Statistics and Computing **11**: 241–255, 2001
© 2001 Kluwer Academic Publishers. *Manufactured in The Netherlands.*

A resampling algorithm for chaotic time series

SILVIA GOLIA* and MARCO SANDRI†

**Dipartimento Metodi Quantitivi, Università di Brescia, C. da S, Chiarra 48/b, 25422 Brescia, Italy*
golia@eco.unies.it
†*Centro di Informatica e Calcolo Automatico, Università di Verona, Via dell' Artigliere 19, 37129 Verona, Italy*
marco.sandri@univr.it

Received April 1998 and accepted May 1999

In the field of chaotic time series analysis, there is a lack of a distributional theory for the main quantities used to characterize the underlying data generating process (DGP). In this paper a method for resampling time series generated by a chaotic dynamical system is proposed. The basic idea is to develop an algorithm for building trajectories which lie on the same attractor of the true DGP, that is with the same dynamical and geometrical properties of the original data. We performed some numerical experiments on some short noise-free and high-noise series confirming that we are able to correctly reproduce the distribution of the largest finite-time Lyapunov exponent and of the correlation dimension.

Keywords: resampling, chaotic time series, shadowing property, neural networks

1. Introduction

A typical problem in statistical data analysis is the estimation of an unknown quantity $\boldsymbol{\theta}$ from a given sample. For time series, $\boldsymbol{\theta}$ could be for instance the mean, the variance or the autocorrelation at a specified lag. For chaotic time series, it could be the average rate of divergence of nearby trajectories, the dimension of the attractor or the entropy of the data generating process (DGP).

Two main questions that immediately follow are: what estimator $\hat{\boldsymbol{\theta}}$ for $\boldsymbol{\theta}$ has to be used; and, after having chosen a specific $\hat{\boldsymbol{\theta}}$, how accurate it is. In other words, together with the choice of a 'good' estimator, it is generally of fundamental importance to be able to give a measure of its accuracy. There are two ways to accomplish this task: deriving formulas via analytical argument (often difficult or too complicated) based on assumptions (often unrealistic and unverifiable), or making use of resampling techniques.

Following Simon and Bruce (1993), by resampling procedures we mean "*...a collection of computer intensive methods to use experimental data to obtain different estimates of a statistics or underlying distribution*", with (possibly) minimal assumptions. Roughly speaking, they are computer-based methods, which substitutes considerable amounts of computation in place of theoretical analysis for assigning measures of accuracy to statistical estimates. Jackknife and bootstrap are the most known and widely used resampling methods (Efron and Tibshirani 1993).

In the field of chaotic time series analysis the inferential component is often completely absent. Many works (see Geist *et al.* (1990), Cutler (1993), Kurths *et al.* (1995) and the references therein) focus their attention on the estimation of quantities which characterize chaotic systems from a dynamical or geometrical point of view (the Lyapunov spectrum, the fractal dimension and some complexity and entropy measures) but none of them, with few exceptions, are worried about the necessity of assessing a significance level for the estimated quantities.

Consequently, these estimations are of little or no use for practical purposes. For instance in the case of the largest Lyapunov exponent λ_1, a strictly positive value is, from a theoretical point of view, a strong indication of the presence of a chaotic DGP, but, from a practical perspective, without a confidence region or a test of hypothesis, it does not tell us whether this is due to an accidental cause or whether it correctly reflects one of the properties of the dynamical system. Lai and Chen (1995), Gencay (1996), Bailey *et al.* (1997), Golia and Sandri (1997)

0960-3174 © 2001 Kluwer Academic Publishers

and Ziehmann *et al.* (1998) represent the first contributes in this area. Starting from an idea contained in the works of Farmer and Sidorowich (1988) and Casdagli (1992), the present work tries to give an answer to this problem, proposing a deterministic resampling algorithm (DRA) for chaotic time series, based on local neural approximations of the DGP.

One important remark is necessary. We warn the reader against classifying our algorithm as a *surrogate data method* (Theiler *et al.* 1992). In fact the aim of the latter technique is to test the null hypothesis that the underlying DGP is of a specified type, for example white noise. For this purpose, the method starts generating surrogate data, that is a set of time series which are similar to the observed data but consistent with the null hypothesis. Then it computes a discriminant statistic for the original and the surrogate data sets which allows one to decide if the null hypothesis has to be accepted or not. Our aim here is completely different. We do not consider this kind of hypothesis testing. Producing an arbitrary number of resampled series which mimic the deterministic behavior of the underlying DGP, we want to supply density estimations for the estimated statistics.

The paper is organized as follows. In Section 2 we present some basic definitions which will be used in the next sections. After a short review of the main methods for time series bootstrapping and after considering their inability to resample chaotic trajectories, Section 3 presents the main ideas upon which the proposed DRA is founded. A detailed description of the algorithm is given in Section 4. In Section 5, we discuss under what conditions the randomness shown by the bootstrap estimators is a meaningful approximation of the 'real' randomness which characterize the estimates of the parameters of interest. In Section 6, we test our deterministic resampling algorithm on time series yielded by the Lorenz model with and without additive noise, and on the geophysical Southern Oscillation Index. Conclusions follow in Section 7. Descriptions of neural networks and the time delay method are briefly given in the Appendix.

2. Some basic notions

An m-dimensional differentiable dynamical system is a time evolution defined by an evolution equation (continuous-time case)

$$\dot{\mathbf{x}} = F(\mathbf{x}), \quad \mathbf{x} \in \mathbf{R}^m,$$

which yields a smooth function $f^t(\mathbf{x})$, the *flow*, such that $\frac{d}{dt}(f^t(\mathbf{x}))_{t=\tau} = F(f^\tau(\mathbf{x}))$ for all $\tau \in \mathbf{R}$, or by a *map* (discrete-time case)

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t), \quad \mathbf{x} \in \mathbf{R}^m,$$

where f or F are differentiable functions and $t \in \mathbf{N}$.

The point x is a *periodic point* of period n if $f^n(x) = x$.

We say $f : D \to D$ is *topologically transitive* if for any open sets $U, V \subset D$ there is a n such that $f^n(U) \cap V \neq \emptyset$.

$f : D \to D$ displays *sensitive dependence on initial conditions* (SDIC) if there exists $\delta > 0$ such that for any x in D and any neighborhood V of x, there exists a y in V and $n \geq 0$ such that d $(f^n(x), f^n(y)) > \delta$, where d is a distance defined on D. It means that orbits with close initial conditions, will diverge after a small number of steps.

If $V \subseteq D$ is a set, then $f : V \to V$ is *chaotic* on V if (1) f has SDIC (2) f is topologically transitive and (3) periodic points are dense in V (Berliner 1992). The quantities most frequently used to characterize the dynamics of a chaotic system are the correlation dimension and the Lyapunov exponents. They measure the degree of geometrical complexity of the attractor and the degree of dynamical instability of the trajectories respectively.

The *correlation dimension*, d_{corr}, is defined by

$$d_{\text{corr}} = \lim_{\varepsilon \to 0} \frac{\log C(\varepsilon)}{\log \varepsilon} \tag{1}$$

where $C(\varepsilon)$ is

$$C(\varepsilon) = \lim_{N \to \infty} \frac{1}{N^2} C_N(\varepsilon)$$

where N is the number of available points and $C_N(\varepsilon)$ is the number of pairs of points on the attractor whose distance from one another is less that ε. The correlation dimension depends on the spatial correlation of points on the attractor and thus on the degree of its inhomogeneity. Moreover, it gives information on the minimum number of variables present in the evolution of the corresponding dynamical system (Grassberger and Procaccia 1983).

The Lyapunov exponents λ_i are measures of the average rate of divergence or convergence of typical trajectories in time. If the initial state of a time evolution is perturbed, even only slightly, the exponential rate at which the perturbation increases, or decreases, with time is called the Lyapunov exponent. The largest Lyapunov exponent is given by

$$\lambda(\mathbf{x}, \mathbf{w}) = \lim_{t \to \infty} \frac{1}{t} \ln \; \|D_{\mathbf{x}} f^t(\mathbf{x}) \cdot \mathbf{w}\| \tag{2}$$

for almost any vector $\mathbf{w}$, where $D_{\mathbf{x}} f^t(\mathbf{x})$ is the derivative of f^t in $\mathbf{x}$ (Benettin *et al.* 1980, Eckmann and Ruelle 1985). Henceforth we will use λ_1 instead of $\lambda(\mathbf{x}, \mathbf{w})$. The multiplicative ergodic theorem of Oseledec (1968) ensures that (2) exists and does not depend on the specific initial condition chosen. The other exponents can in principle be obtained by diagonalizing the positive matrices $D^*_{\mathbf{x}} f^t D_{\mathbf{x}} f^t$, where $D^*_{\mathbf{x}} f^t$ is the adjoint of $D_{\mathbf{x}} f^t$, and using the fact that their eigenvalues behave like $e^{2t\lambda_1}$, $e^{2t\lambda_2}, \ldots$. In terms of information theory the largest Lyapunov exponent λ may be interpreted as giving the rate of loss of information about the location of the initial point. If the largest Lyapunov exponent is positive, then this is one of the main signatures of chaos. This notion can also be extended to stochastic dynamical system (see Tong 1995).

Many methods were proposed in literature to estimate d_{corr} and λ_1 (see Geist *et al.* 1990 and the references therein, Brown *et al.* 1991, McCaffrey *et al.* 1992). In this paper we will use the

Grassberger and Procaccia (1983)'s estimator for d_{corr} and the Rosenstein *et al.* (1993)'s estimator for λ_1.

Using a finite-precision computer in order to numerically calculate the orbits of a chaotic dynamical system, it is inevitable to introduce at any step of the computation small truncations or rounding errors which will be greatly magnified by SDIC in the future evolution of the system. This leads to a fundamental paradox: the true orbit starting from an initial condition $\mathbf{x}_0$ can be expected to have no correlation with the numerical orbit starting from the same $\mathbf{x}_0$ after few steps. Moreover, it has been shown that numerical experiments can sometimes yield dynamics which are qualitatively completely different from the true ones. For instance, Corless *et al.* (1991) show that the implicit Euler method can, for large enough step size, artificially stabilize truly unstable fixed points and completely destroy any possible chaotic attractors. Conversely, Lorenz (1989) gives examples of chaotic behavior which occurs when difference equations, used as approximations to ODEs, are solved numerically with an 'excessively' large time step. Corless *et al.* (1991) also show that floating-point simulation of a discrete map with a globally attracting fixed point at the origin can appear chaotic, for extremely long time, purely due to rounding error effects. An interesting review on numerical problems for chaotic dynamical systems is Corless (1994).

