ESANN'2000 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 26-28 April 2000, D-Facto public., ISBN 2-930307-00-5, pp. 347-352

Chaotic Time Series Prediction using the Kohonen Algorithm

Luis Monzón Benítez*,**,†, Ademar Ferreira* and
Diana I. Pedreira Iparraguirre**

*Departamento de Engenharia de Telecomunicações e Controle
Escola Politécnica da Universidade de São Paulo CEP: 05508 900
São Paulo SP, Brazil.
lmonzon@lac.usp.br, ademar@lac.usp.br

**Universidade Medica Carlos J. Finlay.
Departamento de Bioestatística e Computação.
Camaguey, Cuba.
ivette@finlay.cmw.sld.cu

Abstract.
Deterministic nonlinear prediction is a powerful technique for the analysis and prediction of time series generated by nonlinear dynamical systems. In this paper the use of a Kohonen network as a component of one deterministic nonlinear prediction algorithm is suggested. In order to evaluate the performance of the proposed algorithm, it was applied to the prediction of time series generated by two well known chaotic dynamical systems and the results were compared with those obtained using the Modified Method of Analogues with the same time series. The generated time series were corrupted by superimposed observational noise. The experimental results have shown that the Kohonen network can learn the neighborhood relations present in the reconstructed attractor of the time series and that good predictions can also be obtained with the proposed algorithm.

1. Introduction

In the last decades many different techniques have been developed for the analysis and prediction of time series, ranging from the well-known Box and Jenkins models [1] to the latest neural network based ones [3, 4]. The most used approach consists in considering the time series as a realization of a stochastic process and in applying the different methods devised in the framework of

The authors would like to thank Professor Manuel Lazo of the Institute for Cibernetics, Mathematics and Physics, Cuba, for his useful comments.
†Partially supported by the National Council for Scientific and Technological Development (CNPq), Brazil.

Statistics to obtain a model for the process [1, 3]. Today, with the latest results concerning deterministic chaos, even if the data look random apparently it is important to consider the possibility that the time series was generated by a low order nonlinear deterministic dynamical system [4, 5]. One of the techniques developed for the analysis of chaotic time series is deterministic nonlinear prediction[5].

In this paper the use of a Kohonen neural network as a component of one deterministic nonlinear prediction algorithm is suggested. The algorithm was originally proposed by Lorenz [7] and later modified by Ikeguchi and Aihara [5] and is known as the Modified Method of Analogues (MMA). The paper is organized as follows: In section 2 the MMA is explained and in the next section the proposed prediction algorithm is introduced. In section 4 some experiments are conducted to evaluate the performance of the algorithm and the results are discussed. Finally, the main conclusions are highlighted.

2. The Modified Method of Analogues

This method was introduced by Lorenz in 1969 [7] and was named "The Method of Analogues" because the prediction of a point in the attractor is obtained analogizing the movement of its nearest neighbor. In 1995, Ikeguchi and Aihara [5] proposed to use M nearest neighbors instead of only one neighbor to find the prediction of the point. They named this algorithm "The Modified Method of Analogues".

Let v_T be the point of the state space whose future behavior will be predicted. First, the M nearest neighbors of v_T are searched from all the points in the attractor and designated by $v_{k_i}(i = 1, 2, ..., M)$, with v_{k_i} being the nearest to the point v_T and continuing in ascending order. After p time steps v_{k_i} is mapped to v_{k_i+p}.

The prediction $\hat{v}_{T+p}$ of v_T is given as follows:

$$\hat{v}_{T+p} = \frac{\sum_{i=1}^{M} \frac{v_{k_i+p}}{|v_{k_i} - v_T|}}{\sum_{i=1}^{M} \frac{1}{|v_{k_i} - v_T|}}. \tag{1}$$

Our aim with this work is to use a Kohonen network that properly trained with the different points of the attractor could be used to obtain the neighbors v_{k_i+p} of the point v_T. This imply that the complete prediction process could be done extremely fast (once the network has learned the neighborhood relations of the attractor), because of the high degree of parallelism associated with the Kohonen neural network.

3. The proposed prediction algorithm.

As it was mentioned in the previous section, we expect the Kohonen algorithm could be used to learn the metric relationships between the points in the reconstructed attractor and later that it could be included as a part of the MMA,

ESANN'2000 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 26-28 April 2000, D-Facto public., ISBN 2-930307-00-5, pp. 347-352

without significant loss on the accuracy of predictions. Therefore, the training algorithm and the prediction phase will be described in the next subsections.

3.1. The training process.

The well-known Kohonen algorithm was introduced by Teuvo Kohonen in 1982 as a model of the self-organization of neural connections [6]. We will give a brief overview of this artificial neural network and its main features, including an added statement. The interested reader could find more comprehensive descriptions in [2, 6].

The network has n units distributed in a one or two-dimensional array. There exists a neighborhood function Λ defined on IxI (I being the set of units) that depends only on the distance between two units of I ($\Lambda(i, j)$ decreases with increasing distance between i and j).

