Nonlinear Prediction, Chaos, and Noise

J. B. Elsner* and A. A. Tsonis+

Abstract

We present a brief overview of some new methodologies for making predictions on time-series data. These ideas stem from two rapidly growing fields: nonlinear dynamics (chaos) theory and parallel distributed processing. Examples are presented that show the usefulness of such methods in making short-term predictions. It is suggested that such methodologies are capable of distinguishing between chaos and noise. Implications of these ideas and methods in the study of weather and climate are discussed.

1. Introduction

One of the basic tenets of science is making predictions. If we know previous behavior, how can we predict future behavior? The approach in modern meteorology, like many sciences, requires two steps: construct a model based on theoretical considerations and use measured data as initial input. Since many of the underlying theoretical principles in meteorology are known, model construction has been and continues to be a primary area of research for meteorologists.

Today's numerical weather-prediction models for forecasting tomorrow's weather (also for climate prediction) solve a set of partial differential equations describing fluid flow over a rotating globe. The problem in prediction may not lie here. However, as was stated by Thompson (1957), significant problems may arise with the second step, where measured data are used as initial input to the model. Correct specification of initial state demands the measurements of variables in a three-dimensional volume. Routine measurements of relevant variables are taken at widely spaced locations providing only a discrete initial state. The spatially continuous differential equations simply cannot operate on discrete initial input (Farmer and Sidorowich 1987). Because of this inherent forecasting limitation in fluid-flow problems, we are motivated to try other approaches.

One class of alternative approaches is to build models directly from the available data. For these methods, the data, given as a time series, are usually considered as a single realization of a continuous random process (see, e.g., Pandit and Yu 1983). As Farmer and Sidorowich (1987) point out, this is appropriate when the randomness is a result of complex interactions involving many independent and irreducible degrees of freedom. Although linear methods of analyzing time series from weather and climate processes have had some success, especially in regard to relating cause and effect to physical phenomena, their predictive power is limited. The predictive limitation of linear methods is perhaps related to their inability to model feedback dynamics of the weather and climate systems (Farmer and Sidorowich 1988; hereafter referred to as FS88).

In the last decade, advances in the theory of dynamical systems have demonstrated the existence of dissipative systems whose trajectories that depict their asymptotic final states are not confined in limit cycles (periodic evolutions) or tori (quasi-periodic evolutions), but in sets of the total available phase space, which are not topological. These sets are fractal sets and are often called strange attractors. The corresponding dynamical systems are called chaotic systems and their trajectories never repeat. Thus, their evolution is aperiodic but completely deterministic. Because the evolution is aperiodic, any signal measured from a chaotic dynamical system "looks" quite irregular and exhibits frequency spectra with energy at all frequencies (broadband spectra) similar to those of random signals (see Tsonis and Elsner 1989 for a discussion of chaos and weather). Another important property of chaotic dynamical systems and their strange attractors is the divergence of initially nearby trajectories. Due to the action of the attractor, the evolution of the system from two (or more) nearby initial conditions will soon become quite different. Since the measurement of any initial condition is subject to some error, such a property imposes limits on long-term prediction. Nevertheless, for a short time, nearby trajectories may not diverge significantly, and thus, even though each evolution might be quite complex, knowledge of the dynamics and especially of the structure of the attractor (e.g., dimensions, Lyapunov exponents) may prove beneficial to the goal of short-term predictions.

For a system containing many irreducible degrees of freedom, the linear statistical approach is probably as good as any and may even be optimal (FS88). If, however, the irregular behavior is a result of low-

*Department of Meteorology, Florida State University, Tallahassee, FL 32306
+Department of Geosciences, University of Wisconsin-Milwaukee, Milwaukee, WI 53201

dimensional chaos, nonlinear models ought to be able to do much better at prediction than simple linear models. In fact, since chaos does not occur unless the system is to some extent nonlinear, nonlinear models are necessary to approximate chaotic dynamics.

The purpose of the present paper is to outline some recent advances in modeling time series and to demonstrate, through the application of a particular technique, their usefulness in making short-term predictions over standard autoregressive models. The paper is not intended to be definitive; rather, it serves as an interim report on time-series modeling efforts currently being explored in the physics and applied mathematical communities. No attempt is made to sort out the particular advantages and disadvantages of the various methods mentioned. More details concerning particular methods and applications are given in some of the references provided.

The problems of weather and climate forecasting offer a unique arena for testing and developing nonlinear prediction algorithms, not only because current numerical weather-prediction models are limited to some extent in their prediction capabilities, but because long-term reliable observational records have recently been made available for climate research. Diagnostic studies with these data utilizing nonlinear prediction schemes are a required step in the direction of understanding and quantifying the complexity of the global weather and climate systems.

2. Nonlinear prediction

The term "nonlinear prediction" covers a broad spectrum of methodologies. Our focus here is on dynamical state-space models. The two components of such models, determinism and state-space representation, can be considered separately. In fact, more familiar in meteorology are two variants of these models containing one or the other component. The analog method, suggested by Lorenz (1969), while not strictly limited to time-series data, is essentially a deterministic nonstate-space model. It is based on the idea of finding a historical weather pattern (analog) that closely resembles the current weather. The evolution of the historical analog provides a model for the evolution of the present weather. Although the method, as tried by Lorenz, was successful in estimating practical limits on atmospheric predictability, it was largely unsuccessful for operational forecasting due to the lack of an adequate history of large-scale weather patterns. With longer data archives now available and a focus on limited areas, the analog method has reemerged and appears to hold promise for weather forecasting (e.g., Van den Dool 1989; Toth 1989).

Another class of related nonlinear models are the threshold autoregressive (AR) models (Tong and Lim 1980; Tong 1983). These models rely on a state-space representation but are essentially statistical, having a deterministic component limited to a single variable. While this is a considerable improvement (for modeling chaos) over the strictly statistical linear representation of the traditional AR models, it may not provide enough nonlinearity for geophysical signals in general. Recently, Zwiers and von Storch (1990) have shown that such models are quite useful for modeling the Southern Oscillation.