Therefore, a fundamental question arises: under what conditions will the computed orbit be close to a true orbit of the model? This is equivalent to ask if the dynamical system has the so-called shadowing property (or pseudo orbit tracing property, POTP).

A sequence of points $\{\mathbf{x}_t\}_{t>0}$ in $\mathbf{R}^m$ is called an *orbit* of the dynamical system f if $\mathbf{x}_{t+1} = f(\mathbf{x}_t) = f^{t+1}(\mathbf{x}_0)$, where $f^t = f \circ f \circ \cdots \circ f$ (t times). Note that an orbit, often referred to a *true orbit* in contrast with the notion of pseudo orbit, can be completely identified by the couple $\{f, \mathbf{x}_0\}$. Moreover, a piece of orbit of length N $\{\mathbf{x}_t\}_{t=0}^N$ is identified by the triplet $\{f, \mathbf{x}_0, N\}$. In fact, given an initial condition $\mathbf{x}_0$, the future evolution of the system is completely defined. Alternatively, a true orbit can be seen as a single realization (defined by $\mathbf{x}_0$) of the deterministic process f. A *δ-pseudo orbit* is a sequence of points $\{\mathbf{x}_0, \mathbf{x}_1, \ldots\}$ which satisfies $d(f(\mathbf{x}_t), \mathbf{x}_{t+1}) \leq \delta$ for every $t \geq 0$. We say that f has the *shadowing property* if for every $\varepsilon > 0$, there is a $\delta > 0$ such that every δ-pseudo orbit can be *ε-shadowed* by an actual orbit, i.e. there is an initial condition $\tilde{\mathbf{x}}_0$ such that $d(f^t(\tilde{\mathbf{x}}_0), \mathbf{x}_t) \leq \varepsilon$ for all $t \geq 0$.

To prove the existence of the shadowing property is not an easy task. The earliest result in this field is the Shadow Lemma by Anosov and Bowen, which proved that hyperbolic maps have the POTP. In the last decades, many other works on shadowing appeared in the literature. They prove the existence of the POTP for many classes of non-hyperbolic dynamical systems. Diamond *et al.* (1995) state that semi-hyperbolic systems have the shadowing property (and a stronger form of shadowing called *bi-shadowing*). Pilyugin and Plamenevskaya (1999) show that POTP is a property of generic homeomorphisms. Reinfelds (1997) extends the shadowing property to the discrete semidynamical systems generated by a continuous map in a complete metric space. Without assuming hyperbolicity, Sauer and Yorke (1991) prove that, as long as the system is 'sufficiently' hyperbolic along the finite-length computed trajectory $\{f, \mathbf{x}_0, N\}$, then $\{f, \mathbf{x}_0, N\}$ can be shadowed by a true trajectory. The proof of this theorem is constructive and particularly interesting because it shows that, under the conditions of the theorem, the iterated application of a specific refinement procedure on the original pseudo orbit, results in a sequence of refined pseudo orbits with decreasing noise level, and whose limit is a true orbit. Dawson *et al.* (1994) and Yuan and Yorke (1999) find conditions for nonshadowability. The brittleness B of a pseudo trajectory $\{\mathbf{x}_t\}_{t=0}^N$ is the ratio of shadowing distance ϵ over the magnitude of the one-step error δ, that is $B = \epsilon/\delta$. A necessary condition for shadowability is that the brittleness times the error magnitude of the pseudo trajectory is smaller than the extent of the attractor in phase space (Dawson *et al.* 1994). Many other important contributions to the field of numerical computation of chaotic orbits are collected in Kloeden and Palmer (1994).

Sometimes a numerical orbit is ϵ-shadowed by a true one only for a finite number of steps. We say that a pseudo orbit $\{\mathbf{x}_t\}_{t=0}^N$ has a *glitch* at the n-th step, $n < N$, if $\{\mathbf{x}_t\}_{t=0}^n$ can be shadowed but $\{\mathbf{x}_t\}_{t=0}^{n+1}$ can not. Consider one of the simplest examples of glitch: the one-dimensional logistic map $x_{t+1} = 4x_t(1 - x_t)$ and the δ-pseudo orbit which begins with $x_0 = 0.5$ (Fig. 1).

Let assume the following step is $x_1 = 1 + \delta$ and from then on the δ-pseudo orbit is computed without errors. Clearly $x_2 = 4(1 + \delta)(-\delta) < 0$, the pseudo orbit diverges to $-\infty$ and no ϵ-shadowing true orbit exists. Sauer and Yorke (1991) conjecture that one expects a glitch to occur on the order of every $1/\sqrt{\delta}$ steps. See also the results reported in Sauer *et al.* (1997).

In the next sections we will show that the shadowing property plays a crucial role in the resampling scheme we are going to propose. As a matter of fact, it is one of the necessary requisites to ensure that the resampled time series stay close to true orbits of the unknown dynamical system.

3. Resampling chaotic time series

A frequent practical problem that one must face in the analysis of chaotic time series is the estimation of some invariants $\boldsymbol{\theta}$ (like d_{corr} and λ_1) using, for instance, one of the estimators mentioned in the previous section. A key feature must be pointed out: pieces of trajectory $\{f, \mathbf{x}_0, N\}$ with equal length N, starting from different initial conditions lead to different values of the estimators $\boldsymbol{\theta}_f(\mathbf{x}_0, N)$ and the smaller N is, the more its variability is marked. In the case of the Lyapunov exponent, this fact is due to the presence of regions of the phase space characterized by different levels of the rate of contraction/expansion of the trajectories. In other words, the estimators assume different values due to the choice of different initial conditions $\boldsymbol{x}_0$. Thus, even if we are working in a purely deterministic context, it is correct and convenient to consider them as random variables with their own probability distributions (Nychka *et al.* 1992, Berliner 1992).

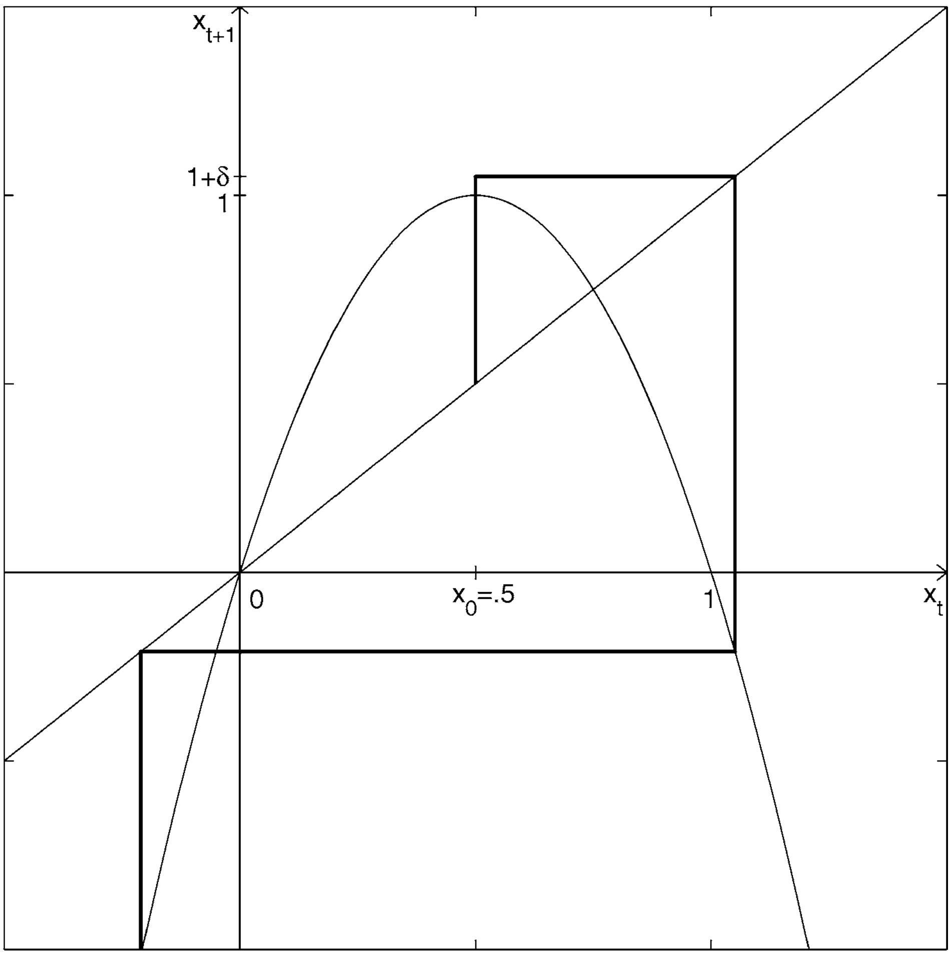

Fig. 1. *Example of glitch for the logistic map*

For chaotic as well as stochastic processes, it is fundamental to quantify the variability of the estimates, constructing, for example, confidence intervals. However, trying to derive analytically the sample distribution of the estimates in many cases is a very complex and difficult task. Bailey *et al.* (1997) present a central limit theorem for local Lyapunov exponents in stochastic systems whose state-space representation is

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t) + \varepsilon_t, \tag{3}$$

where f is a deterministic map and $\{\varepsilon_t\}$ is a sequence of i.i.d. random variables. An other asymptotic result concerning the estimate of the Lyapunov exponent λ_1 in the case of unidimensional maps is given by Lai and Chen (1995).

The alternative is to use a data-resampling method or bootstrap.

In a general probabilistic situation, one can summarize the bootstrap method following the scheme proposed by Efron and Tibshirani (1986) (see Fig. 2): suppose we have a sample $\mathbf{X} = \{x_t\}_{t=1}^N$ yielded by a statistical model P and let $\boldsymbol{\theta}_P(\mathbf{X}, N)$ the statistics of interest. The first step in the bootstrap is to estimate P with data $\mathbf{X}$. Let $\mathbf{X}^*$ be a bootstrap data set generated from the estimated model $\hat{P}$. The conditional distribution of $\boldsymbol{\theta}_{\hat{P}}(\mathbf{X}^*, N)$ given $\mathbf{X}$ is then the bootstrap estimator of the distribution of $\boldsymbol{\theta}_P(\mathbf{X}, N)$.

