Evolving Time-Lagged Feedforward Neural Networks for Time Series 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 both individual and organizational decisions. In this work, we propose a novel automatic Evolutionary Time-Lagged Feedforward Network (ETLFN) approach for TSF, based on an Estimation Distribution Algorithm (EDA) that evolves not only Artificial Neural Network (ANN) parameters but also which set of time lags are fed into the forecasting model. Such approach is compared with similar strategy that only selects ANN parameter and the conventional TSF ARIMA methodology. Several experiments were held by considering six time series from distinct domains. The obtained multi-step ahead forecasts were evaluated using SMAPE error criteria. Overall, the proposed ETLFN method obtained the best forecasting results. Moreover, it favors simpler neural network models, thus requiring less computational effort.

Categories and Subject Descriptors

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

General Terms

Algorithms, Design

1. INTRODUCTION

In this paper, we focus on ANN as Computational Intelligence (CI) method to TSF. When applying ANN to TSF, the Time-Lagged Feedforward Neural Network (TLFN) is a popular approach [1]. The TLFN adopts a multilayer perceptron ANN as the learning base model and uses a sliding time window method to create supervised training examples. The sliding time window defines a set of time lags that are used as inputs by the ANN. A crucial issue is the design of the best TLFN model for a particular time series. Instead of manually tunning the ANN, one interesting approach is to perform a fully automatic ANN design based on evolutionary computation (EANN). Yet, once the number of input nodes was set, all input time lags were used. In this paper, we compare such approach with a novel Evolutionary TLFN (ETLFN), which uses (EDA) to also optimize which time lags are used by the TLFN model. The paper is organized as follows. In Section 2, we describe the ETLFN approach. In Section 3 we analyze the results and conclude.

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

2. EVOLUTIONARY TIME LAGGED FEEDFORWARD NETWORK

Every time a new individual (i.e. ANN) is generated, training and validation patterns subsets have to be obtained. In previous work [2], if the ANN had k input nodes, all k previous time series values were used to generate the patterns subsets. Here we select the relevant previous time lags of the series to generate the patterns. So, we add an extra information to the chromosome explained in [2]. The new information, new genes in binary codification, defines if the time lag is (or not) used by the model.

Table 1: Comparison of %SMAPE errors (best values in bold).

Series	FP	EEANN	ETLFN
Passengers	4.50	**3.39**	3.78
Temperature	**3.42**	3.51	3.74
Dow-Jones	**4.78**	6.28	5.54
Abraham 12	6.20	6.42	**4.34**
Quebec	10.36	10.83	**9.30**
Mackey-glass	26.20	7.06	**4.93**

Yet, it should be noted that number of input nodes of the ANN (i) now sets the maximum number of input nodes, i.e., only the up to b_i time lags are considered by the model. As shown in Fig. 1, the number of input nodes of the ANN are set not only by i, but also depends on the binary encoding, which only activates some of the lags, and only these will the inputs of the ANN.

Table 2: **Example of the best ETLFN forecasting models.**

Series	i	sliding window	lag deletions	inputs
Passengers	49	{3,4,5,9,13,15,16,18,22,23,24,26,28,32,33,42,44,45,46,}	31	18
Temperature	67	{3-5,7,9-12,15,17,19,21,22,28,34-36,38,41,46-49,53-57,59-61,65,66}	32	35
Dow-Jones	41	{3,4,7,8,11,13,17,22,25-28,30,32,34,36,37,41}	23	18
Abraham12	30	{6,8,11-13,17-19,21,23,25,27,28}	17	13
Quebec	43	{2,5,8,10,14,16,17,23,26,29,30,34,37,41-43}	26	17
Mackey-Glass	25	{6,8,10,11,13,15-17,19-25}	10	15

Table 3: Comparison of the best models optimized by EEANN and ETLFN.

Series	EEANN				ETLFN				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	23.0	72.0	1728.0	61	48.9%	63.0%
Temperature	63.6	64.8	4186.1	315	37.6	80.6	3111.2	199	25.7%	36.8%
Dow-Jones	35.8	48.8	1795.8	161	21.4	64.8	1451.5	67	19.2%	58.4%
Abraham12	30.4	117.8	3698.9	270	16.0	95.4	1621.8	109	56.2%	59.6%
Quebec	14.6	136.6	2131.0	6603	16.2	115.4	1984.9	3906	6.9%	40.8%
Mackey-Glass	13.0	90.4	1265.6	8529	12.0	120.4	1565.2	8493	-23.7%	0.4%

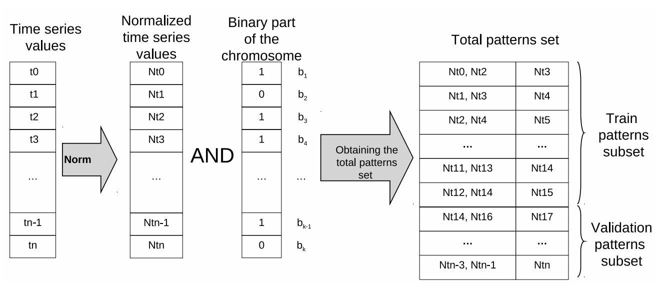

Figure 1: Process to obtain training and validation sets with time lag selection.

3. RESULTS AND CONCLUSIONS

Each evolutionary approach (i.e. EANN and ETTFN) was executed five times for all time series and we report the mean results of these five executions. To evaluate the error for each method, forecasted values were compared with real values, under the SMAPE criteria. As a baseline comparison, we have chosen the popular forecasting tool ForecastPro© (FP). The obtained results are shown in Table 1. Table 1 shows that time lag selection strategy (ETLFN) outperforms the no time lag selection approach (EEANN) in four of the six datasets. The average result, when considering all time series, also favors ETLFN when compared with EEANN and FP. ETLFN outperforms FP in four of the six series. As an example, we present the best individuals achieved by ETLFN during a given execution in Table 2, where the third column (sliding window) shows the selected time lags (i.e. b_j values up to i). The binary time lag selection genes perform a substantial pruning of the maximum number of input nodes (i), thus leading to much simpler models. Table 3 compares the characteristics of the best ANNs evolved by EEANN and ETLFN. For each series and evolutionary method, we report the number of inputs (I), hidden nodes (h), total number of connections (c) and computational effort (in min). In general, ETLFN obtains simpler ANN structures. In particular, high reduction rates were achieved for Passengers and Abraham12 series. The exception is for Mackey-Glass, where ETLFN optimizes an ANN with more hidden nodes when compared with EEANN. Moreover, ETLFN is always faster than EEANN, requiring much less computation in all cases except Mackey-Glass.

We compared ETLFN and EEANN over six distinct time series and the obtained multi-step forecasts were analyzed under SMAPE error criteria. The ETLFN approach achieved competitive results, outperforming both EEANN and also the well known automatic modeling ForecastPro tool. Furthermore, when compared with EEANN, ETLFN tends to optimize simpler ANN structures, with less input nodes and total number of connections, thus requiring much less computational effort. In the future, an interesting research direction is to use sparsely connected ANNs, selecting not only the time lags but also which connections are used by the ANN.

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. *Time Series Forecasting by Evolutionary Neural Networks*, chapter III, pages 47–70. Idea Group Publishing, USA, 2006.

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