Forecasting chaos from small data sets: a comparison of different nonlinear algorithms

J. Phys. A: Math. Gen. 28 (1995) 3381–3388. Printed in the UK

Forecasting chaos from small data sets: a comparison of different nonlinear algorithms

H D Navone and H A Ceccatto

Instituto de Física Rosario, Consejo Nacional de Investigaciones Científicas y Técnicas, and Universidad Nacional de Rosario, Bvd. 27 de Febrero 210 Bis, 2000 Rosario, Argentina

Received 19 December 1994

Abstract. We compare the capabilities of different nonlinear algorithms for forecasting chaotic time series when a limited number of past values of the series is available, a situation most often found in real-world problems. In particular, we consider instance-based methods and neural network techniques, which are frequently advocated in the literature as universal, simple, and fairly reliable algorithms for time-series analysis. Furthermore, we propose a linear correction to the instance-based Wimplex method that produces remarkably good results on clean data. Finally, we present a preliminary application of the ideas discussed here to the real-world series of solar activity, which is often taken as a benchmark for these kinds of studies.

Real-world phenomena generally have complex dynamics, very often masked by noise coming from measurement errors and/or unexpected perturbations which couple temporarily with the system. Moreover, in many cases intrinsic nonlinearities lead to chaotic behaviour, which makes long-term predictions impossible [1]. Under these conditions, developing first-principle models can be very difficult or even infeasible. In such cases, one alternative approach for short-term forecasting is to build models directly from the available data. The observed past values of the dynamical variables are thus organized as a time series, and the intrinsic dynamics reconstructed from this historical data set. Several nonlinear methods have been developed for this purpose in recent years, some of them with high levels of sophistication and effectiveness (for short reviews see [2, 3]). However, most methods are successful only when a large record of observations is available.

In this paper we will compare the performances of different nonlinear algorithms for chaotic time-series forecasting when the available data are relatively scarce. In particular, we will consider two methods which have been advocated as universal, simple, and fairly reliable. The first one is a global attempt to fit the underlying attractor by means of the computational paradigm known as artificial neural networks [2, 4]. The second one—known as Wimplex [5]—is local in character and belongs to the so-called instance-based methods, which make use of similar situations to the current state in the history of the system to predict its future behaviour. These two methods have been selected because, as mentioned above, (i) they can be applied to almost any problem at hand, (ii) they are simple to program and make no extensive use of computational resources (this is not true for large neural networks), and (iii) they have been proved to be reliable and even superior to other techniques [6, 7]. Moreover, in general they do not require a large record of series iterates, since the number of internal parameters can be kept to a minimum (this is not the case, for instance, with methods based on polynomial representations of the unknown dynamical laws). A preliminary comparison of these methods have been presented elsewhere [8]. We will also consider the introduction of linear corrections to the Wimplex algorithm, in a scheme that we will term local hyperplane approximation (LHA). According to this scheme, the Wimplex predictions are corrected by fitting an optimal hyperplane in the neighbourhood of the current state. Similar ideas concerning local fittings of the attractor have been discussed a number of times in the literature [9–11], but, to our knowledge, this particular implementation in connection with the Wimplex algorithm is new and will be shown to lead to a large improvement in forecast performance.

The examples on which we will discuss these ideas are the chaotic time series corresponding to the logistic map

$$x_{t+1} = r x_t (1 - x_t) \tag{1}$$

with $r = 3.8$, and the Lorenz system

$$\frac{\mathrm{d}x}{\mathrm{d}t} = -ax + ay \qquad \frac{\mathrm{d}y}{\mathrm{d}t} = -xz + bx - y \qquad \frac{\mathrm{d}x}{\mathrm{d}t} = xy - cz \tag{2}$$

with $a = 16.0$ $b = 120.1$ and $c = 4.0$. The choice of constants is the same as in [2]. These two examples have been chosen in order to compare the different methods' performances on a series with negligible autocorrelation (logistic map), and others with significant autocorrelation (Lorenz equations)†. These series are thus clean, computer generated data, where the methods can be checked without the complications intrinsic to actual observations of a system. In the last part of this paper we will also present results for real-world data corresponding to the solar activity (Wolf number).