The original jackknife and bootstrap were developed for independent and identical distributed data, so they cannot be roughly applied to non-i.i.d. cases like time series and other dependent data. For this purpose, a special class of bootstrap methods for stationary time series has been developed in the last decade. Two different approaches can be identified. The first is a parametric model-based approach, which is founded on the idea of fitting parametric models first and then resampling from the residuals (Efron and Tibshirani 1993). On the contrary, the second approach is based on a nonparametric, purely model-free bootstrap scheme. One of the earliest non parametric methods is the

$$\begin{tikzcd}
P \arrow[rr] \arrow[dr] & & \mathbf{X} \arrow[dl] \arrow[r, Rightarrow] & \hat{P} \arrow[rr] \arrow[dr] & & \mathbf{X}^* \arrow[dl] \\
& \boldsymbol{\theta}_P(\mathbf{X}, N) & & & \boldsymbol{\theta}_{\hat{P}}(\mathbf{X}^*, N) &
\end{tikzcd}$$

Fig. 2. *Diagram of the bootstrapping process in a probabilistic framework*

grouping bootstrap proposed by Carlstein (1986). Lai and Chen (1995) apply this scheme to estimate the empirical distribution of an estimator of λ_1 for chaotic time series.

A more efficient non parametric method is the *blockwise* or *moving block bootstrap* developed by Künsch (1989). Künsch's idea is to resample overlapping blocks of consecutive observations and to concatenate the selected blocks. All the same, the series obtained in this way show artifacts which are caused by joining randomly selected blocks, so the serial dependence is preserved within, but not across, the blocks. Developments of this method are in Politis and Romano (1994) and in Carlstein *et al.* (1998). Another promising methodology is the *sieve bootstrap*, recently proposed by Bühlmann (1997).

Lall and Sharma (1996) introduce a methodology called *nearest neighbor bootstrap* which makes use of the time delay method and the nearest neighbor techniques. After reconstructing the series in a d-dimensional phase space using the method of delay (see Appendix), the algorithm starts choosing an initial point $\mathbf{x}_t$ and the relative k-nearest neighbors $\{\mathbf{x}_{ti}\}_{i=1}^{k}$. The k successors of these points, say $\mathbf{x}_{(t+1)i}$, are then resampled according to a discrete kernel K and a new vector $\mathbf{x}_{t+1}$ is produced. The successors of the k-nearest neighbors of $\mathbf{x}_{t+1}$ are subsequently resampled giving $\mathbf{x}_{t+2}$, and so on. This method does not need prior assumptions on the form of the dependence or the form of the probability density function and it seems to be able to preserve the dependence structure in many real cases, like hydrologic time series.

In general, all these methods, even if they have a validity in the stochastic ambit, do not seem to be applicable to time series yielded by a chaotic DGP. As a matter of fact, the application of these bootstrap schemes to deterministic time series produces series in which the original dynamics is partially or completely destroyed. Lall and Sharma (1996)'s method, for example, which seems at first sight the most suitable among all, actually leads, step after step, to 'jump' randomly from an orbit to a nearby one, producing time series which look like noisy chaotic orbits. This is the same as inserting a spurious randomness in the bootstrap estimation of the sample distribution of $\boldsymbol{\theta}_f(\mathbf{x}_0, N)$.

At our knowledge, there are only few works which propose specific bootstrap techniques for chaotic time series. Gencay (1996) bootstraps the points in the reconstructed phase space rather than in the time domain and, from a certain point of view, this is equivalent to a moving block bootstrap. Ziehmann *et al.* (1998) show the inappropriateness of Gencay's method for dependent data and propose an alternate bootstrap scheme (the *dynamically conditioned bootstrap*) for aleatoric systems as (3).

Here we propose a completely different approach. Following the scheme of Fig. 2, it can be summarized as follows (see Fig. 3): the sample data set $\mathbf{X}(\mathbf{x}_0) \equiv \{f, \mathbf{x}_0, N\} = \{\mathbf{x}_t\}_{t=0}^{N}$ is the piece of orbit starting from $\mathbf{x}_0$ with length N, yielded by the unknown deterministic model f, and $\boldsymbol{\theta}_f(\mathbf{x}_0, N)$ is the statistics of interest. The first step in our resampling method is the estimation of f using the data set $\mathbf{X}(\mathbf{x}_0)$. The resampled data sets $\mathbf{X}^*$ are then generated using the estimated model $\hat{f}$. Taking into account that the variability of $\boldsymbol{\theta}_f(\mathbf{x}_0, N)$ depends on the piece of the observed orbit and, by consequence, on the initial condition $\mathbf{x}_0$, we yield the bootstrap data sets $\mathbf{X}^*(\mathbf{x}_{0j}) \equiv \{\hat{f}, \mathbf{x}_{0j}, N\} = \{\mathbf{x}_t^*\}_{t=0}^{N}$ choosing a suitable set of different initial conditions $\mathbf{x}_{0j}$ and iterating N times the model $\hat{f}$:

$$\begin{tikzcd}
f \arrow[rr] \arrow[dr] & & \{\mathbf{x}_t\}_{t=0}^{N} \arrow[dl] \arrow[rr, Rightarrow] & & \hat{f} \arrow[rr] \arrow[dr] & & \{\mathbf{x}_t^*\}_{t=0}^{N} \arrow[dl] \\
& \boldsymbol{\theta}_f(N, \mathbf{x}_0) & & & & \boldsymbol{\theta}_{\hat{f}}(N, \mathbf{x}_0^*) &
\end{tikzcd}$$

Fig. 3. *The resampling procedure for chaotic time series*

$$\mathbf{x}_t^* = \hat{f}(\mathbf{x}_{t-1}^*) = \hat{f} \circ \hat{f} \circ \cdots \circ \hat{f}(\mathbf{x}_{0j}).$$

In the next section we give a detailed description of the algorithm together with an overview of the methods for the optimal choice of its parameter values.

4. The algorithm

As sketched in the previous section, our data-resampling method is composed by three fundamental steps:

1) the estimation of the flow or map $\hat{f}$
2) the proper choice of a set of initial conditions $\{\mathbf{x}_{0j}\}_{j=1}^{M}$
3) the calculation of the orbits which start from $\{\mathbf{x}_{0j}\}_{j=1}^{M}$ by repeated applications of $\hat{f}$, $\mathbf{X}^*(\mathbf{x}_{0j}) = \{\mathbf{x}_t^*\}_{t=0}^{N}$.

Regarding the estimate of f, it is possible to choose between global or local models (Farmer and Sidorowich 1988) and between linear and nonlinear models. In the present work, we adopt a nearest neighbor neural model, i.e. a nonlinear, local model. As shown in the next section, this model offers some important advantages.

Regarding the second stage of the algorithm, assuming the ergodicity of the dynamical system under study, the choice of the initial conditions is not problematic, and it does not need to be random (Berliner 1992). Taking into account that the $\mathbf{x}_0$'s have to stay into the basin of attraction of the system, we preferred to choose them randomly selecting one of the N points of the sample orbit and adding a 'small' random disturbance η.

Given $\hat{f}_{\mathbf{x}}$, the local estimator of f at the point $\mathbf{x}$, the j-th resampled orbit starting from $\mathbf{x}_{0j}$ is $\mathbf{X}^*(\mathbf{x}_{0j}) = \{\mathbf{x}_t^*\}_{t=0}^{N}$, where $\mathbf{x}_t^* = \hat{f}_{\mathbf{x}_{t-1}^*}(\mathbf{x}_{t-1}^*)$. In order to avoid transients, we always discard the first n_T iterations.

The proposed deterministic resampling algorithm can be described in details as follows:

1. Starting from the observed time series $\{y_t\}_{t=1}^{N}$, use the time delay method to build the points $\mathbf{x}_t \in \mathbf{R}^d$ in the reconstructed phase space, where $\mathbf{x}_t = \{y_t, y_{t+\tau}, y_{t+2\tau}, \ldots, y_{t+(d-1)\tau}\}$, d is the embedding dimension and τ is the time delay (see a brief description of the method in the Appendix).
2. Choose an initial condition $\mathbf{x}_0$.
3. Look for the k closest points to $\mathbf{x}_0$, $\{\mathbf{x}_{n_i}\}_{i=1}^{k}$, i.e. find the k points which minimize the distance from the reference point

$\mathbf{x}_0$; this set of points represents the total input of the neural network.

4. Select the successors $\{\mathbf{x}_{n_i+1}\}_{i=1}^{k}$, i.e. find the k points in the phase space which correspond to the evolutions of the dynamical system starting from the set of points $\{\mathbf{x}_{n_i}\}_{i=1}^{k}$ selected in step 3; this set of points is the total output of the neural network.
5. Locally estimate the flow f fitting a neural model $\hat{f}_{\mathbf{x}_0}$ to the two sets of points in $\mathbf{R}^d$ calculated in steps 3 and 4.
6. Put $\mathbf{x}_0$ in the input of the network and calculate $\mathbf{x}_1^* = \hat{f}_{\mathbf{x}_0}(\mathbf{x}_0)$; this is the first point of the new orbit which is going to be constructed.
7. Go back to 3, look for the k nearest points of $\mathbf{x}_1^*$, estimate $\hat{f}_{\mathbf{x}_1^*}$, compute $\mathbf{x}_2^* = \hat{f}_{\mathbf{x}_1^*}(\mathbf{x}_1^*)$, and so on.

The parameters involved in the deterministic resampling algorithm are the number h of hidden units of the neural network, the time delay τ, the embedding dimension d and the number of nearest neighbors k.

One of the most important problems encountered in the practical application of neural networks is to find a suitable minimal topology. In fact, too large a value of h increases the time required by the training procedure, may cause non-convergence of the error-minimization algorithm and usually decreases the generalization capabilities of the network. Pruning methods are a basic approach to find optimal network topologies. The idea behind these methods is to remove neurons or connections (during or after the training process) according to an optimality criteria. Optimal Brain Damage and Optimal Brain Surgeon represent two popular pruning procedures (see Thimm and Fiesler 1997). White (1992) gives useful indications on the choice of the optimal number of neurons as a function of the sample size N.

The proper choice of the time delay τ, together with the embedding dimension d, is of great importance in the process of reconstruction of the attractor. For small τ the dynamics take place on the hyper-diagonal of the embedding space so x_t and $x_{t+\tau}$ become almost linear dependent which is not the case for the real observables of a nonlinear system. Too large a τ causes the coordinates to disjoin by the stretching and folding, so the characteristic structures tend to disappear. There are many methods for choosing τ (see Rosenstein *et al.* 1994). Usually a good choice is based on mutual information (Fraser and Swinney 1986). If d is selected too small, there could be intersection problems for the orbits, so points which seem near, are actually very far; if d is chosen too big, this reduces the density of the utilizable points. The method of false nearest neighbor is often used for choosing the right d (Kennel *et al.* 1992).