Building a dynamical state-space model from time-series data requires two steps: finding an appropriate state-space reconstruction, and then choosing a nonlinear representation that maps visited regions into regions not yet visited in the reconstructed space. The state space can be replaced by the phase space using the method of delays. This is done by taking a scalar time series $x(t_i)$ and its successive time shifts as coordinates of a new vector time series given by

$$\mathbf{X}(t_i) = \{x(t_i), x(t_i+\tau),\ldots, x(t_i + (n-1)\tau)\}, \qquad (1)$$

where n is the dimension of the vector $\mathbf{X}(t_i)$ and τ is a time delay taken to be some suitable multiple of the sampling time Δt (see Packard et al. 1980; Takens 1981). Thus, for an n-dimensional phase space, a "cloud" of points will be generated. From this cloud the various dimensions and exponents can be calculated. The proper choice of τ to obtain a suitable reconstruction has been the subject of considerable debate. In principle, τ can be any length. However, if it is too small, then, in general, $x(t_i)$ will be nearly equal to $x(t_i + \tau)$ and not enough separation will exist between the chosen coordinates. If the dynamics take place on an attractor of dimension N, then it is necessary for determinism that $n \geq N$ (i.e., the attractor must be embedded in at least its dimension, otherwise it fills the embedding space, thus behaving like a random process) (FS88). For proper reconstructions, Takens (1981) showed that $n \sim 2N + 1$ is sufficient at least in principle.

There are other ways to construct a phase space. The use of derivatives, whereby the coordinates of the phase space are successively ordered higher derivatives instead of discrete time shifts, is an alternative. In fact, this is the underlying intuitive concept of the method of delays, but it is not recommended in practice, except for perhaps extremely clean data, since differentiation amplifies noise. A better alternative is the method suggested by Broomhead and King (1986), whereby the Karhunen–Loeve principal value decomposition is applied to the vector time series in Eq. (1). The procedure, called singular-spectrum analysis

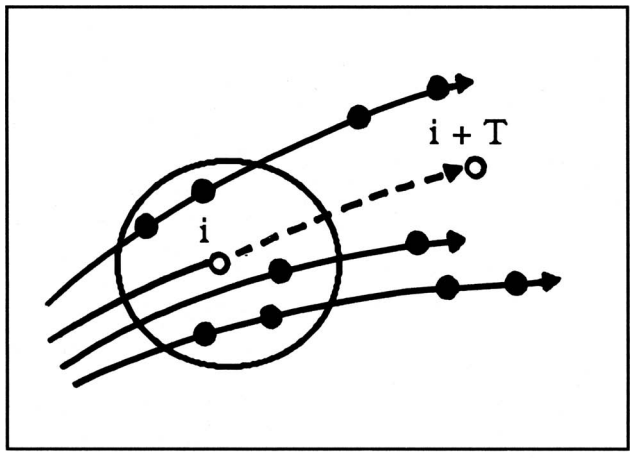

FIG. 1. An illustration of how local approximation works (after FS88). The present state $x(i)$ and its unknown future value $x(i + T)$ are represented by open circles. The black dots inside the circle define the neighborhood of $x(t)$ in this hypothetical state space. To make a prediction, we determine an appropriate mapping that takes the points in the neighborhood to states they move to a time T later, and then evaluate the mapping.

(SSA), is to extract the eigenvalues and eigenvectors of the covariance matrix of the vector series. The eigenvalues are the root-mean-square projection of the n-dimensional delay coordinate time series onto the orthogonal eigenvectors and, thus, represent a natural means of resolving the cloud of points in a higher dimensional space. In addition to providing a proper reconstruction of state space, SSA appears to be a useful tool for separating signal from noise in natural time series (Vautard and Ghil 1989; Ghil and Vautard 1991; Elsner and Tsonis 1991).

In any case, once we have reconstructed the attractor we can begin to think how we will improve short-term prediction. If an underlying deterministic mechanism exists, then the order with which the points appear in the cloud will also be deterministic. Thus, we may be able to somehow extract the rules that determine where the next point will be located in the phase space, and hence obtain a very accurate prediction. For example, consider the sequence $x(t)$: 0.12, 0.4224, 0.4759128, 0.094028, 0.3407468, 0.8985536, 0.36462, 0.9266888, ... If we plot $x(t)$ versus $x(t + 1)$, we find that the points fall on a very well-defined parabola, the expression of which we can easily find. Thus, $x_{n+1} = f(x_n)$ can be estimated and used for predictions [the reader may recognize this sequence as the logistic map $x_{n+1} = 4x_n(1 - x_n)$].

The methodology can be conceptualized by considering Fig. 1, where portions of a trajectory are shown in state space and a terminal point (present state) is denoted by an open circle. The solid circles indicate neighbors of the current state, and the arrowheads indicate movement of the neighbors through a local section of the embedding space. By finding a suitable function (linear or nonlinear) that describes how the neighbors advance, a prediction for the current state can be made. This is called local approximation, as compared with global approximation, which amounts to finding a representation over the entire attractor.

As FS88 point out, finding an appropriate functional representation at this point, either local or global, is largely a matter of trial and error. Some of the ideas currently being explored include polynomials (Farmer and Sidorowich 1987), radial basis functions (Casdagli 1989), or simpler variants (Sugihara and May 1990; Linsay 1991). Another such approach, which we will discuss in more detail, employs what are known as artificial neural networks. Falling within a class known as connectionist models, artificial neural networks or simply neural nets are mathematical models formulated and solved on conventional digital computers. They were originally designed to mimic some functions of the nervous system of animals or humans and can be viewed as another type of functional representation of neighbors in state space. We stress that despite the promise of such prediction methods, work along these lines is still in its infancy and no single method has emerged as fundamentally superior.

3. Neural networks

Let us introduce the philosophy behind neural networks by presenting a highly simplified example. This example is a modification of an example presented by Owens and Filkin (1989). Let us consider global precipitation over the past five years, with the precipitation for 1990 being what we are interested in predicting. Under such an arrangement, we say that we have one training pair consisting of the five inputs, $p[i]$, and a single output node, Q. The relationship between the inputs and the output is shown in Fig. 2. Such a figure is often referred to as the architecture of the network and for this example consists of two layers, an input layer and an output layer. The five inputs can be thought of as a five-component state vector, with the value of each component given as the amount of precipitation for the year. Usually, the inputs are scaled to the range $0 \leq p[i] \leq 1$. The value P is constructed as the inner product given by the sum of the inputs multiplied by their corresponding connection weights $w[i]$:

$$P = \Sigma w[i]p[i]. \qquad (2)$$

The summation in the equation is over the five inputs. The output Q is obtained by passing the inner product