The input space Ω is a subset of $\Re^d$ endowed with a distance (in this paper the Euclidean distance is used).

The unit i, represented by the weight vector:

$$X_i = (X_{i_1}, X_{i_2}, \ldots, X_{i_d}). \tag{2}$$

is fully connected to the d inputs. A vector NDX_i, that contains the references to the points in the attractor which are nearest to neuron i, was also included.

Then, the state of the network at time t is completely defined by:

$$X(t) = (X_1(t), X_2(t), \ldots, X_n(t), NDX_1(t), NDX_2(t), \ldots, NDX_n(t)) \tag{3}$$

For a given state X, the network response to input v is the best matching unit (BMU) i_0, which is the neuron whose weight vector X_{i_0} is the closest to input v.

This network implements what is called a "topology preserving map" in the sense that, as far as possible, neighbors in the input space are mapped onto neighboring neurons [2, 6]. This property of the network is the most important in our attempt to use it as an ordered representation of the attractor generated by the time series we want to predict.

In most real applications only one variable can be measured from the chaotic dynamical system, a scalar time series that can be designated by $y(t)(t = 1, 2, \ldots, N)$. The attractor can be reconstructed from the time series using the method of time delayed vectors proposed by Takens [8] in the following form:

$$v(t) = (y(t), y(t-\tau), \ldots, y(t-(d_e-1)\tau)) \tag{4}$$

where d_e is the embedding dimension of the reconstructed attractor and τ is the time delay.

Now, the only statement that needs to be added to the training algorithm of the Kohonen network is the following:

⇒ Update the vectors $NDX_i(t+1), \forall i \in I$ with the indexes of the M nearest points (in the attractor) to $X_i(t+1)$.

ESANN'2000 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 26-28 April 2000, D-Facto public., ISBN 2-930307-00-5, pp. 347-352

The Kohonen network can be trained by taking the points in the reconstructed attractor as the inputs (we used the points from the first half of the time series values). Once the learning process has ended, the network can be incorporated as a part of the prediction algorithm.

3.2. The prediction phase

The idea is to substitute the search for the nearest neighbors of the point v_T by the BMU i_0 (taking v_T as the input to the network) and its closest neighbors (as indicated by the references in the vector NDX_{i_0}). Then, these points are taken as the v_{k_i}. After that, the prediction is found as in the MMA.

The prediction algorithm can be explicitly written as:

Let v_T be the point on the attractor whose future evolution will be predicted.

1. Present the input v_T to the network (previously trained with the points of the reconstructed attractor) and choose the BMU i_0.

2. Find the v_{k_i} in the original attractor using the indexes in the vector NDX_{i_0} of the winning unit i_0:

$$v_{k_i} = v(NDX_{i_0}(j)), j = 1, \ldots, M \tag{5}$$

3. Transform the v_{k_i} to $v_{k_{i+p}}$ (p is the prediction step) and obtain the prediction $\hat{v}_{T+p}$ using equation (1) as in the MMA.

4. Application of the prediction algorithm to computer generated time series

In order to evaluate the performance of the proposed algorithm, it was applied to the prediction of time series generated by the Hénon and the Ikeda Maps (two well known chaotic dynamical systems) and the results were compared with those obtained using the MMA with the same time series. The generated time series were corrupted by superimposed observational noise [5].

For the systems previously mentioned, time series of different lengths N(128, 256, 512, 1024, 2048) were generated, having the first 10000 points discarded, because considered as transient. After that, the attractors were reconstructed using the method of delays [8] with delay time $\tau = 1$, prediction step $p = 1$ and embedding dimension (reconstruction dimension) $d_e = 3$ in all cases.

Next, different Kohonen networks were trained with the points of the attractors and the prediction algorithm was applied to predict the rest of the points of each time series not used during training. It is important to mention that the trained networks had a number of neurons that varied depending on the size of the attractor, ranging from networks with 7x7 neurons to networks with 33x33 neurons.

ESANN'2000 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 26-28 April 2000, D-Facto public., ISBN 2-930307-00-5, pp. 347-352

The proposed prediction algorithm was implemented using the SOM Toolbox developed by the Neural Networks Research Centre of the Helsinki University of Technology and the experiments were carried out using a SUN Sparc Ultra10 with MatLab 5.2.1. To quantify the performance of the algorithm, the Relative Root Mean Square Error (RRMSE) between the real and predicted time series, was computed [5].

The results of the application of the proposed prediction method are shown in figure 1. In all cases the plotted values are the averaged values between the five time series generated from each system. As it can be seen from the figure, the values of the RRMSE clearly demonstrate the ability of the proposed algorithm to make successful predictions of chaotic time series.