With regard to k, it is important to be reminded that the choice of too small a k involves difficulties in the estimation of f with the neural net, in fact few points of input and output are used, so the estimation becomes more approximate and rough. On the contrary, too big a k introduces an error in the estimation of f caused by having to consider points which are actually very far from the reference one.

Because our target is to minimize the approximation errors in the estimation of f, we prefer to choose jointly the embedding dimension d and the number of nearest neighbors k. One criterium is to minimize the one-step prediction error on an out-of-sample testing set. The data set is divided in two sets of unequal length. The larger one is called *training* or *learning set* and is used to estimate the parameters of the local neural networks. The forecasting performances of the networks are tested on the second set, called *testing set*. In other words, for different values of d and k, one fits the neural model to the learning set and then estimates the forecasting error on the out-of-sample testing set. The optimal d and k correspond to the minimum of the error function. Being aware of the potentially dangerous effects of data splitting (LeBaron and Weigend 1997), we repeated the calculus of d and k for different splitting of the sample series.

5. Theoretical basis

A theoretical question must be faced here: beyond the empirical evidence (shown in section 6), is it correct to state that our method is able to produce resampled orbits with the same dynamical and geometrical properties of the true DGP? In other words, under what conditions does the algorithm correctly approximate the real randomness of the estimators of interest?

Let us consider the problem using three different levels of abstraction:

1) f is known and calculations are made by an infinite-precision computer.
Under these (unrealistic) conditions, our algorithm produces, for each randomly chosen initial condition $\mathbf{x}_{0j}$, resampled series $\{\mathbf{x}_t^*\}_{t=0}^{N}$, where $\mathbf{x}_{t+1}^* = f(\mathbf{x}_t^*)$ and $\mathbf{x}_0^* = \mathbf{x}_{0j}$, which are true orbits of the underlying DGP. Hence, the distribution of the estimator $\boldsymbol{\theta}_f(\mathbf{x}_0, N)$ is exactly reproduced. It is worth pointing out that, because f is known, this is merely a Monte Carlo experiment.

2) f is known and calculations are made by a finite-precision computer.
Under these conditions, iterating f, our algorithm produces δ-pseudo orbits $\{\mathbf{x}_t^*\}_{t=0}^{N}$, $\mathbf{x}_{t+1}^* = f(\mathbf{x}_t^*) + \delta_t$, where δ_t represents numerical noise due to truncation or rounding errors made by the finite-precision computer at any step. For dynamical systems in continuous time, where usually only the vector field F is known and solutions often cannot be obtained in closed form, δ_t also includes the discretization error induced by the integration method (e.g. Euler's or Runge-Kutta methods).

Here we need to guarantee that each resampled series can be legitimately considered as the product of the underlying DGP. The first requirement is therefore the condition that true orbits ϵ-close to the computed δ-pseudo orbits exist. As stated in section 2, this is true is f has the shadowing property. However, shadowing by itself is not quite enough to ensure that the resampling scheme is working correctly. The true orbits doing the shadowing must also be, in some

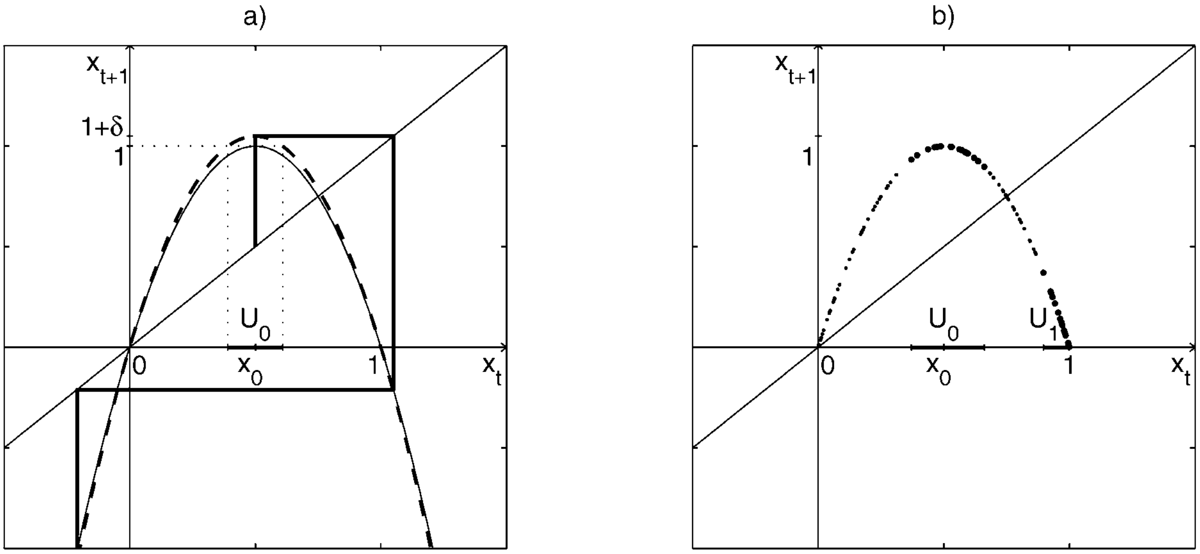

Fig. 4. *Robustness to glitches in the case of a) a global estimator $\hat{f}$ and b) a local one*

sense, *typical* orbits. Briefly, typical or generic orbits are the ones that generate the invariant distribution of the system and that start from a measure-one set of initial conditions. McCauley (1994) correctly points out that this set is the class that would theoretically occur with absolute certainty if one could make a truly random draw of numbers, with infinite precision, from the continuum. Therefore, while typical trajectories are pervasive in natural experiments (and are the only observable ones), they could not be observable in numerical experiments. Unlike shadowability, the study of the conditions under which shadowing orbits are typical is still in its infancy (see McCauley 1994 and the references therein).

Another fundamental question that must be considered is the dependence of ϵ on δ, which in turn influences the rate of convergence of our bootstrap approximations. When a Lyapunov exponent fluctuates about zero, Sauer *et al.* (1997) find that ϵ follows a power law distribution $c \cdot \epsilon^{-2m/\sigma^2}$, where m and σ are the mean and the standard deviation of the time-1 Lyapunov exponent closest to zero. In other words, the greater the finite time fluctuation about zero, the smaller the power law exponent, resulting in large shadowing distances.

As in the previous case, one must note the coincidence between our resampling algorithm and a Monte Carlo replication scheme. This means that the problems discussed above (shadowability, 'genericness' of shadowing orbits) are common to any numerical experiments and not a specific weakness of our DRA.

3) f is unknown and the estimated $\hat{f}$ is used in place of f. Iterating $\hat{f}$, our algorithm generates resampled orbits which are pseudo orbits $\{\mathbf{x}_t^*\}_{t=0}^N$, $\mathbf{x}_{t+1}^* = \hat{f}(\mathbf{x}_t^*) = f(\mathbf{x}_t^*) + \delta_t$, where δ_t represents the truncation or rounding errors plus the error due to the approximation of f by $\hat{f}$.

As stated above, the correct functioning of the DRA depends on the following two conditions: 1) f must have the POTP and 2) the shadowing orbits must be typical. However, in this case, they are not sufficient. Two other requirements must be imposed. First, the estimator $\hat{f}$ has to be more and more close to f as N grows, that is $\hat{f}$ must be a consistent estimate of f. This ensures that, as N grows, the probability of approximation error exceeding any special level, tends to zero. Moreover, the piece of orbit which represents our sample data must belong to a true typical trajectory of the underlying DGP or must belong to a pseudo orbit shadowed by a true typical orbit. In other words the sample has to be representative.

Therefore, the crucial question here is not, as erroneously stated by Farmer and Sidorowich (1988) p. 316, to get 'good' forecasts of the true trajectories starting from the randomly chosen initial points $\mathbf{x}_{0j}$. This is a particularly strong condition when treating chaotic DPGs because 'good' forecasts are only possible for short time periods and, for our purposes, it is quite unimportant if each resampled series $X^*(\mathbf{x}_{0j})$ starting from a given initial condition $\mathbf{x}_{0j}$ moves close or not to the true orbit $X(\mathbf{x}_{0j})$ starting from the same point. Rather, we need a 'good' approximation of the flow or map f which, together with 'typical' POTP, guarantees that the resampled trajectories $X^*(\mathbf{x}_{0j})$ stay ϵ-close to typical true orbits (usually unknown).

When f is unknown and only a sample series $\{\mathbf{x}_t\}_{t=0}^N$ is available, testing the presence of POTP is a challenging task. The first strategy one can follow is to test if one of the conditions for nonshadowability is present (for example the presence of a finite time Lyapunov exponent fluctuating around zero). Secondly, one could try to verify, through suitable tests (see for example the statistics for continuity of Pecora *et al.* 1997), if f has one of the properties which assures the shadowability (for instance, f is an homeomorphism).

The choice of a neural estimator is motivated by two properties. First, the feedforward neural networks can approximate virtually any function of interest to any desired degree of accuracy, provided many sufficiently hidden units are available. In other words, this kind of neural network represents a class of universal approximators (Hornik *et al.* 1992). Secondly, White (1992) demonstrates that, under mild conditions, feedforward neural networks are consistent estimators of f. In the specific context of chaotic dynamics, neural networks are found to be competitive with the best of the approximation methods in the construction of a nonlinear map from a given time series as well as in the recovering of the derivatives of a nonlinear map (Gallant and White 1992).

Moreover, we have decided to use a local model instead of a global one because it shows itself to be more 'robust' in the presence of glitches. Recalling the example of Fig. 1, consider a global estimator of the logistic map f. Suppose to avoid numerical errors and of overestimate f in a neighborhood of the glitch $\mathbf{x}_0 = 0.5$. Fig. 4(a) shows that, if the resampled trajectory enters within the neighborhood U_0 of $\mathbf{x}_0$ (whose amplitude depends on how much we overestimate f), we have $\hat{f}(\mathbf{x}_0) > 1$ and the orbit diverges to $-\infty$, loosing shadowability.

Consider now a local estimator (see Fig. 4(b)). Let $U_0 \subset [0, 1]$ be the set of the k-nearest neighbors of the glitch $\mathbf{x}_0$ and $U_1 \subset [0, 1]$ the set of the one-step evolutions of these k points. Our local estimator is therefore a map $\hat{f}_{\mathbf{x}_0} : U_0 \rightarrow U_1$. This means that, in spite of the presence of the glitch, $\hat{f}_{\mathbf{x}_0}(\mathbf{x}_0)$ belongs to U_1 and the resampled orbit will never diverge, (hopefully) continuing to be close to a shadowing orbit.