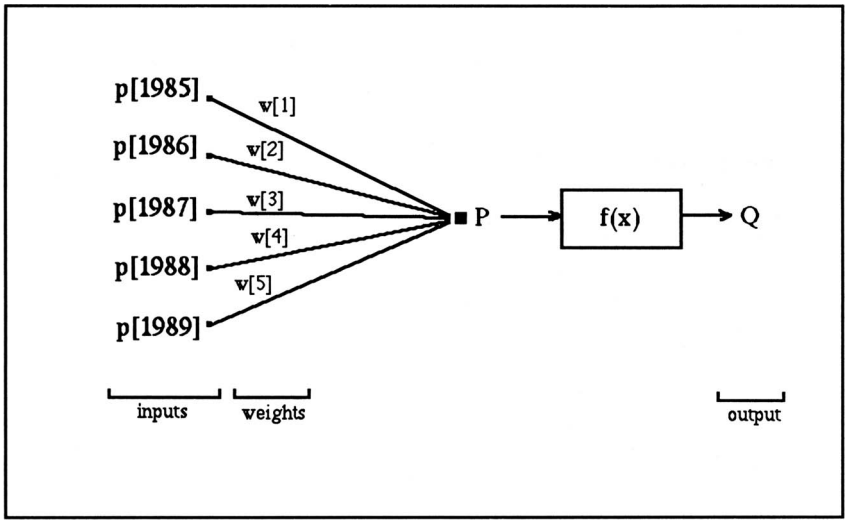

FIG. 2. Schematic (architecture) of a two-layer neural network. Each of the five inputs has a value $p[\,]$ corresponding to the amount of global precipitation for that year. The output has a value corresponding to the amount of global precipitation for 1990. The weights ($w[\,]$) indicate the relative strength of connections between inputs and the output. The input values are combined with the weights by an inner product to give a value P. Inputs are taken to outputs using a nonlinear squashing function such as tanh.

P through a *nonlinear* function $f(x)$, sometimes called the squashing function. The squashing function has limits $0 \leq f(x) \leq 1$, which guarantees that the output Q is limited in range regardless of the value of P. For this example, we have a single input–output training pair denoted as (P, Q). All the connection weights are then varied to minimize the squared error, calculated as the difference between the network's predicted output and the actual value.

In our simple, one-pair, two-layer network shown in Fig. 2, the training pair associates one specified set of inputs, for example, p [1985], p [1986], p [1987], p [1988], p [1989], with a single output Q. The error to be minimized is the squared difference between the actual value, p [1990], and the network value, Q:

$$E = (p\,[1990] - Q)^2. \tag{3}$$

The weights are changed by first finding the gradient of E with respect to $w[i]$ and then adjusting $w[i]$ to force E toward smaller values. This is accomplished with the help of a forward-Euler integration scheme:

$$w[i]^{n+1} = w[i]^n + \eta \Delta w[i],$$

where

$$\Delta w[i] = -\delta E/\delta w[i]. \tag{4}$$

$w[i]^n$ is the weight at iteration number n, and η is the learning rate. This is analogous to finding the root of a polynomial using a generalized Newton's method, whereby convergence to a root is achieved by successive evaluations of the function and its derivative. For the case of more than one training pair, the equation is generalized by summing over all training pairs. Training occurs in discrete iterations, with each iteration requiring one presentation of all training pairs to the network. The network "learns" by presenting the (P, Q) pair sequentially with a number of training pairs relating the values of the input to a corresponding value of the output. To ensure convergence of the integration scheme, the learning rate η must be small. However, using a stiff integration technique can greatly improve the learning rate (Owens and Filkin 1989).

In general, neural network programs are built around the concept of adjustable weights that take inputs to outputs. Each weight carries information that indicates how strongly the input is connected to the output. We can now proceed in presenting a formal definition of neural networks. A simple neural network model can be written as

$$x[i] = x(t - i\tau')$$
$$z = f(\Sigma w[i]x[i]),\ i = 1,n, \tag{5}$$

where $f(x)$ is a nonlinear sigmoidal function such as the hyperbolic tangent and where the $x[i]$'s are the inputs, which in effect form the coordinates of a state space. The parameter τ' is usually taken equal to 1, but it can assume other values as well. Note that the dimensionality of this space is equal to the number of inputs, n. Thus, in the example above, the dimensionality is 5. If we had used only 4 years prior to 1990, then the dimensionality would have been 4. In a way, the number of inputs defines an embedding dimension. This embedding dimension, however, may not be the same as the embedding dimension of the Takens theorem. While intuitively it seems that those two embedding dimensions should be related, up to this point this relation has not been established. In practice, one uses as many inputs as it takes in order to obtain the desired results. Therefore, even though the a priori knowledge of the dimension of the underlying attractor may suggest a first guess for the number of inputs, its exact value is not required.

Starting with arbitrary values for the weights ($w[i]$) an output (z) is calculated and then compared with the

actual value $x(t_i + T)$. The squared error between the model output and the actual value, given by

$$E = [x(t_i + T) - z]^2, \quad (6)$$

is subsequently used to change the weights. This is done by first calculating the derivatives of the error with respect to all the weights ($\delta E/\delta w[i]$). Then, if increasing a given weight leads to more error, the weight is adjusted downward. Otherwise, if increasing the weight leads to less error, the weight is adjusted upward. Since information about the error at the output layer is used to modify the weights at the input layer, the method is called "back propagation" (Rumelhart et al. 1986). The procedure is continued until all the weights settle down and the error converges to below some prescribed tolerance. Commonly, the initial weights are chosen as uniformly distributed random numbers; however, if prior information exists, a better initial guess can be made (Werbos 1990).

Often, when the system of interest is sufficiently complex (involving many degrees of freedom), a second layer, called a *hidden layer,* is added to the network (see Fig. 3). The multilayer neural network can then be written as

$$\begin{aligned} x[i] &= x(t - i\tau') \\ y[j] &= f(\Sigma w[i,j]x[i]) \\ z &= f(\Sigma w[j]y[j]). \end{aligned} \quad (7)$$

Connection weights are specified between input and hidden values ($w[i, j]$) and between hidden values and the output ($w[j]$). The weight-modifying scheme described above is applied in the same manner to the multilayer network.

In summary, with the help of known outputs, the network, initially set to a randomly chosen state, modifies its structure (changes the weights) in such a way as to improve its performance over the training set. If the network architecture is rich enough (i.e., sufficient number of both inputs and layers), this procedure will eventually lead the network to a state in which inputs are correctly mapped to outputs for all chosen training pairs (Keeler 1990).

It should be emphasized here that neural networks, along with the other dynamical state-space models, are phenomenological in that they assess the qualitative characteristics of the underlying system's dynamics and make short-term predictions based on that knowledge without providing a physical understanding of the mechanisms that might be operating within the system. However, successful predictions with such models can lead to useful hypotheses concerning, for example, why certain inputs are associated with stronger connection weights compared to others—which can readily be interpreted as a hypothesis concerning the physical nature of the system. Those that appreciate the above views may also appreciate the views of Wiener (1956, 1961).