The task we face is the reconstruction of the intrinsic dynamics of a time-series

$$x_t = F(\boldsymbol{X}_t) \tag{3}$$

where $\boldsymbol{X}_t = (x_{t-\tau}, x_{t-2\tau}, \dots, x_{t-d\tau})$ is a vector in the τ-lag space where the reconstruction takes place [12] (in the following we will take $\tau = 1$). A dimension d of this embedding space equal to $2D + 1$, where D is the fractal dimension of the underlying attractor, is in principle sufficient for proper reconstruction of the dynamics [13]. Both D and the unknown function F must be obtained from the historical data set that contains the past values of the observable x. The simplest method to tackle this problem is to assume that x_t is a random variable of the previous d values, and use a linear autoregressive model. This corresponds to replacing the actual surface in lag space by a hyperplane. This linear method is in general not satisfactory for chaotic series [9], although it can have a limited use for time series with large autocorrelation.

Other common approaches, nonlinear in character, are instance-based methods like Sugihara and May's Simplex [14]. We will consider here a related, simpler version called Wimplex (W(d, n)) [5]. In this method one looks for the n vectors $\boldsymbol{X}_i$ $(i = 1, n)$ in the record closest to the actual one, $\boldsymbol{X}_t$, and predicts

$$x_t = \overline{x_i} \equiv \sum_{i=1}^{n} x_i P(\boldsymbol{X}_t - \boldsymbol{X}_i)$$

where $x_i = F(\boldsymbol{X}_i)$. The weight function is usually taken to be $P(\boldsymbol{X}_t - \boldsymbol{X}_i) \propto 1/d(\boldsymbol{X}_t - \boldsymbol{X}_i)$, where $d(\boldsymbol{X}_t - \boldsymbol{X}_i)$ is the (Euclidean) distance between $\boldsymbol{X}_t$ and $\boldsymbol{X}_i$.

† We have integrated the Lorenz system using a fourth-order Runge–Kutta method with an arbitrary time step $\Delta t = 0.03$. This generates iterates with an autocorrelation which decays to zero approximately after five time steps.

Yet another method of reconstructing chaotic time series makes use of artificial neural networks [2, 4]. These parallel computational architectures are highly-interconnected simple processors (called neurons), which simulate to some extent the structure and functioning of the brain [15]. In particular, the so-called 'feed-forward' neural networks have a group of neurons (the input layer), which are fed by the external stimuli. These neurons pass the inputs they receive to intermediate units, which form the hidden layer. Neurons in the hidden layer process the information, and send their results to the units in the output layer, which finally produce the response to the external stimuli. The interconnection strength among neurons (weights) are chosen so that the network 'learns' to relate inputs to desired outputs. In this paper we will use feed-forward neural networks with d neurons in their input layer, n neurons in the hidden layer, and a single neuron in the output layer (NN(d, n)). They will be trained to learn the unknown function F in (3) from the examples in the data set. Details of this method can be found in [2, 4].

In what follows we will discuss a linear correction to the Wimplex algorithm which we call the local hyperplane approximation (LHA(d)). To implement the Wimplex, given the actual state $\boldsymbol{X}_t$ one must find the closest $\boldsymbol{X}_i$ $(i = 1, n)$ vectors in the historical data set. If the points in this data set cover the whole attractor reasonably well, the distances $d(\boldsymbol{X}_t - \boldsymbol{X}_i)$ will be small enough to approximate

$$x_i = F(\boldsymbol{X}_i) \cong F(\boldsymbol{X}_t) + \boldsymbol{A}_t \cdot (\boldsymbol{X}_i - \boldsymbol{X}_t) \tag{4}$$

where $\boldsymbol{A}_t = \nabla F(\boldsymbol{X}_t)$. We can now multiply these equations by some prescribed weight *function $P(\boldsymbol{X}_t - \boldsymbol{X}_i)$, and sum up all of them to obtain the prediction*

$$x_t = F(\boldsymbol{X}_t) \cong \overline{x_i} + \boldsymbol{A}_t \cdot (\boldsymbol{X}_t - \overline{\boldsymbol{X}_i}). \tag{5}$$

In order to find the d components of $\boldsymbol{A}_t$ one eliminates the nth equation in (4) and chooses $n - 1 = d$, which reduces the problem to that of solving a squared linear system. Thus, the whole strategy leading to (5) only amounts to predicting x_t as the value corresponding to $\boldsymbol{X}_t$ on the optimal local plane which contains the Wimplex point $(\overline{\boldsymbol{X}_i}, \overline{x_i})$.

