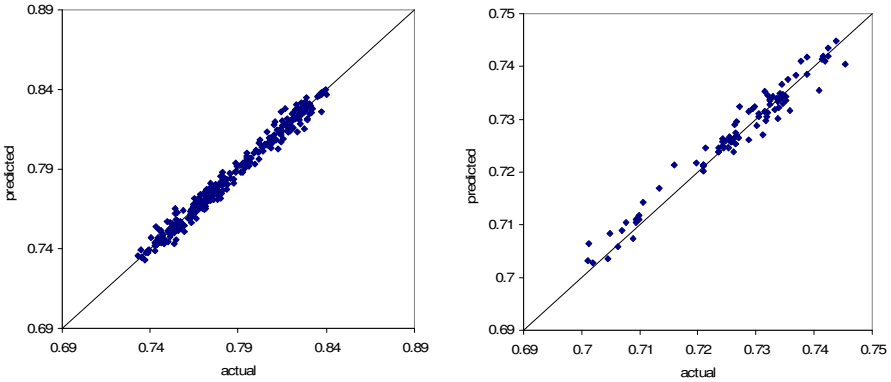
The GA-GMDH algorithm usually produces better results due to its global search features. This comes at a price of being slow and computationally intensive. For accurate results, large generation and wide searching space are required. This can be accommodated by creating large chromosomes with bigger constraints for increasing the search accuracy. In contrast, PSO is fast and has no constraints on the search space. But it produces less accurate results globally, but more accurate locally. Combining both algorithms with GMDH provides better accuracy and speed. By starting with a low accuracy GA (fast calculations) for finding the rough global minima, then switching to PSO for finding a more accurate minima in the global GA minima area.

The GA was set to 8 bit settings for each variable, and then preceded by PSO with the same previous settings. Figs. 5a and 5b show the predicted against the actual data for the GA-PSO-GMDH for training and testing respectively



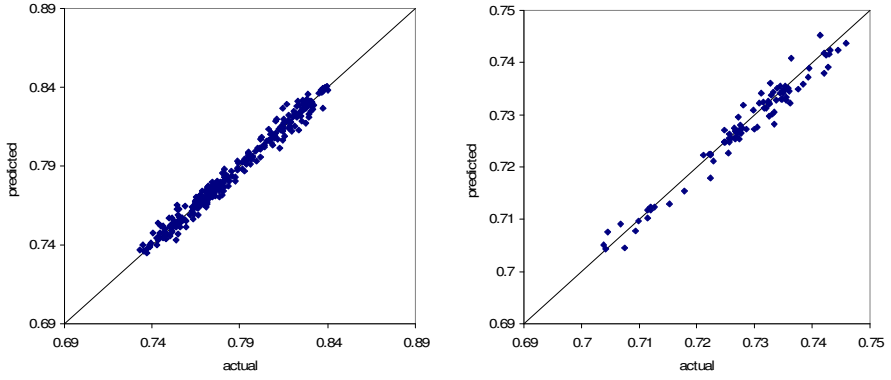
(a) training data (b) testing data

Fig. 3. Graphs of PSO-GMDH predicted against actual USD2EURO.



(a) training data (b) testing data

Fig. 4. Graphs of GA-GMDH predicted against actual USD2EURO.



(a) training data (b) testing data

Fig. 5. Graphs of GA-PSO-GMDH predicted against actual USD2EURO.

The results for both networks when applied to the entire training and testing data are given in Tables 1 and 2 respectively.

Table 1. Results for USD2EURO training data.

	MAPE	RMSE
Linear regression	0.2904	0.003993
GMDH	0.2772	0.003101
PSO-GMDH	0.2604	0.003085
GA-GMDH	0.2602	0.002993
GA-PSO-GMDH	0.2546	0.002989

Table 2. Results for USD2ERO testing data.