4. Examples

In this section we present three examples showing the effectiveness of using a neural network for making predictions on time-series data. Each example uses a different dataset. For the first example, we use data generated artificially; for the second example, we use data generated from a controlled laboratory experiment; and for the third example, we use data observed in nature. The neural network architecture we employ for each example consists of three layers: input, hidden, and output. Learning is achieved using the method of back propagation, as was discussed in the previous section. Training is performed on the first part of the time series, with subsequent predictions made

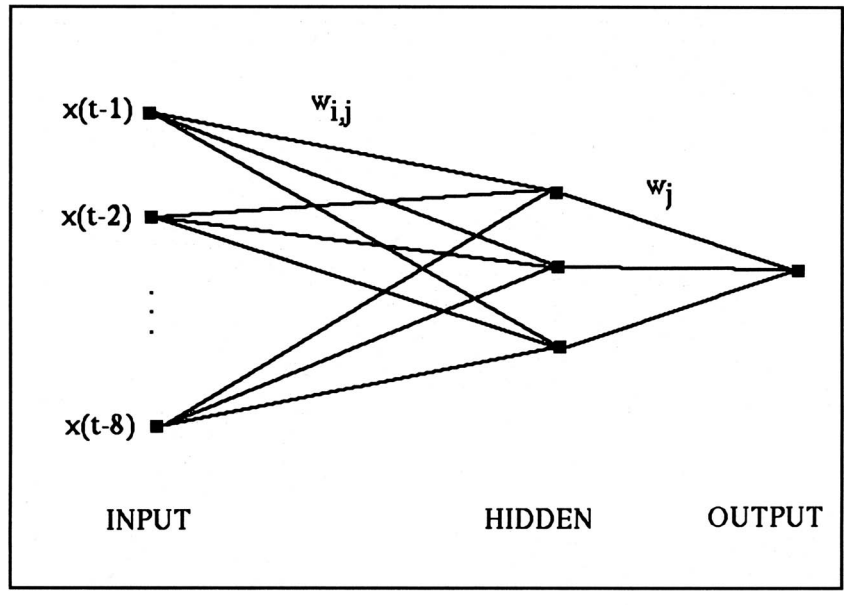

FIG. 3. Architecture of the neural network used in this study. The single output corresponds to the fact that we are making predictions one step into the future. The number of hidden values is set at three. Results from numerous trial runs indicated that adding more hidden values did not significantly improve the network's prediction capabilities. The number of inputs varied for the different examples presented. Again, however, the model was not sensitive to small changes in the number of inputs used. The values of the input nodes are lagged values of the time series representing a reconstructed state space.

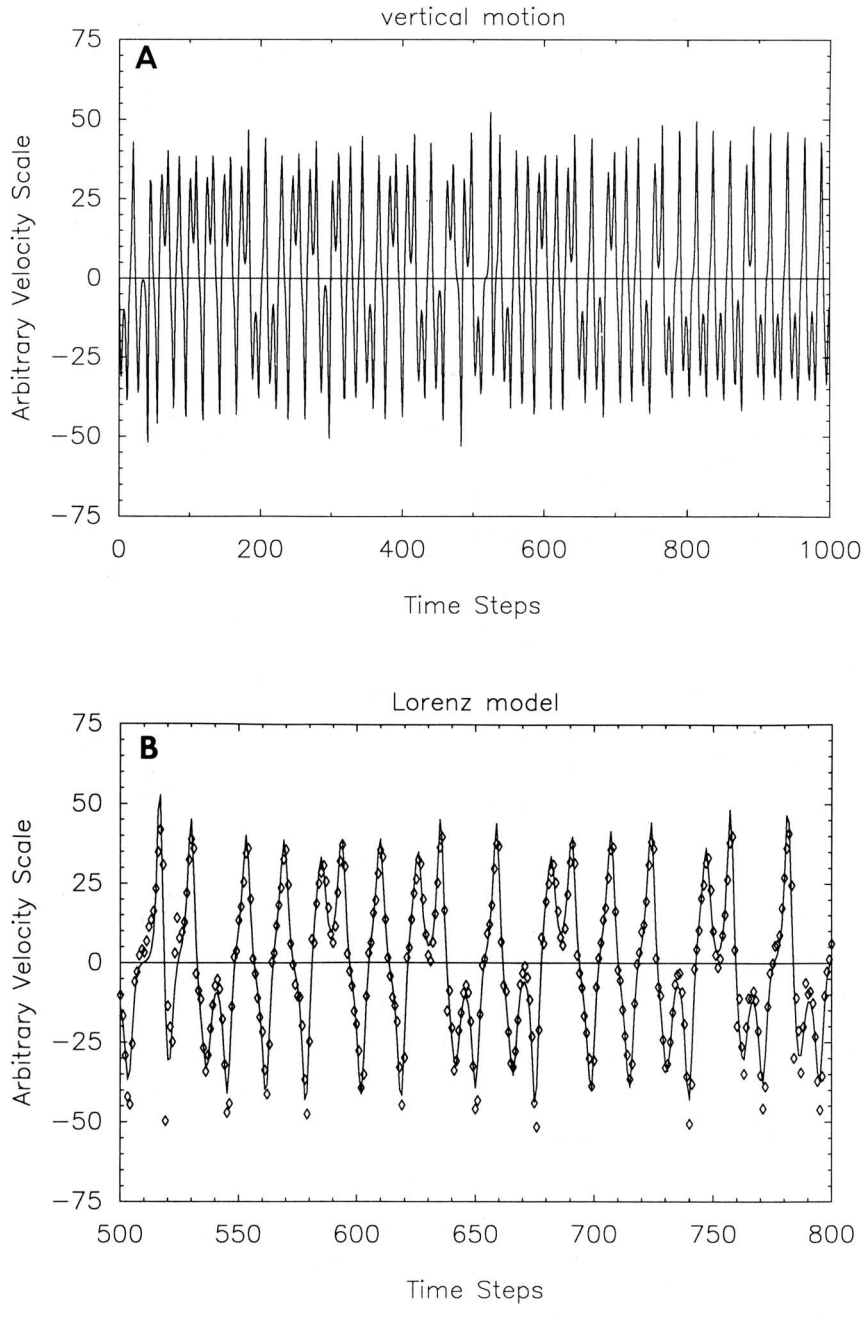

FIG. 4. (a) Time series of convective motions, after all transients have died, generated by numerically integrating the Lorenz system using a fourth-order Runge–Kutta scheme. The time axis is in integration steps and the magnitude of convection is on an arbitrary velocity scale. The series displays chaotic oscillations. (b) Comparison of the actual time series (solid line) with a neural network prediction (points). The number of inputs in the neural network is 8. The actual time series represents a part of a novel portion (second half) of the convective signal. Predicted values correspond quite well with actual values.

on the remaining values. For each example, the number of outputs is set at one, and the number of hidden values is set at three, while the number of inputs depends on the individual example. Numerous trial runs indicated that the accuracy of prediction was not sensitive to small changes in the number of inputs or hidden values. The single output represents some future value of the time series we wish to predict.