In practice one usually faces the problem that the data are scarce and do not cover the whole attractor well, so that the approximation (4) is not always valid. In such cases the linear corrections are sometimes too large, and it is preferable to keep the Wimplex predictions. We have set an upper bound

$$\left|\boldsymbol{A}_t \cdot (\boldsymbol{X}_t - \overline{\boldsymbol{X}_i})\right| \leqslant \alpha x_{max} \tag{6}$$

to discriminate in which cases linear corrections are to be retained. This bound is expressed as a fraction α of the largest iterate $x_{\max}$ in the series (the value of the parameter α is chosen as explained below in the examples).

In order to compare the performances of the above described methods in forecasting the chaotic time series (1) and (2) we have performed the following steps. For a given series, we have first generated $2N$ iterates. Then, we have split this record in a data set consisting of the first N iterates and a validation set containing the second half of points. Using the information in the data set, we have performed m-step forecasts of the iterates in the validation set†. Finally, we have appraised the forecasting abilities of the different methods by means of the average relative variance,

$$\text{ARV}(m, S) = \frac{\sum_{i \in S} (x_i - f_i(m))^2}{\sum_{i \in S} (x_i - \overline{x})^2}.$$

† We call a m-step forecast the prediction of x_t using as input the observed $\boldsymbol{X}_{t-m}$, which requires performing m succesive predictions. Notice, however, that the record used to make the forecasts is always restricted to the N first iterates; the values x_i with $i > N$ in $\boldsymbol{X}_{t-m}$ are taken only to specify the initial state.

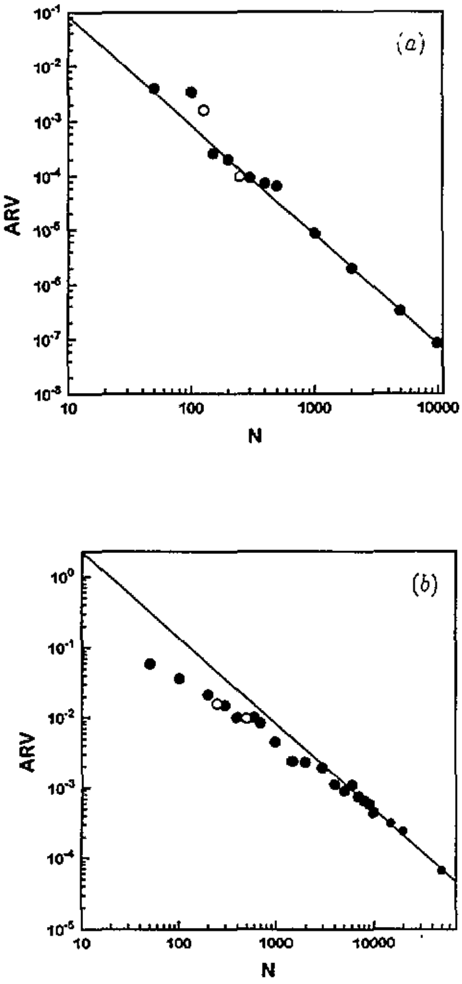

Figure 1. Relative single-step prediction error ARV as a function of the length N of the historical data set for the Wimplex method w(d, n). (a) Logistic map with $d = 1, n = 2$; (b) Lorenz system with $d = 4, n = 1$. The open circles correspond to the values of N used in this paper.