6. Applications

In this section we present some applications of the proposed algorithm to the Lorenz flow, which owns the shadowing property (see Coomes *et al.* 1995), and to a real time series. In Golia and Sandri (1997) the algorithm was tested on the logistic map.

The first application concerns a series of 1000 points (as in common real situations, we prefer to test the method on a relatively short time series) generated by the noise-free Lorenz dynamical system:

$$\begin{aligned}\dot{x} &= \sigma(y - x)\\ \dot{y} &= Rx - y - xz\\ \dot{z} &= -bz + xy\end{aligned}$$

with $\sigma = 16$, $R = 45.92$, $b = 4$. The values of the correlation dimension and the largest Lyapunov exponent are approximately $d_{\text{corr}} = 2.06$ and $\lambda_1 = 1.50$ respectively. Setting $\tau = 5$ and $h = 5$, in Fig. 5 we trace the root mean square (one-step) prediction

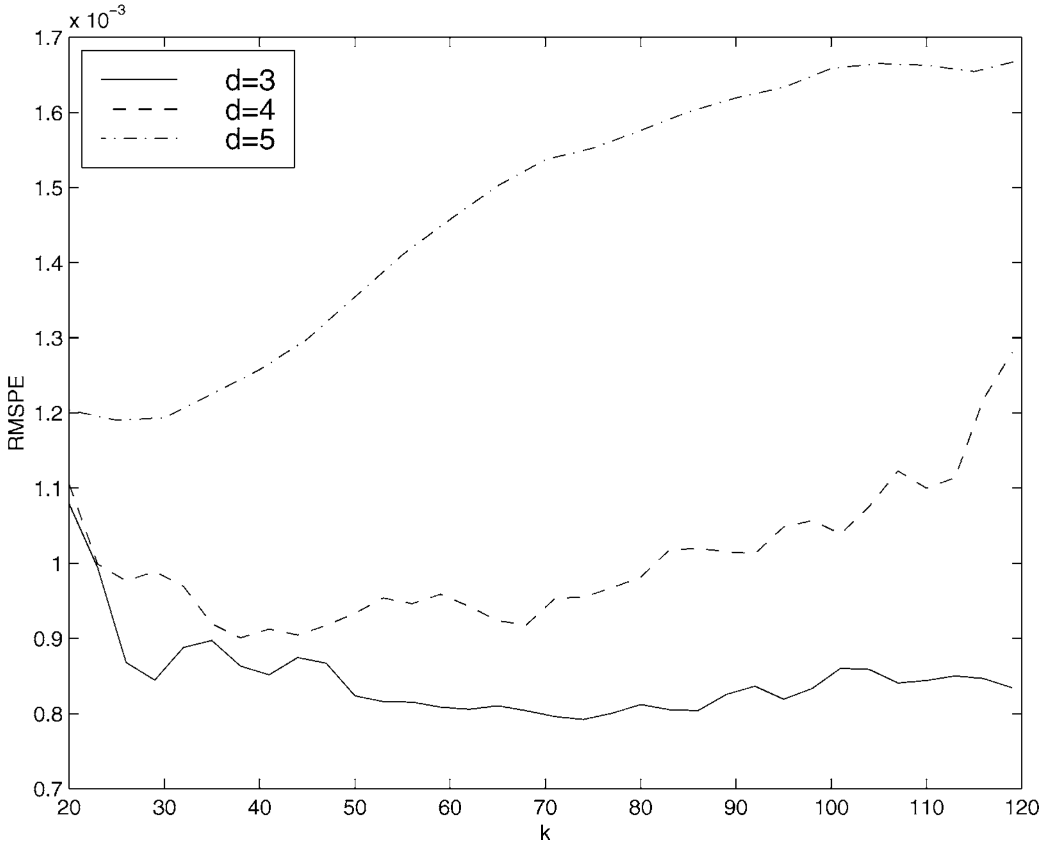

Fig. 5. *Root mean square prediction error as a function of the embedding dimension d and the number of nearest neighbors k*

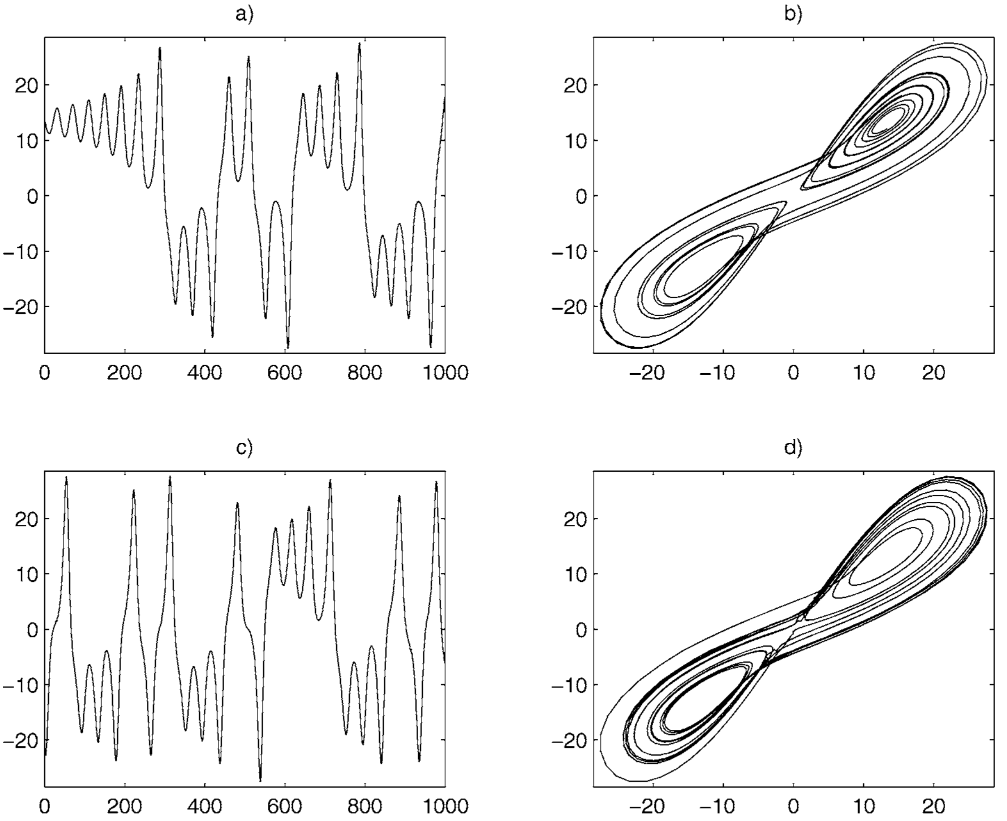

Fig. 6. *a) Time series yielded from the Lorenz model and b) its 2D reconstruction c) Time series simulated by the proposed method and d) its 2D reconstruction*

error (RMSPE) as a function of the embedding dimension d and of the number of nearest neighbors k. It is evident that the optimal parameters are $d = 3$ and $k \approx 80$.

Figure 6 shows the true and a resampled series plotted in the time domain and in the 2-dimensional reconstructed phase space respectively. From a qualitative point of view, no remarkable discrepancies are evident.

The power spectra of the true and resampled series are shown in Fig. 7. One can see that they have approximately the same shape. For high frequences the spectrum of the resampled series shows a shift due to the presence of the resampling noise δ.

In Fig. 8 the box-plot of the autocorrelation estimates for 200 true Lorenz series and 100 resampled series are compared. The autocorrelation structure of data seems correctly reproduced together with the distributional properties of estimates.

Using Grassberger and Procaccia (1983)'s estimator for d_{corr} and Rosenstein *et al.* (1993)'s estimator for λ_1, we calculate the Lyapunov exponent and the correlation dimension for the 200 true and 100 resampled time series. In Table 1 we propose some descriptive statistics for the Lyapunov exponent and correlation dimension estimations: the sample median, mean and standard deviation.

Table 2 shows some percentiles of the empirical distributions of $\hat{\lambda}_1$ and $\hat{d}_{\text{corr}}$ estimated by the true and resampled Lorenz series respectively.

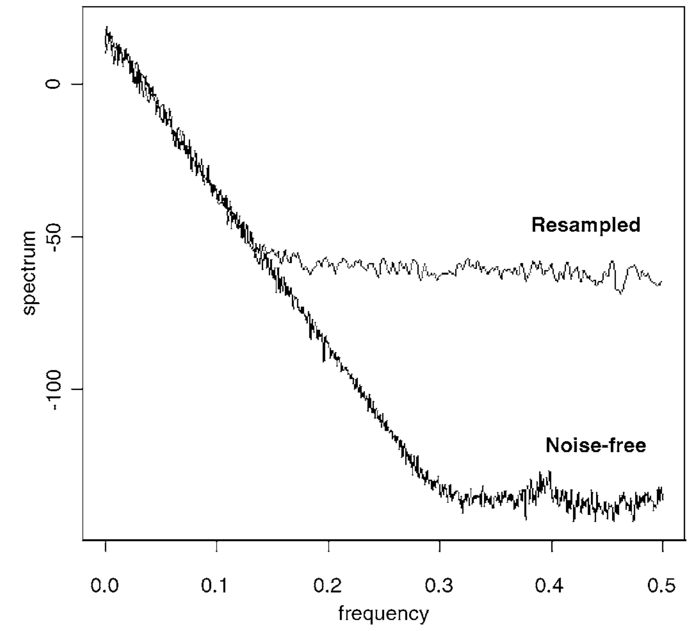

Fig. 7. *a) Power spectra of a noise-free Lorenz series and b) of a resampled one*

Table 1. *Descriptive statistics of based on 200 Lorenz replications and on 100 resampled Lorenz replications respectively*

		Median	Mean	Std. Dev.
Lyapunov Exponent	'True' Replications	1.49	1.50	0.35
	Resampled Replications	1.50	1.54	0.37
Correlation Dimension	'True' Replications	1.90	1.91	0.22
	Resampled Replications	1.88	1.90	0.15

Table 2. *Percentiles of $\hat{\lambda}_1$ and $\hat{d}_{\text{corr}}$ based on 200 Lorenz replications and on 100 resampled Lorenz replications respectively*

		2.5%	5%	10%	16%	50%	84%	90%	95%	97.5%
$\hat{\lambda}_1$	'True' Rep.	0.88	0.93	1.02	1.16	1.49	1.84	1.96	2.11	2.24
	Resampled Rep.	0.86	0.99	1.11	1.24	1.50	1.89	2.18	2.28	2.35
$\hat{d}_{\text{corr}}$	'True' Rep.	1.56	1.64	1.68	1.73	1.896	2.05	2.12	2.21	2.53
	Resampled Rep.	1.64	1.66	1.73	1.77	1.88	2.01	2.06	2.11	2.22