	MAPE	RMSE
Linear regression	0.1986	0.001841
GMDH	0.1985	0.001831
PSO-GMDH	0.1860	0.001611
GA-GMDH	0.1653	0.001312
GA-PSO-GMDH	0.1416	0.001302

It is evident from the values of both measures that the GA-and PSO-GMDH performs better than the conventional GMDH. Values of the percentage improvement in both performance measures for all the data are given in Table 3.

Table 3. GMDH percentage improvement to the testing data performance.

	MAPE Percentage improvement	RMSE Percentage improvement
PSO-GMDH	6.3	12.0
GA-GMDH	16.7	28.3
GA-PSO-GMDH	28.6	28.9

Graphs of the actual against predicted by the conventional GMDH algorithm and predicted by PSO-GMDH algorithm for the USD2EURO exchange rates are shown in Fig. 6. While Fig. 7 shows the predictions of GA-GMDH algorithm and the combinations of GA, PSO and GMDH (GA-PSO-GMDH). The prediction performance of the GMDH network depends on the number of generations over which the algorithm is allowed to evolve. The results given in this paper were produced after only 4 iterations (ambiguous – generations in terms of GA).

Several GMDH runs, using both networks, were carried out each with a different number of generations. It was found that as the individuals were allowed to evolve over a higher number of generations, the value of the minimum of the selection criterion in each generation was decreased. On the other hand, it was found that the

performance measures became progressively worse. This was due to the fact that the algorithms became overspecialized for the data they were trained on to the detriment of their ability to predict the unseen data.

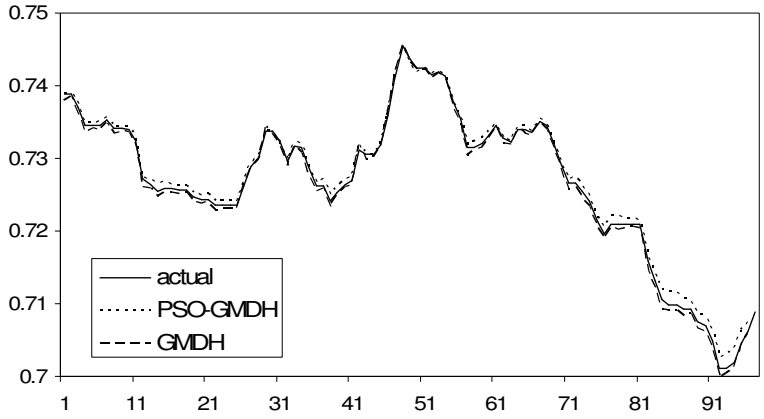


Fig. 6. Graphs of actual testing data, GMDH and PSO-GMDH predictions.

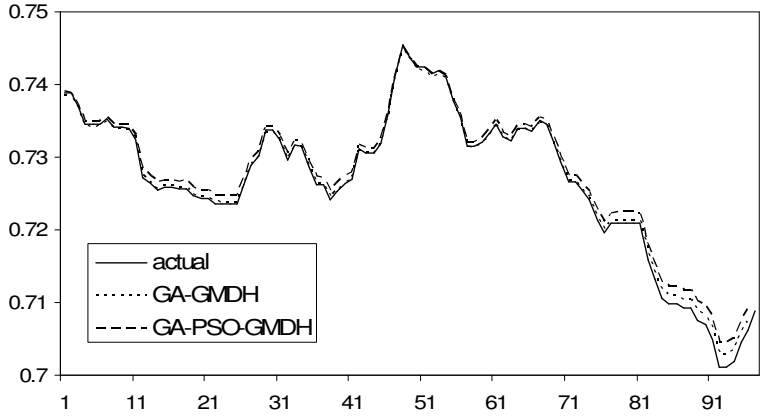


Fig. 7. Graphs of actual testing data, GA-GMDH and GA-PSO-GMDH predictions.

6 Conclusions

It was shown in this paper that the performance of the conventional GMDH algorithm can be improved significantly if some information about the type of the relationship between the input and output was available and used in the GMDH run.