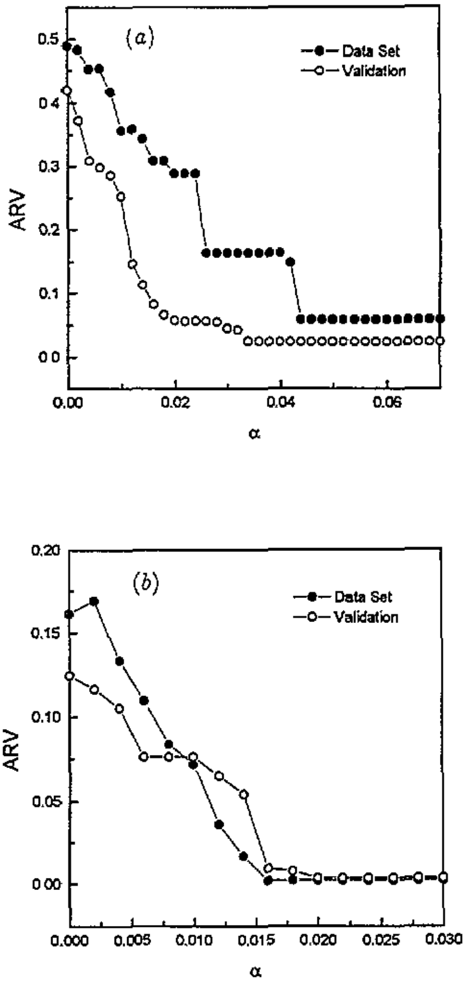

Figure 2. Relative eight-step prediction error ARV for the logistic map, as a function of the maximum allowed size of linear corrections (see equation (6)). (a) $N = 125$ iterates in the historical data set; (b) $N = 250$ iterates in the historical data set.

In this expression S can alternately be the data ($i = 1, N$) or validation ($i = N + 1, 2N$) set, $f_i(m)$ is the m-step forecast of x_i, and $\bar{x}$ is the mean value of x on S. A perfect forecast corresponds to ARV = 0, while a constant prediction equal to the series average gives ARV = 1. Once these results had been obtained, we repeated all the calculations starting with a larger initial record of $4N$ iterates.

Since, as stated above, we are interested in comparing forecasting abilities when the available data are scarce, it is necessary to work out how to choose the length N of the historical record in such situation. In the case where one has a large number of iterates, local fittings of the attractor produce forecasting errors ARV with a nice scaling behaviour $\log \text{ARV} \sim \log N$ [9]. In figure 1 we show this behaviour for both the logistic map and the

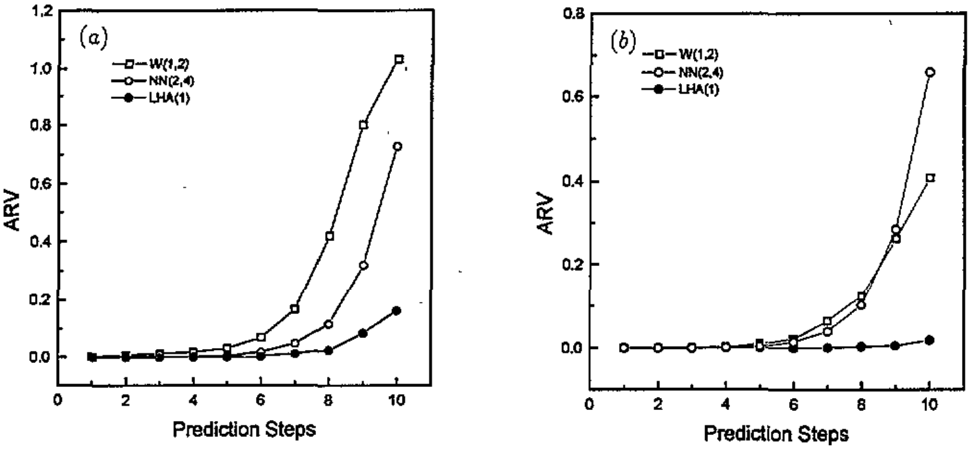

Figure 3. Relative multi-step prediction error ARV in the validation interval for the logistic map, as a function of the prediction steps. (*a*) $N = 125$ iterates in the historical data set; (*b*) $N = 250$ iterates in the historical data set.

Lorenz system, using the Wimplex algorithm as the prediction method. For the study below we have chosen values of N for which the error departs from this scaling law and starts having a sample-dependent value, which indicates that the points do not cover the whole attractor uniformly.

Let us discuss the results for the logistic map first. For this series we took $N = 125$, so we generated 500 iterates from (1) and considered alternately historical data sets consisting of the first 125 and 250 iterates, respectively. In order to choose the parameter α in (6) we proceeded as follows. Both the data and validation sets were forecast for different values of α, and the corresponding ARV plotted as in the typical curves shown in figure 2. As can be seen from this figure, the errors always decrease with α, so that, even for data sets as small as the ones being considered, for the logistic map it is safe to keep all the linear corrections (this is related to the low fractal dimension of its attractor). However, the same will no longer be true for the Lorenz system, as discussed below. Notice that the size of the linear corrections is smaller than 4% (2%) of $x_{\max}$ for the 125- (250)-iterate record. Quite remarkably, their introduction reduces the error by approximately one order of magnitude with respect to the standard Wimplex algorithm (ARV value at $\alpha = 0$).