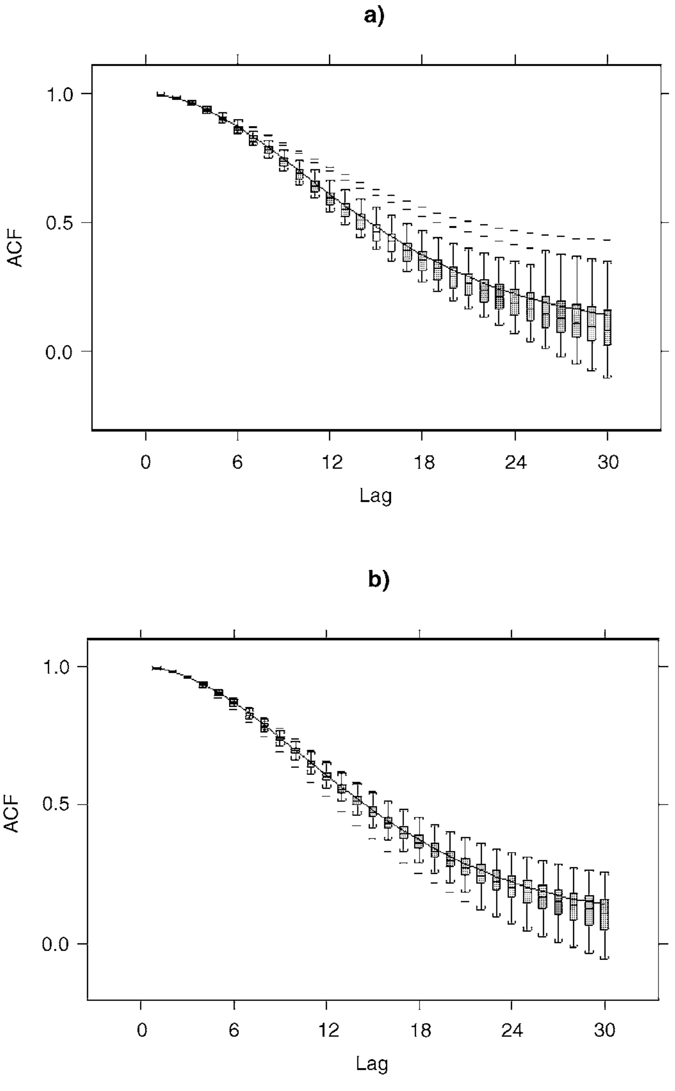

Fig. 8. *Box-plot of the autocorrelations for a) 100 true Lorenz time series and b) 100 resampled ones*

Table 3. *p-values of the two nonparametric tests for Lorenz model*

	Equal location parameters	Equal distribution
Lyapunov Exponent	0.53	0.76
Correlation Dimension	0.65	0.33

As an extra confirmation that the values of the estimates of the parameters λ_1 and d_{corr} computed using the resampled series are reproducing the real randomness of the chosen estimators $\hat{\lambda}_1$ and $\hat{d}_{\text{corr}}$ for the Lorenz model, we apply the following two nonparametric tests: the Wilcoxon test (equal location parameters) and the Kolmogorov-Smirnov test (equal distribution) (see Conover 1980). On the basis of the p-values reported in Table 3 we can accept the whole set of hypotheses, within a significance level of 5%.

The results give us confirmation that, in the present application, the method is working correctly. We can therefore state that the dynamical and geometrical properties of the resampled time series and of the series generated by the DGP under study are the same and the randomness of the estimators is preserved.

In the second experiment we consider a deterministic series contaminated by a strong additive Gaussian white noise. The data are yielded by taking the Lorenz series described in the previous application and adding an i.i.d. Gaussian noise whose variance σ_n^2 is equal to the variance σ_s^2 of the noise-free component (SNR $=$ 0dB). This is a particularly hard test which allows us to explore the power of the proposed method on a more realistic ground.

We do not directly apply the resampling procedure to the raw series because we believe that the deterministic and stochastic components of the data set cannot be subjected to the same resampling 'treatment': while one can bootstrap the stochastic part using one of the methods described in Section 3, the deterministic signal require a completely different 'ad hoc' resampling scheme.

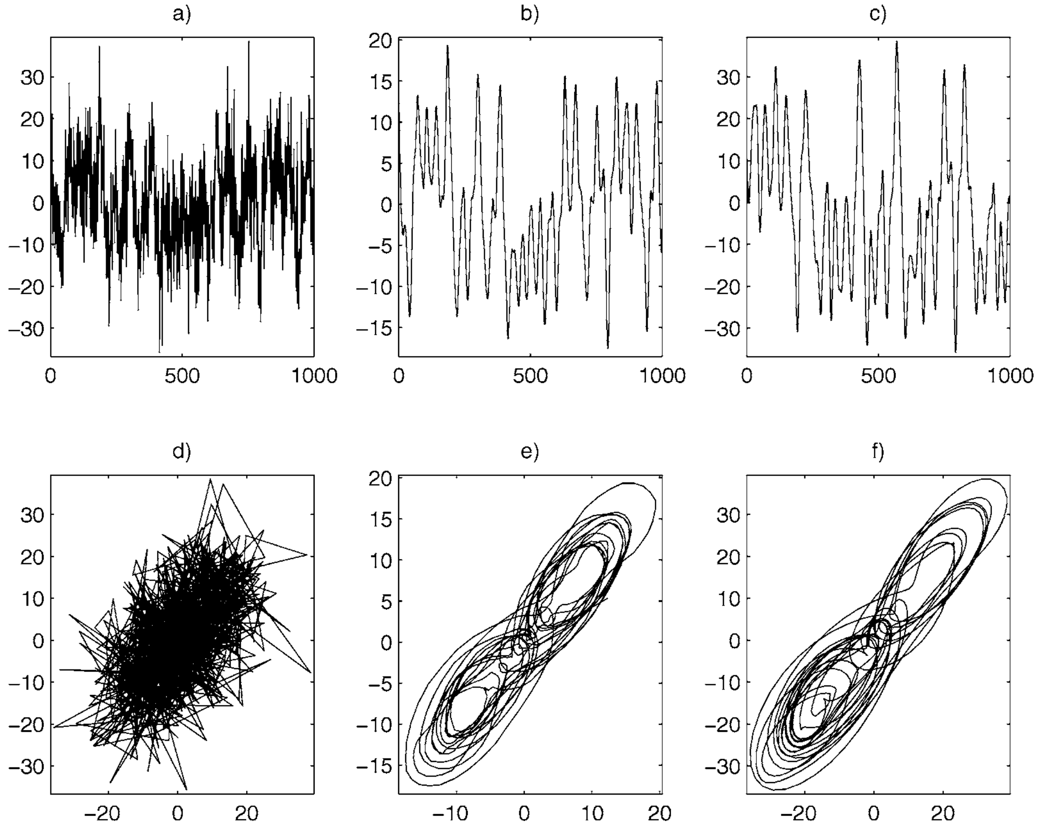

Fig. 9. *a) Noisy Lorenz time series and d) its 2D reconstruction, b) filtered Lorenz time series and e) its 2D reconstruction, c) resampled time series from the filtered Lorenz series and f) its 2D reconstruction*

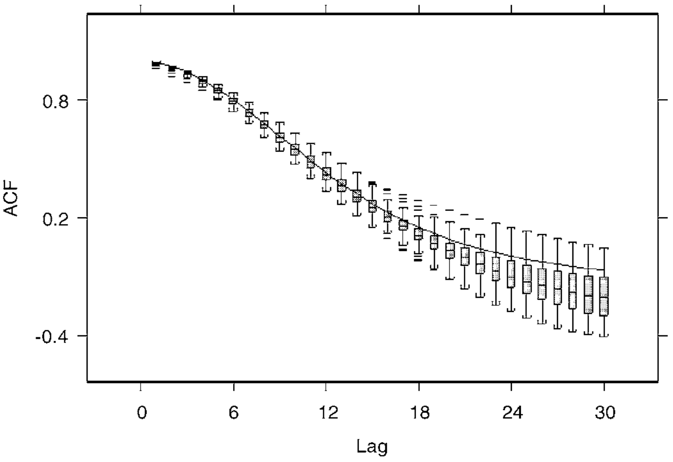

Fig. 10. *Box-plot of the autocorrelations for* Lor_{fil} *and the 100 resampled time series*

In order to distinguish these two components we must face here an identification problem. There is a large amount of literature on the subject, also for chaotic signal (see the review article of Grassberger *et al.* 1993). In our simulations we use a non-parametric filtering method called Singular Spectrum Analysis (SSA) (see Broomhead and King 1986 and Vautard *et al.* 1992), based on the classical principal component analysis. Choosing a number of principal directions equal to 2 (Lisi *et al.* 1995), we divide the noisy series, which we call $\text{Lor}_{\text{noise}}$, in two components:

$$\text{Lor}_{\text{noise}} = \text{Lor}_{\text{fil}} + \varepsilon$$

where Lor_{fil} is an estimate of the deterministic part of the original series and ε represents the stochastic component. The noisy Lorenz series, the filtered series and a resampled one, together with their reconstructed 2-dimensional attractors are plotted in Fig. 9.

Table 4. *Descriptive statistics of the 100 resampled filtered Lorenz time series*

	Median	Mean	Std. Dev.
Lyapunov Exponent	1.51	1.53	0.53
Correlation Dimension	2.16	2.20	0.21

Applying our deterministic resampling algorithm on Lor_{fil} with parameters $d = 4$, $\tau = 5$, $h = 5$ and $k = 20$, we yield 100 time series. In the first step we compare the autocorrelation structure for Lor_{fil} and the resampled series.

The graph in Fig. 10 shows clearly how, in this case, such structure is preserved by the resampling procedure. The estimated values of the correlation dimension and the Lyapunov exponent for Lor_{fil} are: $\hat{\lambda}_1 = 1.398$ and $\hat{d}_{\text{corr}} = 2.218$. Table 4

Table 5. *Percentiles of $\hat{\lambda}_1$ based on 100 resampling filtered Lorenz replications*

2.5%	5%	10%	16%	50%	84%	90%	95%	97.5%
0.64	0.78	0.88	0.98	1.51	2.00	2.14	2.52	2.59

Table 6. *Percentiles of $\hat{d}_{\mathrm{corr}}$ based on 100 resampling filtered Lorenz replications*

2.5%	5%	10%	16%	50%	84%	90%	95%	97.5%
1.90	1.93	1.98	2.01	2.16	2.41	2.51	2.61	2.68

shows the descriptive statistics of the same estimations corresponding to the resampled series.

