Evolving Sparsely Connected Neural Networks for Multi-Step Ahead Forecasting

Juan Peralta Donate
University Carlos III of Madrid
Av Universidad 30 28911
Leganes, Spain
jperalta@inf.uc3m.es

Paulo Cortez
University of Minho
Campus Azurém,4800-058
Guimarães, Portugal
pcortez@dsi.uminho.pt

German Gutierrez Sanchez
University Carlos III of Madrid
Av Universidad 30 28911
Leganes, Spain
ggutierr@inf.uc3m.es

Araceli Sanchis de Miguel
University Carlos III of Madrid
Av Universidad 30 28911
Leganes, Spain
masm@inf.uc3m.es

ABSTRACT

Time Series Forecasting (TSF) is an important tool to support decision making. Artificial Neural Networks (ANN) are innate candidates for TSF due to advantages such as nonlinear learning and noise tolerance. However, the search for the best ANN is a complex task that highly affects the forecasting performance. In this paper, we propose a novel Sparsely connected Evolutionary ANN (SEANN), which evolves more flexible ANN structures to perform multi-step ahead forecasts. This approach is compared with a similar strategy but that only evolves fully connected ANNs (FEANN) and a conventional TSF method (i.e. ARIMA methodology). A set of six time series, from different real-world domains, was used in the comparison. Overall, the obtained results reveal the proposed SEANN approach as the best forecasting method, optimizing more simpler structures and requiring less computational effort when compared with the fully connected evolutionary ANN strategy.

Categories and Subject Descriptors

I.2.6 [**Learning**]: Connectionism and neural nets; C.1.m [**Miscellaneous**]: Hybrid systems

General Terms

Algorithms, Design

1. INTRODUCTION

TSF has become increasingly used in areas such as agriculture, finance, management, production or sales. In this paper we adopt multilayer perceptrons ANN for TSF and a crucial issue is the design of the best forecasting model. Instead of manually tunning the ANN, one interesting approach is to perform a fully automatic ANN design based on evolutionary computation (EANN) [1]. Yet, several of these works make use of full connected structures, evolving only ANN hyperparameters, such as number of input and hidden nodes [3]. In contrast, the evolution of sparsely connected structures allows the search of more flexible ANN models, with the potential of achieving better forecasts with simpler models, which may contain less connections. In this paper, we propose the use of EDA, as the search engine of an EANN, in order to evolve sparsely connected multilayer perceptrons for multi-step ahead forecasting. Such approach is compared against the fully connected EDA ANN and also the popular ARIMA methodology, using several real-world time series from distinct domains. The paper is organized as follows. Section 2 explains sparsely connected evolutionary artificial neural networks implementation. In Section 3 results and conclusions are tackled.

GECCO'11, July 12–16, 2011, Dublin, Ireland.
ACM 978-1-4503-0690-4/11/07.

2. SPARSELY CONNECTED EANN

Miller et al. [2] identified two approaches to code the topology in a string: direct encoding scheme and indirect encoding scheme. One advantage of the direct approach, adopted by Whitley et al. [4], is that it is easy to evolve networks with special connectivity properties. The proposed Sparsely EANN (SEANN) works as the previous FEANN [3], except that now we can evolve more flexibly structures. To achieve this, we adopted a direct binary encoding scheme. We added a binary codification to our decimal previous one [3], where the last binary digits (i.e. connection matrix) set which are the active connections of the model.

In Fig. 1, it can be observed an example of how the direct binary codification works, in order to obtain the ANN connection matrix from the chromosome. In general, each matrix cell represents a valid connection between an incoming node (at the row) with the outgoing node (at the column). In the example, the third digit ($b_3 = 1$) sets a connection between the first input node (relative to the first time lag) and the third hidden node. The exception is the last row, which represents the connections between the hidden nodes (at the columns) and the output one (the last row). By default, the largest possible connection matrix is always set, with the dimensions $Row \times Col$, although the real dimensions are limited by the i and h values. We have opted for this solution to set the same fixed length of the chromosome, for all the individuals.

Table 1: Comparison of the best models optimized by FEANN and SEANN.