In figure 3 we present the best results obtained with the methods discussed above: W(d, n), NN(d, n) and LHA(d). To obtain these results we have performed an extensive search, playing with the available parameters d and n in every case. As can be seen from this figure, the LHA clearly outperforms the other two.

Figures 4 and 5 show the results of a similar investigation for the Lorenz system (2). Here we generated 1000 iterates ($N = 250$) and took alternately data sets of 250 and 500 points. Because of the Lorenz attractor structure, the data are relatively scarcer in this case (figure 1). As a consequence, there is an optimal size of allowed linear corrections which depends on both the number of data points and forecast steps into the future (figure 4). A near optimal value of α can be chosen by finding the minimum error in the historical record. For this in-sample prediction, it is better to disregard points directly correlated to the one being predicted, which avoids taking unfair advantage of the series autocorrelation. Following this procedure, the LHA results again show this to be the best forecasting method.

Up to this point we have considered only clean data. As a preliminary application of

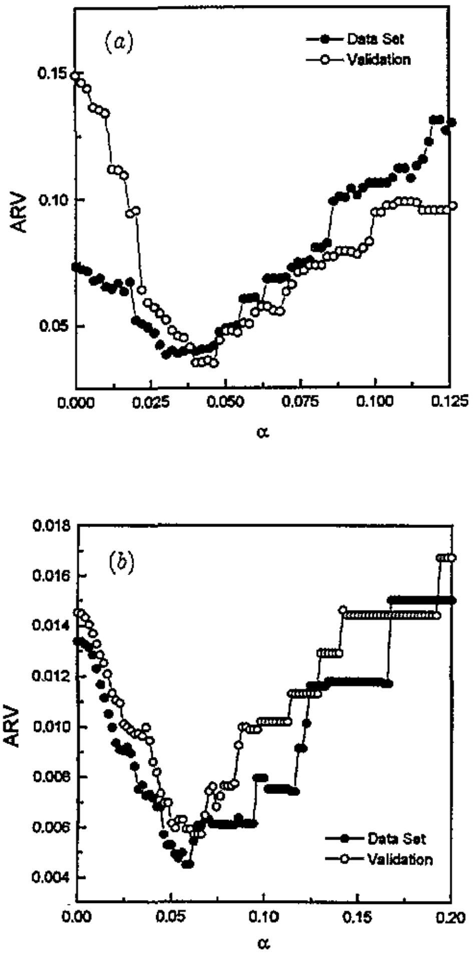

Figure 4. Relative multi-step prediction error ARV for the Lorenz system, as a function of the maximum allowed size of linear corrections (see equation (6)). (*a*) Five-step predictions and $N = 250$ iterates in the historical data set; (*b*) Two-step predictions and $N = 500$ iterates in the historical data set.

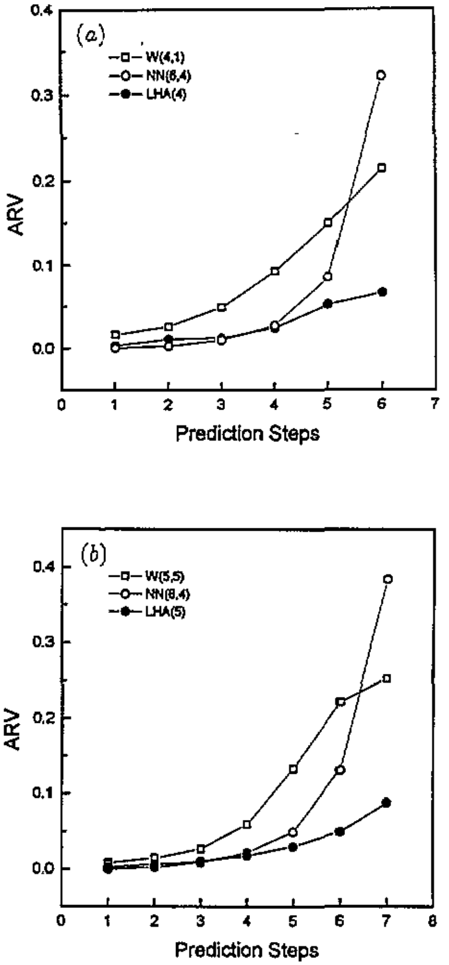

Figure 5. Relative multi-step prediction error ARV in the validation interval for the Lorenz system, as a function of the prediction steps. (*a*) $N = 250$ iterates in the historical data set; (*b*) $N = 500$ iterates in the historical data set.