The neural network architecture is shown in Fig. 3. The inputs are the components of a reconstructed n-dimensional state space consisting of successive time-delayed values of the series. The method is similar to the one used by Perrett and van Stekelenborg (1990) to predict annual sunspot numbers. For example, if we represent the series as $x(t_i)$, where $i = 1, 2, ..., L$, then with $\tau' = 1$, and using an 8-dimensional phase space (i.e., eight inputs) beginning with the first value of the time series, the first set of inputs is $\{x(1), x(2), ..., x(8)\}$ and the output we are trying to predict is $x(9)$. Similarly, the second set of inputs is $\{x(2), x(3), ..., x(9)\}$ and the output we are trying to predict is $x(10)$. Training continues over all training pairs (set of inputs, output) for several thousand iterations.

For the first example, we generate a time series by numerically integrating the Lorenz system (Lorenz 1963) consisting of three ordinary differential equations describing convection of a fluid, warmed from below in time. The system is given as

$$\begin{aligned} dx/dt &= ax + ay \\ dy/dt &= -xz + bx - y \\ dz/dt &= xy - cz, \end{aligned} \qquad (8)$$

where x is proportional to the intensity of convective motion, y is proportional to the horizontal temperature variation, z is proportional to the vertical temperature variation, and a, b, and c are constants. For a choice of constants corresponding to sufficient heating, the convection will exhibit chaos. We use a fourth-order Runge–Kutta integration scheme and constants $a = 16.0$, $b = 120.1$, and $c = 4.0$. The time series of convective motion (x component of the system), after all transients (104 iterations) have diminished, is shown in

Fig. 4a. Positive values indicate upward motion in the fluid. We take 1000 values from the time series, train the network on the first 500 values, and make predictions on the last 500 values. The number of inputs in the network was eight. Results of the neural network at predicting one step into the future (points) compared with the actual values (solid line) are given in Fig. 4b. The normalized root-mean-square error (RMSE) between the actual and predicted values is 0.072, where zero implies a perfect forecast. Clearly, the network is capable of capturing the underlying chaotic dynamics of the system (see also Frison 1990a).

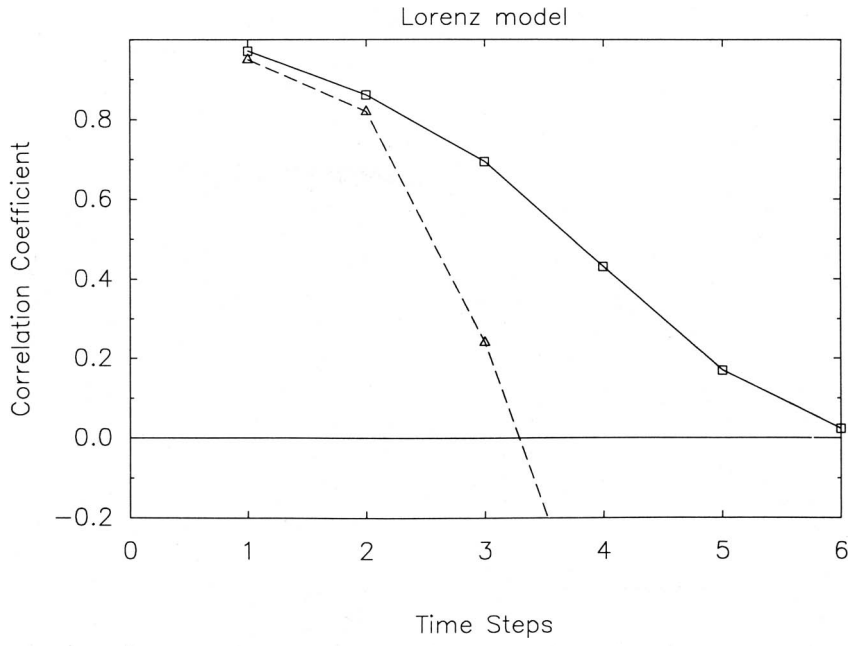

FIG 5. Correlation coefficients between actual values and predicted values as a function of prediction time for the convective motions using a neural network model (solid line) and using an optimum autoregressive model (dashed line). Prediction time is given as discrete time steps into the future. A correlation coefficient of 1 corresponds to perfect prediction. The neural network model clearly outperforms the autoregressive model.

To assess the predictive ability of the neural network against that of a standard statistical model, we fit the first half of the time series using an optimum autoregressive process and then compare predictions on the second half of the series from both models. For the autoregressive (AR) model, the time series is viewed as a single realization of a stochastic process, which is taken to be stationary and having a Gaussian distribution. For model selection we employed the Bayesian Information Criteria as outlined in Katz (1982) and determined that the optimum order of the AR model for the time series is 12.

Comparisons between the neural network and AR models are made by quantifying how the prediction accuracy (skill) decreases as predictions are made further into the future. To do this we make a prediction one step into the future and then use this predicted value as one of the lagged inputs for the next prediction, two time steps into the future. Similarly, the prediction at this second time step, as well as the previous time step, are used as lagged inputs for the next prediction, three time steps into the future. Doing this successively allows us to compute the correlation coefficient between actual and predicted values as a function of prediction time, where prediction time is given as discrete time steps into the future. The correlation coefficient between actual and predicted values is defined in the standard statistical way and is widely used as a measure of predictive skill (Anthes 1984). This procedure is followed for both the neural network model and for the optimum AR model.

Results are shown in Fig. 5. For the first few steps into the future, predictions from both models are good and the difference between the two models in terms of predictive skill is small. In contrast, the neural network makes significantly better forecasts than does the AR model as prediction time increases. The predictive skill on a nonuniform chaotic attractor will vary in time (Nese 1989); however, by using the same segment of the attractor to compare the models, as was done here, we ensure a fair comparison. We note that the AR model is essentially a linear model and therefore incapable of capturing the inherent nonlinear nature of such a record. Since the signal is, in fact, chaotic, we cannot hope to make accurate predictions with any model too far into the future. As we see, the predictive skill of the neural network also drops to near zero after a relatively short time.

Results similar to the above conclusions are obtained when the y or z component of the system is considered. This is not surprising, since every component is the result of the global dynamics of the system.