Series	FEANN				SEANN				R_c	R_{te}
	inputs (i)	hidden (h)	connect. (c)	time (min)	inputs (i')	hidden (h')	connect. (c)	time (min)		
Passengers	49.2	67.4	3383.4	165	49.6	71.4	1813.6	71	46.4%	56.9%
Temperature	63.6	64.8	4186.1	315	65.0	59.8	2011.1	114	51.9%	63.8%
Dow-Jones	35.8	48.8	1795.8	161	38.6	93.6	1868.8	73	-4.1%	54.7%
Abraham12	30.4	117.8	3698.9	270	21.2	99.4	1118.4	89	69.8%	67.0%
Quebec	14.6	136.6	2131.0	6603	12.2	111.2	824.8	5221	61.3%	20.9%
Mackey-Glass	13.0	90.4	1265.6	8529	14.6	117.8	924.6	5590	26.9%	34.5%

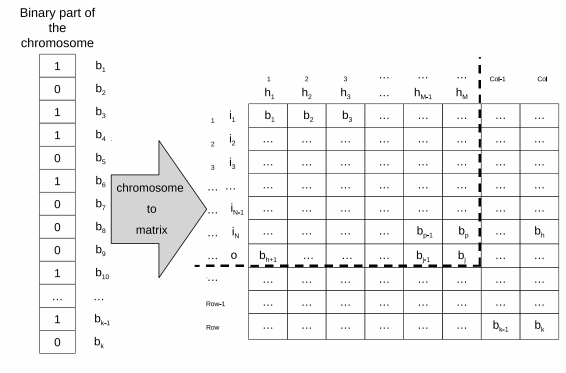

Figure 1: Example of the process to obtain a connection matrix from a given chromosome.

3. RESULTS AND CONCLUSIONS

Each EANN method explained in this paper (i.e. FEANN and SEANN) was executed five times for each time series and we present the mean of the five runs. The forecasted values were compared with the real ones and SMAPE error metric was computed. As a baseline comparison, we have chosen the popular forecasting tool ForecastPro© (FP). The obtained results are shown in Table 2.

Table 2: Comparison of the forecasting errors (%SMAPE, best values in bold)

Series	FP	FEANN	SEANN
Passengers	4.50	3.39	**3.20**
Temperature	**3.42**	3.51	4.17
Dow-Jones	**4.78**	6.28	5.79
Abraham 12	6.20	6.42	**4.48**
Quebec	10.36	10.83	**8.00**
Mackey-glass	26.20	7.06	**4.03**

An analysis to tables shows that the proposed SEANN provides in general better forecasts when compared with FEANN. The sparsely connected method outperforms the fully connected one in five cases. Also there are interesting improvements of SEANN over FEANN (e.g. a difference of 1.9, 2.8 and 3 pp for the last three series). SEANN also outperforms the popular ARIMA methodology, comparing favorable against FP in four cases. The average also ranks FEANN as the best method. Table 1 compares the characteristics of the best ANNs evolved by both EANN approaches. For each series and evolutionary method, we report the number of inputs (i or i'), hidden nodes (h or h'), total number of connections (c) and computational effort (in min). The table shows that, in general, FEANN obtains simpler ANN structures. In particular, high reduction rates were achieved for Abraham12 and Quebec series. Furthermore, SEANN is always faster than FEANN, requiring much less computation in all time series considered. The three different forecasting methods were compared over six distinct time series. Also, the obtained multi-step forecasts were analyzed under SMAPE error criteria. The results of the experiments held reveal the proposed FEANN as the best forecasting method. Moreover, when compared with FEANN, SEANN tends to favour simpler structures and requires less computational effort. An interesting future research direction is to evolve other TSF base learners, such as Support Vector Machines.

4. ACKNOWLEDGMENTS

This work was supported by the University Carlos III of Madrid and by Community of Madrid under project CCG10-UC3M/TIC-5174.

5. REFERENCES

[1] P. Cortez, M. Rocha, and J. Neves. Evolving time series forecasting arma models. *Journal of Heuristics*, 10:415–429, July 2004.

[2] G. F. Miller, P. M. Todd, and S. U. Hegde. Designing neural networks using genetic algorithms. In *Proceedings of the third international conference on Genetic algorithms*, pages 379–384, San Francisco, CA, USA, 1989. Morgan Kaufmann Publishers Inc.

[3] J. Peralta, G. Gutierrez, and A. Sanchis. Time series forecasting by evolving artificial neural networks using genetic algorithms and estimation of distribution algorithms. In *Neural Networks (IJCNN), The 2010 International Joint Conference on*, pages 1 –8, July 2010.

[4] D. Whitley, T. Starkweather, and C. Bogart. Genetic algorithms and neural networks: optimizing connections and connectivity. *Parallel Computing*, 14(3):347 – 361, 1990.