these methods to real-world data, we studied the time series of solar activity as measured by the annual mean sunspot number (Wolf number). There is a fairly long record of this series, which starts in the year 1700. In order to compare with previous studies in the literature [4, 6] we will split this record into two intervals, 1700–1920 and 1921–1955, which correspond to the data and validation sets respectively. Because of the important amount of noise in sunspot numbers, in this case the linear corrections produce only a slight improvement of ~5–10% over the naked Wimplex. However, figure 6 shows that this method compares favourably against the usual benchmark [16], and its predictions are close to the best results in the literature [4]. Moreover, for the interval 1956–1979 a single-step prediction using the W(4, 5) model gives ARV = 0.25, which should be compared with

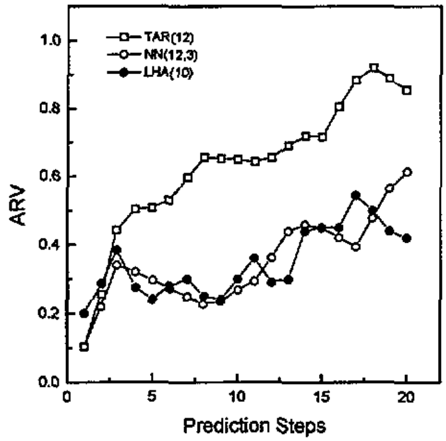

Figure 6. Relative multi-step prediction error ARV for the solar-activity time series in the validation interval 1921–1955. The values of α are chosen from the in-sample prediction of the historical data set 1700–1920. The results for TAR and NN models were taken from [4].

$\mathrm{ARV}_{\mathrm{TAR}} = 0.28$ [16] and $\mathrm{ARV}_{\mathrm{NN}} = 0.35$ [4]. In passing we note that since the forecasting performance in figure 6 does not deteriorate appreciably over a long period of time, the solar-activity dynamics might be masked only by noise instead of chaotic behaviour [17].

In this paper we have proposed the LHA as a simple nonlinear algorithm for chaotic time-series forecasting. Moreover, we have shown that it is very efficient even for small data sets, outperforming more involved methods like neural networks. This conclusion is apparent from our results on clean data. However, in addition to being scarce, real data are most often plagued by noise. As expected, the forecasting capabilities of the LHA are affected by the presence of large amounts of noise in the data. We are currently investigating, in a controlled fashion, the effects of noise on this algorithm, which is crucial in determining its capabilities for real-world applications.

Acknowledgment

HAC is grateful to Fundación Antorchas for partial financial support.

References

[1] Schuster H-G 1988 *Deterministic Chaos* (VCH)
[2] Elsner J B and Tsonis A A 1992 *Bull. Amer. Meteor. Soc.* **73** 49 (Corrigendum 1993 **74** 243)
[3] West B J and Mackey H J 1992 *J. Sci. Ind. Res.* **51** 634
[4] Weigend A S, Huberman B A and Rumelhart D E 1990 *Int. J. Neural Syst.* **1** 193
[5] Fowler A D and Roach D E 1993 *Comput. Geosci.* **19** 41
[6] Calvo R A, Ceccatto H A and Piacentini R D 1995 *Astrophys. J.* in press
[7] Navone H D and Ceccatto H A 1994 *Climate Dynamics* **10** 305
[8] Navone H D and Ceccatto H A 1995 *Chaos, Solitons and Fractals* in press
[9] Farmer J D and Sidorowich J J 1987 *Phys. Rev. Lett.* **59** 845
[10] Martinetz T M, Berkovich S G and Schulten K J 1993 *IEEE Trans. Neural Networks* **4** 558
[11] Hsu W, Hsu L S and Tenorio M F 1993 *Int. J. Neural Systems* **4** 247
[12] West B J 1985 *An Essay on the Importance of Being Nonlinear* (Berlin: Springer)
[13] Takens F 1981 *Lecture Notes in Mathematics* vol 898 ed D A Rand and L S Young (Berlin: Springer)
[14] Sugihara G and May R M 1990 *Nature* **344** 734

[15] Rumelhart D E, McClelland J L and the PDP Research Group 1986 *Parallel Distributed Processing* (Cambridge, MA: MIT Press)

[16] Tong H and Lim K S 1980 *J. R. Stat. Soc.* B **42** 245

[17] Price C P, Prichard D and Hogenson E A 1992 *J. Geophys. Res.* **97** 19, 113