We next turn our attention to data generated from a controlled fluid dynamics experiment. The data were recorded from a rotating, differentially heated annulus of fluid. The experiments were performed at the Geophysical Fluid Dynamics Institute (GFDI) to study the transition to turbulence in fluids. The experiment from which the data were taken is described in detail by Pfeffer et al. (1980a,b). For our purposes it is sufficient

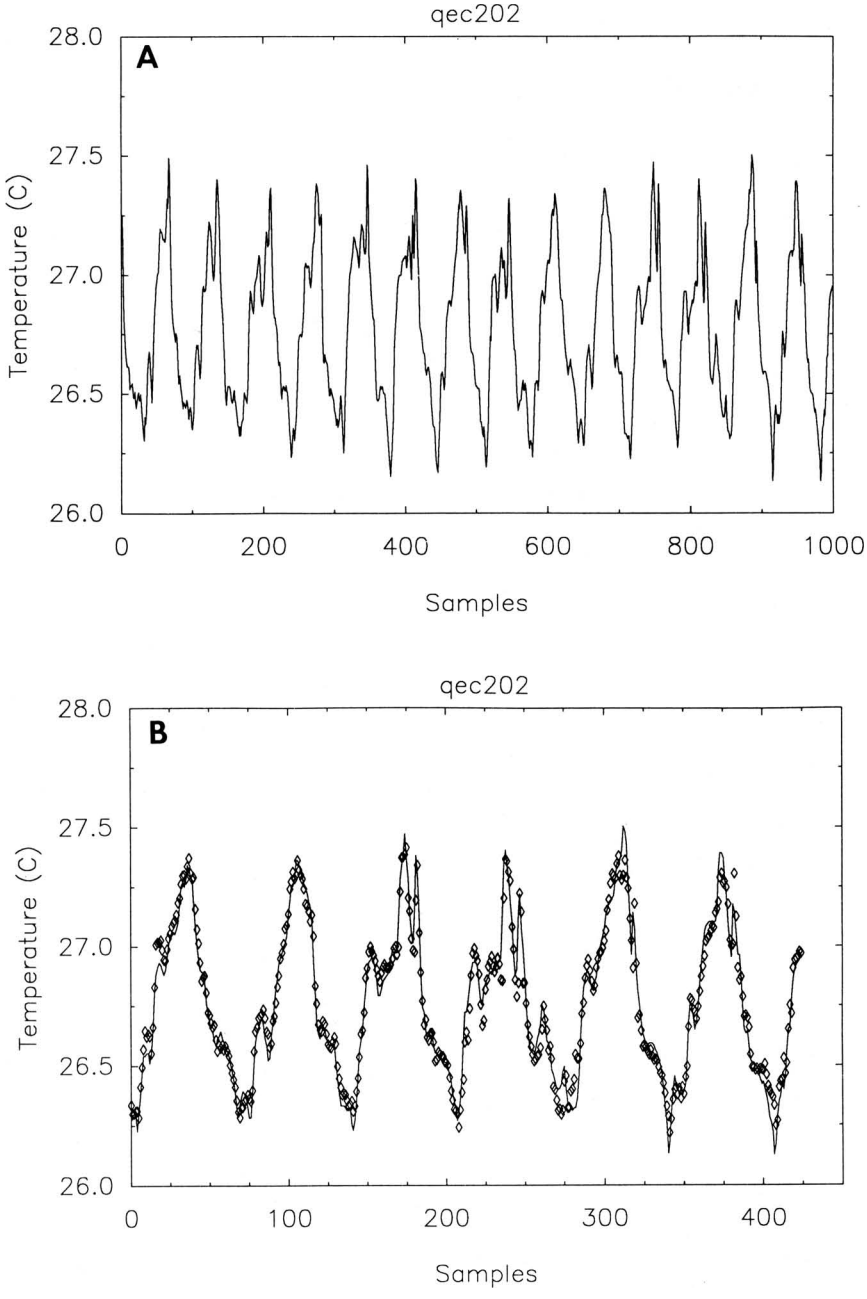

FIG. 6. (a,b) Same as figure 4(a,b), except that the time series represents temperatures (°C) taken from a rotating, differentially heated fluid in an annulus. The record is taken at middepth in the fluid. The time axis is given in number of rotations times two. The experiment was performed at the Geophysical Fluid Dynamics Institute. The number of inputs in the model is 75. The neural network makes excellent predictions one step into the future.

to say that the data recorded in time series represent temperatures in degrees C at a single location near middepth in the fluid. The temperature contrast from the inner to outer wall of the annulus is held constant at 10°C. Sampling rate is once every two rotations, with each rotation analogous to one sidereal day.

As was done previously, we take 1000 values from the time series, train the neural network on the first 500 values, and make predictions on the last 500 values. The complete record is shown in Fig. 6a and a forecast one step into the future is shown in Fig. 6b. The normalized RMSE between actual and predicted values one step into the future is 0.065, indicating very good predictions. Shown in Fig. 7 is the correlation coefficient between actual and predicted values as a function of prediction time for both the neural network and an optimum fourth-order AR model. As was seen previously, for the first few time steps into the future predictions from both models are good and the differences between the two models in terms of predictive skill is small. After that, however, the neural network clearly outperforms the linear AR model. Here the number of inputs in the model was 75.

For the third example, we take a time-series record of sea surface temperatures in degrees C, constructed by proxy using deep-sea ice-core records of oxygen isotope concentrations. Data are available for the period of approximately 1700–700 thousand years (kyr) before present at a sampling rate of 2 kyr, for a total of 498 values (Ruddiman et al. 1986). Similar records have been used in climate research. The complete time series is shown in Fig. 8. We train the neural network using eight input values on the first 400 values and make predictions on the remaining 98 values. The RMSE between actual and predicted values one time step into the future is 0.170, indicating some skill. For comparisons we again employ an optimum fourth-order autoregressive model and compare correlation coefficients between actual and predicted values as a function of prediction time for both models (Fig. 9). As was the case with the previous two examples, the neural network forecast demonstrates considerably more skill than does the forecast using an AR model, especially after the first few time steps.

5. Spatial dynamics

The idea behind neural networks and other such nonlinear prediction methods is that if deterministic rules dictate the system, then, even if the behavior is chaotic, the future may to some extent be predictable from the behavior of the past states of the system that are similar to those of the present (Sugihara and May 1990). The concept of using a neural network to emulate complex or nonlinear processes can be extended to many types of problems beyond time-series prediction. As long as a good record of dynamical states is known for the period of interest, a neural network can learn to simulate the behavior of the system (Frison 1990b). One class of problems we are currently working on is the modeling of spatial dynamics of satellite imagery from hurricanes and mesoscale convective complexes (MCCs), where successive images represent past and present states of the system.