Then we want to draw a confidence interval for the estimated values of λ_1 and d_{corr}. In order to do that, we use the *percentile interval* that is a confidence interval based on the percentiles of the bootstrap distribution of the parameters (Efron and Tibshirani 1993). The $1 + 2\alpha$ percentile interval is defined as

$$[\hat{\theta}_{\%,low}, \hat{\theta}_{\%,up}] = \left[\hat{\theta}^{*(\alpha)}, \hat{\theta}^{*(1-\alpha)}\right]$$

where $\hat{\theta}^{*(\alpha)}$ is the $100{\cdot}\alpha$th percentile of the bootstrap distribution.

Tables 5 and 6 show some percentile values for the estimations of λ_1 and d_{corr}.

The 0.95 percentile interval for $\hat{\lambda}_1$ is [0.6374, 2.587] and the 0.95 percentile interval for $\hat{d}_{\mathrm{corr}}$ is [1.8975, 2.683].

In the last application we test our deterministic resampling algorithm on a real time series: the monthly Southern Oscillation Index (SOI) which measures the difference between the pression level in Tahiti and Darwin Sea (Australia). The series contains 588 observations. A preliminary analysis of the data set shows the presence of a deterministic signal contaminated by a high amount of noise, a situation very similar to the 0 dB noisy Lorenz series considered above. Applying the filtering method based on SSA (number of principal directions $p = 3$, see Lisi *et al.* 1995), we get the series shown in Fig. 11(b). A resampled series is plotted in 11(c).

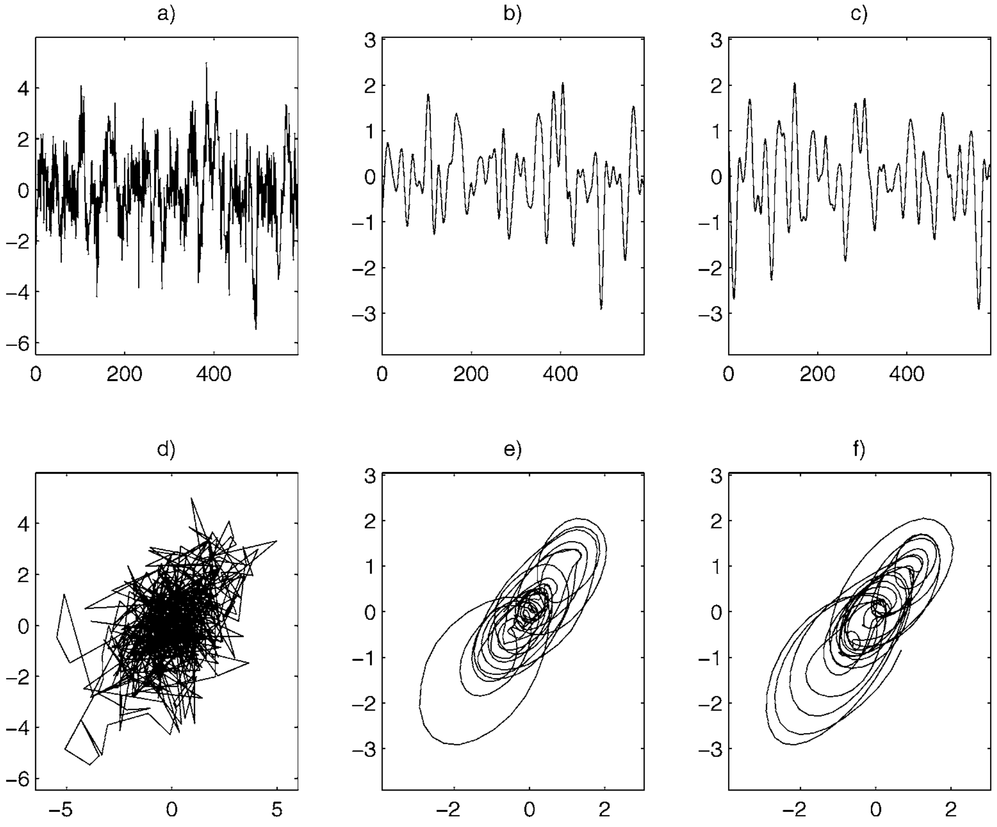

Fig. 11. *a) SOI time series and d) its 2D reconstruction, b) filtered SOI and e) its 2D reconstruction, c) resampled time series from the filtered SOI and f) its 2D reconstruction*

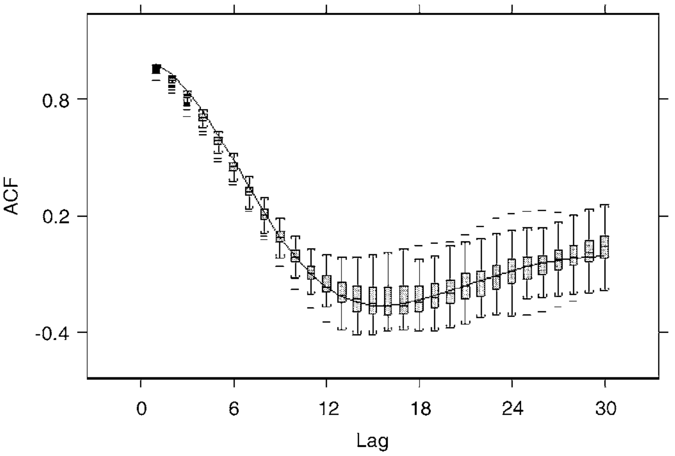

Fig. 12. *Box-plot of the autocorrelations for the filtered SOI time series and the 100 resampled ones*

Table 7. *Descriptive statistics of the 100 resampled filtered SOI time series*

	Median	Mean	Std. Dev.
Lyapunov Exponent	0.0364	0.0357	0.007
Correlation Dimension	3.17	3.19	0.43

The deterministic resampling algorithm parameters for the filtered SOI are: $d = 5$, $\tau = 5$, $h = 5$ and $k = 20$. Again, we yield 100 resampled time series and analyze the autocorrelation structure of the filtered SOI series and the resampled ones (see the box-plot of Fig. 12).

The sample median, mean and standard deviation of the estimates of the largest Lyapunov exponent and of the correlation dimension for the resampled series are shown in Table 7.

The corresponding estimates for the filtered SOI are: $\hat{\lambda}_1 = 0.03523$ and $\hat{d}_{\text{corr}} = 3.227$. We report the percentiles of $\hat{\lambda}_1$ and $\hat{d}_{\text{corr}}$ in Tables 8 and 9.

The 0.95 percentile interval for the largest Lyapunov exponent is [0.0216, 0.0492]. The corresponding interval for $\hat{d}_{\text{corr}}$ is [2.3298, 4.0605].

7. Conclusions

In this paper we present a resampling algorithm for chaotic time series. The basic idea is to develop a method for building trajectories which lie on the same attractor of the underlying DGP, that is series with the same dynamical and geometrical properties of the original data.

The tests performed on short noise-free and noisy Lorenz series confirm that our deterministic resampling algorithm correctly reproduce the distribution of the largest Lyapunov exponent, of the correlation dimension and of the autocorrelation function. We can therefore give estimates of the confidence intervals for these system invariants and throw the bases for testing the hypothesis of chaoticity.

In the present work, we approximate the unknown flow making use of local feed-forward neural networks. This is only one of the many statistical instruments available for nonlinear model estimation. Experimenting new and more advanced models (e.g. the recurrent neural networks proposed by Mozer 1994) is an interesting direction for future research.

Appendix

The reconstruction of the phase space is usually the first step in the analysis of dynamical systems. The most widespread approach to this problem is the so-called *time delay method* because it is the most straightforward and the noise level is constant for each delay component. It consists in using time delay coordinates (see Broomhead and King 1986) to form the multiple state-space vectors $\mathbf{x}_t \in \mathbf{R}^d$.

Given a time series $\{x_t\}_{t=1}^n$, the reconstructed state of the system at each time t is:

$$\mathbf{x}_t = \left\{x_t, x_{t+\tau}, x_{t+2\tau}, \dots, x_{t+(d-1)\tau}\right\}$$

where τ is the time delay and d the embedding dimension for the DGP under study. These vectors form points on a trajectory in a d-dimensional space which is diffeomorphically related to the actual phase space of the real DGP. This result is due to Takens (1981) and it means that the reconstructed space, called embedding space, is topologically equivalent to the original one and the dynamical parameters are left invariant.

Table 8. *Percentiles of* $\hat{\lambda}_1$ *based on 100 resampling filtered SOI replications*

2.5%	5%	10%	16%	50%	84%	90%	95%	97.5%
0.021	0.022	0.027	0.029	0.0364	0.042	0.045	0.047	0.049

Table 9. *Percentiles of* $\hat{d}_{\text{corr}}$ *based on 100 resampling filtered SOI replications*

2.5%	5%	10%	16%	50%	84%	90%	95%	97.5%
2.33	2.54	2.77	2.84	3.17	3.59	3.79	3.89	4.06

Feed-forward neural networks are a class of nonlinear models which are capable of approximating any continuous function f uniformly on compacta, that is they are universal approximators. This property makes neural models very appealing in nonlinear regression. Moreover, under proper smoothness conditions on f, they permit to consistently estimate functionals of f, including partial derivatives (Gallant and White 1991).

The input values $\mathbf{x} \in \mathbf{R}^d$ (d is the embedding dimension) are received by the d input units, which pass them to the hidden units using a linear transformation determined by their connection strength γ_{ij}. Each hidden unit performs a nonlinear transformation on its total input, producing a total output using a sigmoid function $K(x)$, called activation function, which is the same for all the hidden units. The network output O can be represented as

$$O = \sum_{i=1}^{q} \beta_i K \left(\sum_{j=1}^{d} \gamma_{ij} x_j + b_i \right) \tag{4}$$

where β_i, with $\beta_i \in \mathbf{R}^q$, are the weights of the output produced by every hidden unit, q is the number of hidden units, and b_i is the bias of the hidden units.

A great advantage of the neural nets is their robustness against an incorrect choice of the model's dimension d (in the chaotic contest is the embedding dimension).

Acknowledgments

We wish to thank L.A. Smith and M. Spano for many useful comments and suggestions. We also thanks R. Perli and D. Pirrie.