The fitness of the best individual in each generation improved while the individuals were allowed to evolve over more generations. This had the reverse effect on the overall performance of the network for the unseen data points. This is an area

that is in need of further investigation in order to find a suitable point for terminating the building of the polynomial process.

When the same procedure carried out in this paper was repeated a few times using a different size of training and checking sets, it was noticed that the performance of both GMDH networks, particularly the conventional one, was greatly affected. This leads to the following hypothesis: there might exist a rule for dividing the sample of data that when applied the accuracy of the GMDH algorithm reaches its optimum. Work needs to be carried out to investigate the validity of this hypothesis.

The work carried out in this paper has provided an insight into the way the GMDH deals with time series prediction. While it was shown that it is possible to improve its prediction performance, the GMDH remains a robust and popular method of mathematical modeling.

References

1. Billings, S.A.: Identification of Non-Linear Systems – a Survey. IEE Proceedings on Control Theory and Applications, vol. 127, no. 6, pp. 272--285 (1980)
2. Chappell, D., Padmore, J., Mistry, P., Ellis, C.: A Threshold Model for the French Franc/Deutschmark Exchange Rate. Journal of Forecasting, vol. 15, no. 3, pp. 155--164 (1995)
3. Chen, C.H.: Neural Networks for Financial Market Prediction, IEEE International Conference on Neural Networks, vol. 7, pp. 1199--1202 (1994)
4. Deboeck, G. (Ed.): Trading of the Edge: Neural, Genetic and Fuzzy Systems for Chaotic Financial Markets. John Wiley & Sons Inc (1994)
5. Eberhart, R.C., Shi, Y.: Comparing Inertia Weights and Constriction Factors in Particle Swarm Optimization. Proceedings of the Congress on Evolutionary Computing, San Diego, USA, pp. 84--89 (2000)
6. Episcopos, A., Davis, J.: Predicting Returns on Canadian Exchange Rates with Artificial Neural Networks and EGARCH-M Models. Abu-Mostafa, Y., Moody, J., Weigend, A. (Eds.) Neural Networks Financial Engineering. Proceedings of the third International Conference on Neural Networks in the Capital Markets, Singapore, London, World Scientific, pp. 135--145 (1995)
7. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley (1989)
8. Hayashi, I., Tanaka, H.: The Fuzzy GMDH Algorithm by Possibility Models and its Application. Fuzzy Sets and Systems, vol. 36. pp. 245--258 (1990)
9. Holland, J.H.: Adaption in Natural and Artificial Systems. The University of Michigan Press, Ann Arbor (1975)
10. Ikeda, S., Ochiai, M., Sawaragi, Y.: Sequential GMDH Algorithm and Its Application to River Flow Prediction. IEEE Transaction on Systems, Man and Cybernetics, July (1976)
11. Ivakhnenko, A.G.: The Group Method of Data Handling-A Rival of the Method of Stochastic Approximation. Soviet Automatic Control, vol. 13 c/c of automatika, vol. 1, no. 3, pp. 43--55 (1968)

12. Kennedy, J., Eberhart, R.: Particle Swarm Optimization. Proc. IEEE Int'l. Conf. on Neural Networks, Perth, Australia, pp. 1942--1948, November (1995)

13. Koza, J.R.: Genetic Programming: on the programming of computers by means of natural selection. MIT Press, Cambridge (1992)

14. Parks, P., Ivakhnenko, A.G., Boichuk, L.M., Svetalsky, B.K.: A Self-Organizing Model of British Economy for Control with Optimal Prediction using the Balance-of-Variables Criterion. Int. J. of Computer and Information Sciences, vol. 4, no. 4 (1975)

15. Robinson, C., Mort, M.: Predicting Foreign Exchange Rates Using Neural and Genetic Models. Proceedings of 2nd Asian Control Conference, Seoul, Korea, vol. 3, pp. 115--118, July 22-25 (1997)