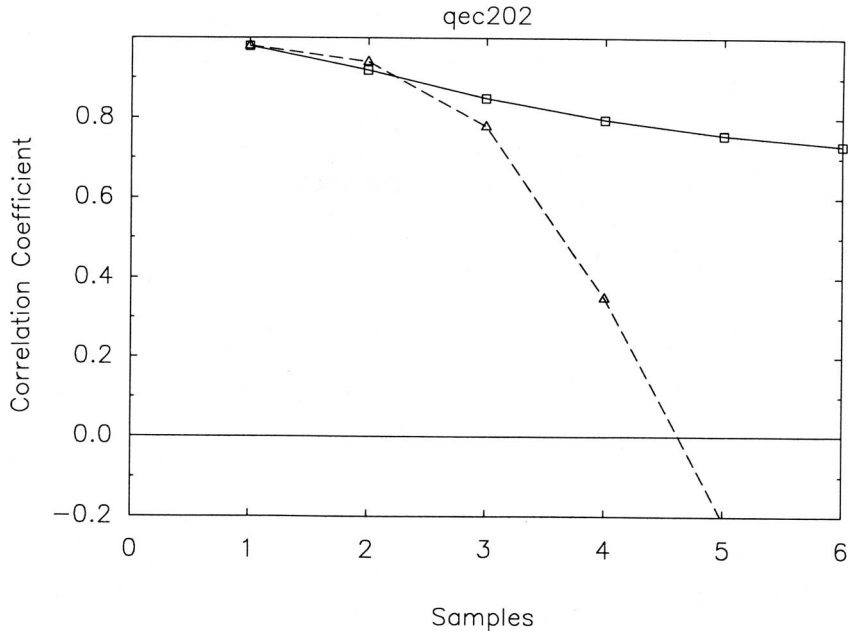

FIG. 7. Correlation coefficient as a function of prediction time as in Fig. 5, except for the laboratory experiment data. The solid line represents predictions made with the neural network and the dashed line represents predictions made with an optimum autoregressive model. Both models do well in the short term; however, the network clearly outperforms the linear autoregressive model as prediction time increases.

As noted by Maddox (1980), MCCs are convectively driven organized weather systems whose physics are not well understood, much less included in operational convective parameterization schemes despite the fact that they appear to be organized in a nonrandom manner and evolve on scales large enough to be resolved by current numerical forecast models. MCC events are all characterized by similar life cycles. Thus, regardless of the synoptic setting and internal structure of MCCs, satellite data exhibit a generalized consistent life cycle that reflects a meso-α–scale organization (McAnelly and Cotton 1989). This suggests the possibility of making use of past behavior of such systems, as viewed from space for example, to forecast future behavior, rather than relying solely on numerical forecast models.

Lee et al. (1990) have already demonstrated the usefulness of a neural network at distinguishing various cloud types from satellite imagery. Each satellite image of an MCC contains a wealth of information regarding the underlying atmospheric structure of the storm. A typical infrared image has a 4-km resolution, and each pixel stores a brightness value between 0 and 256 depending on the radiation emission. For any chosen brightness threshold, there will be areas on the image delineating regions above and below this level. With successive images in time, changes in size and shape of the regions represent an underutilized source of dynamical information about the storm evolution. By using pixel information as input, a neural network model can be trained to learn the spatial evolution of the storm as viewed from space, assuming we have data from a large number (e.g., 100; Packard 1990) of past storms.

6. Chaos and noise

Recently it has been suggested that certain nonlinear prediction techniques are capable of distinguishing between chaos and noise in time-series records (Sugihara and May 1990). We demonstrate next that neural networks share this capability by comparing results of the Lorenz system with results from a model trained on a time series generated from discrete points on a sine wave, having a unit amplitude, and adding to it at each step a uniformly distributed random variable in the interval [–0.5, 0.5]. Such a time series may display dynamical character similar to chaotic systems. Fourier analysis will result in spectra exhibiting peaks superimposed on a continuous background, and dimensional analysis may indicate anything from a low-dimensional system (if noise is weak) to a random signal (if noise is strong).

After training the neural network on the first half of

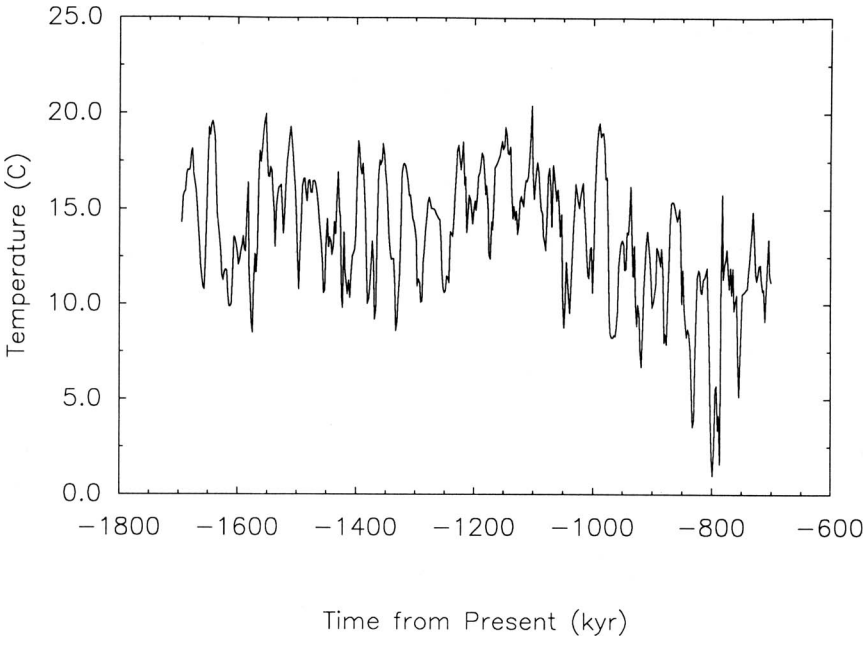

FIG. 8. Time series of proxy sea surface temperature in °C for the period 1700–700 thousand years (kyr) before present, at intervals of 2 kyr. The length of the record is 498 values. Similar records have been used in the study of climate dynamics.

how chaotic a system is. For example, one measure of the rate of degradation might simply be how many prediction steps are necessary before the correlation coefficient between actual and predicted values reaches some nearly asymptotic value. We note the loss of predictive skill of the neural network model on the proxy temperature record (Fig. 9) is suggestive of a chaotic signal. This result supports earlier evidence of deterministic chaos in climate (Nicolis and Nicolis 1984).