References

Bailey B.A., Ellner S., and Nychka D.W. 1997. Chaos with confidence: Asymptotics and applications of local lyapunov exponents. In: D.C. Cutler and D.T. Kaplan (Eds.), Nonlinear Dynamics and Time Series. Building a Bridge Between the Natural and Statistical Sciences. American Mathematical Society, 115–133.

Benettin G., Galgagni L., Giorgilli A., and Strelcyn J.M. 1980. Lyapunov characteristic exponents for smooth dynamical systems and for Hamiltonian systems; a method for computing all of them. Part 1: Theory. Part 2: Numerical application. Meccanica 15: 9–30.

Berliner L.M. 1992. Statistics, probability and chaos. Statistical Science 7(1): 69–122.

Broomhead D.S. and King G.P. 1986. Extracting qualitative dynamics from experimental data. Physica D 20: 217–236.

Brown R., Bryant P., and Abarbanel H.D.I. 1991. Computing the Lyapunov spectrum of a dynamical system from an observed time series. Physical Review A. 43(6): 2787–2806.

Bühlmann P. 1997. Sieve bootstrap for time series. Bernoulli 3(2): 123–148.

Carlstein E. 1986. The use of subseries values for estimating the variance of a general statistic from a stationary sequence. The Annals of Statistics 14(3): 1171–1179.

Carlstein E., Do K., Hall P., Hesterberg T., and Künsch H.R. 1998. Matched block bootstrap for dependent data. Bernoulli 4: 305–328.

Casdagli M. 1992. Chaos and deterministic versus stochastic non-linear modeling. The Journal of the Royal Statistical Society B 54(2): 303–328.

Conover W.J. 1980. Practical Nonparametric Statistics. Wiley & Sons, New York. second ed., New York.

Coomes B.A., Kocak H., and Palmer K.J. 1995. Rigorous computational shadowing of orbits of ordinary differential equation. Numerische Mathematik 69: 401–421.

Corless R.M., Essex M., and Nerenberg M.A.H. 1991. Numerical methods can suppress chaos. Physics Letter A 157: 27–36.

Corless R.M. 1994. What good are numerical simulations of chaotic dynamical systems? Computers and Mathematics with Applications. An International Journal 28(10–12): 107–121.

Coven E.M., Kan I., and Yorke J.A. 1988. Pseudo-orbit shadowing in the family of tent maps. Transaction of the American Mathematical Society 308: 227–241.

Cutler C.D. 1993. A review of the theory and estimation of fractal dimension. *Technical report series*. **STAT-93-06**. Department of Statistics and Actuarial Science. University of Waterloo-Canada.

Dawson S., Grebogi C., Sauer T., and Yorke J.A. 1994. Obstructions to shadowing when a Lyapunov exponent fluctuates about zero. Physical Review Letters 73(14): 1927–1930.

Diamond P., Kozyakin V., Kloeden P., and Pokrovskii A. 1995. Computer robustness of semi-hyperbolic mappings. Random and Computational Dynamics 3(1–2): 57–70.

Eckmann J.P. and Ruelle D. 1985. Ergodic theory of chaos and strange attractors. Reviews of Modern Physics 57: 617–656.

Efron E. and Tibshirani R. 1986. Bootstrap method for standard errors, confidence intervals, and other measures of statistical accuracy. Statistical Science 1: 54–77.

Efron E. and Tibshirani R. 1993. An introduction to the bootstrap. Chapman & Hall, New York.

Farmer J.D. and Sidorowich J.J. 1988. Exploiting chaos to predict the future and reduce noise. In: Y.C. Lin (Ed.), Evolution, Learning and Cognition, World Scientific Publishing, 277–330.

Fraser A.M. and Swinney H.L. 1986. Independent coordinates for strange attractors from mutual information. Physical Review A 33(2): 1134–1140.

Gallant A.R. and White H. 1992. On learning the derivatives of an unknown mapping with multilayer feedforward networks. In: H. White (Ed.), Artificial Neural Networks. Approximation and Learning Theory. Blackwell Publication, Oxford, 206–223.

Geist K., Parlitz U., and Lauterborn W. 1990. Comparison of different methods for computing Lyapunov Exponents. Progress of Theoretical Physics 83(5): 875–893.

Gencay R. 1996. A statistical framework for testing chaotic dynamics via Lyapunov exponents. Physica D 89: 261–266.

Golia S. and Sandri M. 1997. Resampling chaotic time series. Physical Review Letters 78(22): 4197–4201.

Grassberger P., Hegger R., Kantz H., Schraffrath C., and Schreiber T. 1993. On noise reduction methods for chaotic data. Chaos 3(2): 127–141.

Grassberger P. and Procaccia I. 1983. Measuring the strangeness of strange attractors. Physica D 9: 189–208.

Hornik K., Stinchcombe M., and White H. 1992. Multilayer feedforward networks are universal approximators. In: H. White (Ed.), Artificial neural networks. Approximation and Learning Theory. Blackwell Publication. Oxford, pp. 12–28.

Kennel M.B., Brown R., and Abarbanel H.D.I. 1992. Determining embedding dimension for phase-space reconstruction using a geometrical construction. Physical Review A 45(6): 3403–3411.

Kloeden P.E. and Palmer K.J. (Eds.), 1994. Chaotic numerics: An international workshop on the approximation and computation of complicated dynamical behavior. July 12–16, 1993. Deakin University. American Mathematical Society. Contemporary Mathematics Series 172.

Künsch H.R. 1989. The jackknife and the bootstrap for general stationary observations. The Annals of Statistics 17: 1217–1241.

Kurths J., Schwarz U., Witt A., Krampe R.Th., and Abel M. 1995. Measure of complexity in signal analysis. Presented at 3rd Tecnical Conference on Nonlinear Dynamics (Chaos) and Full Spectrum—July 10–13, 1995.

Lai D. and Chen G. 1995. Computing the distribution of the Lyapunov exponent from time series: The one-dimensional case study. International Journal of Bifurcation and Chaos 6(5): 1721–1726.

Lall U. and Sharma A. 1996. A nearest neighbor bootstrap for resampling hydrologic time series. Water Resources Research 32(3): 679–694.

LeBaron B. and Weigend A.S. 1997. A bootstrap evolution of the effect of data splitting in financial time series. Working paper. IS-97-013. Leonard N. Stern School of Bussines, New York University, USA.

Lisi F., Nicolis O., and Sandri M. 1995. Combining singular-spectrum analysis and neural networks for time series forecasting. Neural Processing Letters 2(4): 6–10.

Lorenz E.N. 1989. Computational chaos—A prelude to computational instability. Physica D. 35: 299–317.

McCaffrey D., Ellner S., Gallant R., and Nychka D. 1992. Estimating the Lyapunov exponent of a chaotic system with nonparametric regression. Journal of the American Statistical Association 87(419): 682–695.

McCauley J.L. 1994. Chaos, Dynamics, and Fractals: An Algorithmic Approach to Deterministic Chaos. Cambridge University Press, Cambridge.

Mozer M.C. 1994. Neural net architectures for temporal sequence processing. In: A.S. Weigend and N.A. Gershenfeld (Eds.), Time Series Prediction: Forecasting the future and understanding the past, Addison-Wesley Publishing Company, pp. 243–264.

Nychka D., Ellener S., McCaffrey D., and Gallant A.R. 1992. Finding chaos in noisy systems. Journal of the Royal Statistical Society B. 54(2): 399–426.

Oseledec V.I. 1968. A multiplicative ergodic theorem. Lyapunov characteristic numbers for dynamical systems. Trans. Moscow Math. Society. 19: 197–231.

Pecora L.M., Carroll T.L., and Heagy J.F. 1997. Statistics for the continuity and differentiability: An application to attractor reconstruction from time series. In: D.C. Cutler and D.T. Kaplan (Eds.), Nonlinear Dynamics and Time Series. Building a Bridge Between the Natural and Statistical Sciences. American Mathematical Society, pp. 49–62.

Pilyugin S.Y. and Plamenevskaya O.B. 1999. Shadowing is generic. Topology and its Applications, 93(3): 253–266.

Politis D.N. and Romano J.P. 1994. The stationary bootstrap. Journal of the American Statistical Association 89(428): 1303–1313.

Reinfelds A. 1997. The shadowing lemma in a metric space. Universitatis Iagellonicae Acta Mathematica 35: 205–210.

Rosenstein M.T., Collins J.J., and De Luca C.J. 1993. A practical method for calculating largest Lyapunov exponent from small data set. Physica D. 65: 117–134.

Rosenstein M.T., Collins J.J., and De Luca C.J. 1994. Reconstruction expansion as a geometry-based framework for choosing proper delay times. Physica D. 73: 82–98.

Sauer T., Grebogi C., and Yorke J.A. 1997. How long do numerical chaotic solutions remain valid? Physical Review Letters 79: 59–62.

Sauer T. and Yorke J.A. 1991 Rigorous verification of trajectories for the computer simulation of dynamical systems. Nonlinearity 4: 961–979.

Simon J.L. and Bruce P. 1993. Probability and statistics with Resampling Stats and Mathematica. The Mathematica Journal 3(1): 48–55.

Takens F. 1981. Detecting strange attractors in turbulence. In: D.A. Rand and L.S. Young (Eds.), Dynamical systems and turbulence, Lecture Notes in Mathematics. 898 (Springer-Verlag, Berlino), 366–381.

Theiler J., Galdrikian B., Longtin A., Eubank S., and Farmer J.D. 1992. Using surrogate data to detect nonlinearity in time series. In: M. Casdagli and S. Eubank (Eds.), Nonlinear Modeling and Forecasting Addison-Wesley Publishing Company, pp. 163–188.

Thimm G. and Fiesler E. 1997. Pruning of neural networks. IDIAP Research Report. IDIAP-RR 97-03. Dalle Molle Institute for Perceptive Artificial Intelligence, Valais, Switzerland.

Tong H. 1995. A personal overview of non-linear time series analysis from a chaos perspective. Scandinavian Journal of Statistics 22(4): 399–445.

Vautard R., Yiou P., and Ghil M. 1992. Singular-spectrum analysis: A toolkit for short, noisy and chaotic signals. Physica D. 58: 95–16.

Yuan C. and Yorke J.A. 2000. An open set of maps for which every point is absolutely nonshadowable. Proceedings of The American Mathematical Society, 128: 909–918.

White H. 1992. Some asymptotic results for learning in single hidden layer feedforward networks. In: H. White (Ed.), Artificial Neural Networks. Approximation and Learning Theory. Blackwell Publication. Oxford, 135–157.

Ziehmann C., Smith L.A., and Kurths J. 1998. The bootstrap and Lyapunov exponents in deterministic chaos. Physica D, 126: 49–59.