As it can be expected (considering that the proposed method is only a modification of the MMA) the RRMSE has a similar behavior for both prediction algorithms, but with a slightly better performance of the proposed method. It is important to notice that our aim with this work was to propose a modification of the MMA that could take advantage of the high degree of parallelism and the possibility of a VLSI implementation of the Kohonen self-organizing map, to improve the prediction speed of the MMA without significant loss in the accuracy of predictions. The experimental results have shown that this objective was accomplished.

5. Conclusions

In this paper the use of a Kohonen network as part of a deterministic nonlinear prediction method has been proposed and the short term predictions of different computer generated time series were analyzed.

The experimental results have shown that the Kohonen network can effectively learn the neighborhood relations present in the reconstructed attractor of the time series, and that the proposed algorithm can achieve good prediction performance. The MMA can thus be implemented using a Kohonen network as one of its components. The benefits of the high parallelism of this neural network, can thus be obtained, without loss on the accuracy of predictions.

As a future work we are interested in the use of the proposed algorithm to develop a test to distinguish between chaos and noise. The performance of the algorithm with dynamic selection of the number of neighbors can also be studied.

References

[1] G. E. P. Box and G. M. Jenkins. *Time Series Analysis: Forecasting and Control*. San Francisco, CA: Holden-Day, 1976.

[2] M. Cottrell, J.C. Fort, and G. Pages. Theoretical aspects of the som algorithm. *Neurocomputing*, 21:119–138, 1998.

ESANN'2000 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 26-28 April 2000, D-Facto public., ISBN 2-930307-00-5, pp. 347-352

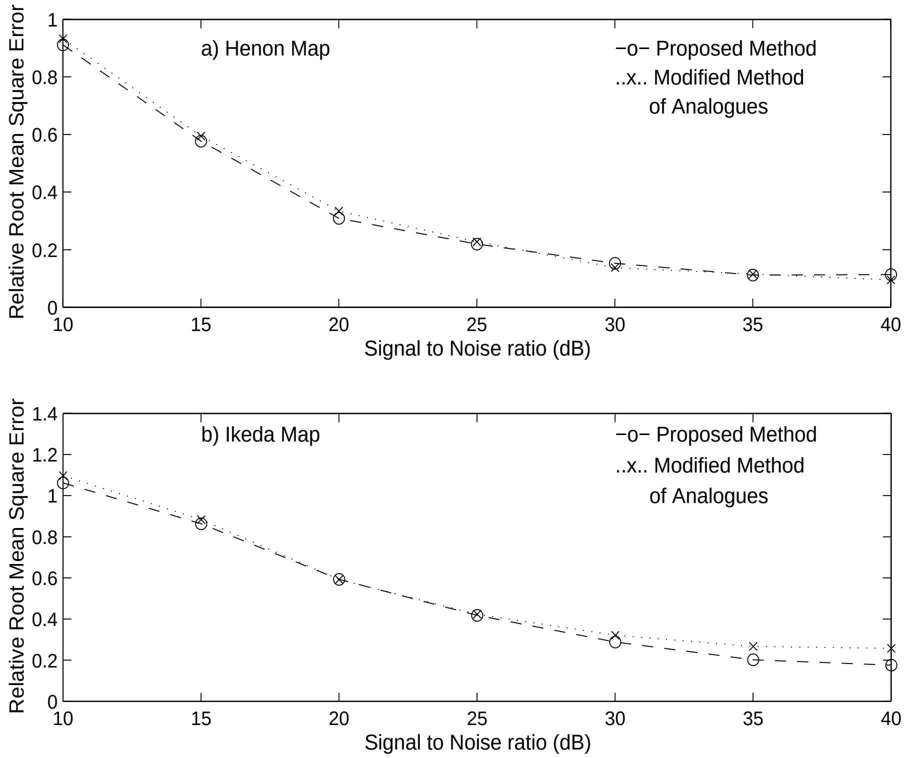

Figure 1: Averaged values of the RRMSE for the time series generated from the Hénon Map (a) and the Ikeda Map (b).

[3] M. Cottrell, B. Girard, Y. Girard, M. Mangeas, and C. Muller. Neural modelling for time series: Statistical stepwise method for weight elimination. *IEEE Trans. Neural Networks*, 6(6), November 1995.

[4] J. D. Farmer and J. J. Sidorowich. Predicting chaotic time series. *Physical Review Letters*, 59(8):845–848, 1987.

[5] T. Ikeguchi and K. Aihara. Prediction of chaotic time series with noise. *IEICE Trans. Fundamentals*, E78-A(10):1291–1298, October 1995.

[6] T. Kohonen. Analysis of a simple self-organizing process. *Biol. Cybern*, 44:135–140, 1982.

[7] E. N. Lorenz. Atomospheric predictibility as revealed by naturally occuring analogues. *Journal of the Atomospheric Sciences*, 26:636–646, 1969.

[8] F. Takens. *Detecting strange attractors in turbulence*, volume 898 of *eds., D.A. Rand and B.S. Young, Lecture Notes in Mathematics*. Springer-Verlag, Berlin, 1981.