In applying a nonlinear theory (chaos) in the analysis of weather and climate data, one usually begins with estimating the dimension of the underlying attractor (e.g., Nicolis and Nicolis 1984; Fraedrich 1986; Essex et al. 1987; Tsonis and Elsner 1988; Sharifi et al. 1990) by reconstructing a state space from the time series and then applying some variant of the correlation algorithm (Grassberger and Procaccia 1983) on the set of points. The dimension, which is given by the power-law (scaling) behavior of the

the signal composed of a sine wave plus noise, we make predictions on the second half and, as was done with the Lorenz system, we compute the correlation coefficient between actual and predicted values as a function of prediction time. The dashed horizontal line in Fig. 10 is the result of this procedure. The independence of predictive skill with prediction length is in sharp contrast to the rapid decrease of predictive skill for the chaotic signal from the Lorenz system (solid line). From the differences, we suggest that predicting time series using neural networks is another method for differentiating additive noise from deterministic chaos (Elsner 1991). Predictions on time series with additive noise will appear to have a fixed amount of error, regardless of how far into the future one tries to predict. On the other hand, prediction accuracy on chaotic time series will degrade as one tries to predict too far into the future. It is suggested that it might be possible to quantitatively compare the rates of degradation in prediction skill as an indication of just

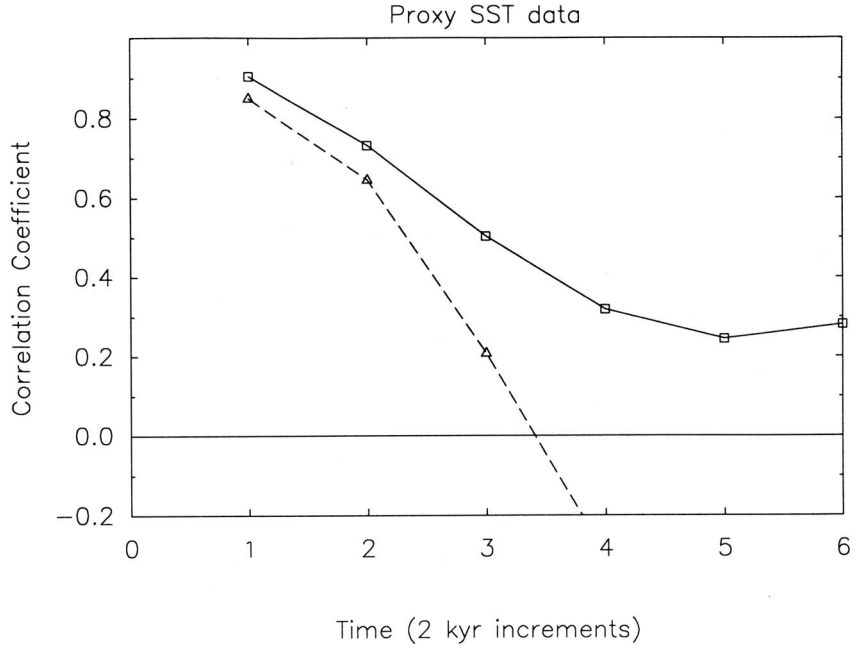

FIG. 9. Correlation coefficient as a function of prediction time for the data in Fig. 8. The solid and dashed lines represent predictions made with the neural network and with an optimum autoregressive model, respectively. Both models start well, but the autoregressive model cannot compete with the network as time increases.

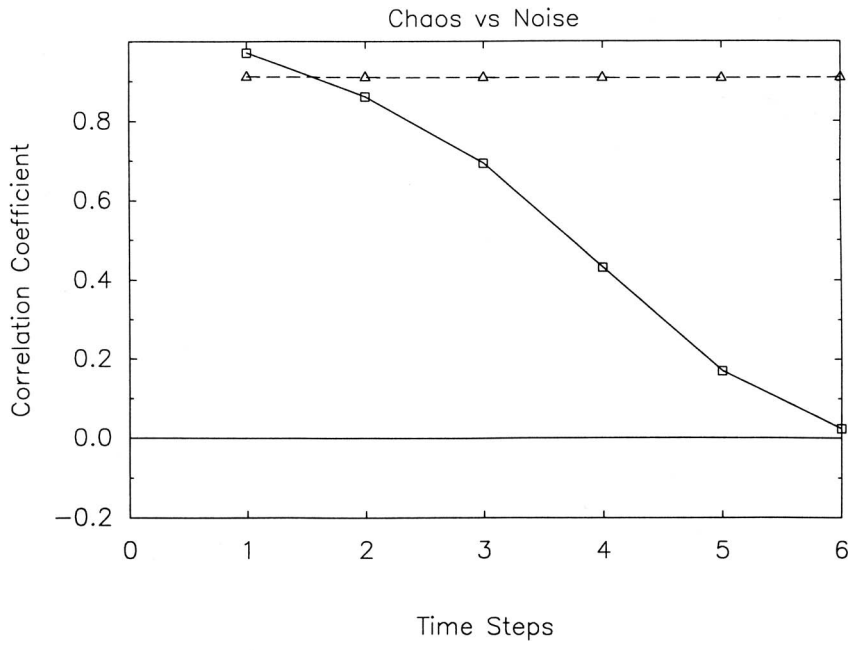

FIG. 10. Correlation coefficients between actual and neural-network predicted values for the Lorenz system (solid line) and for a signal consisting of a sine wave plus noise (dashed line). The rapid drop of the correlation coefficient with prediction time is a characteristic of chaotic signals. In contrast, the independence of predictive skill with prediction time of the sine wave-plus-noise signal demonstrates that the neural network is capable of distinguishing between additive noise and chaos.

correlation integral, gives a measure of the effective number of degrees of freedom of the system. Application of the algorithm, however, is subjected to problems like proper length of time series, proper τ, etc. Also, because the scaling regions used to estimate the dimension involve only a small number of distances between points in the state space, much of the information in the time series is lost, which for relatively short weather and climate records can cause serious problems (Sugihara and May 1990). In contrast, prediction methods like the one discussed here have the advantage that standard statistical procedures (such as correlation coefficients between actual and predicted values) can be used to evaluate their performance. And their performance should provide a more stringent test of underlying determinism in complex systems (Farmer and Sidorowich 1988; Casdagli 1989). Note that the underlying dimension could be useful information to the neural networks as well. An estimate may be obtained by the standard approaches mentioned above, or it may simply be guessed. In the case of neural networks, this estimate does not constitute a proof of chaos, but serves only as a guide in the training of the network. It is the improved predictions (if any) that provide proofs of underlying deterministic dynamics.

7. Conclusion

We have tried to demonstrate with examples ranging from mathematical models to controlled laboratory experiments to measured climate data that neural networks are capable of short-term predictions even if the underlying dynamics generating the data are chaotic. What Smolensky (1988) says about the human mind can be applied equally well to the atmosphere. That is, the rich behavior displayed by the atmosphere has the paradoxical character of appearing, on the one hand, tightly governed by a complex system of hard rules, and, on the other, awash with variance, deviation, exception, and a degree of flexibility that has eluded to some extent our attempts at simulation. Neural network models may be capable of demonstrating a greater degree of precision and accuracy in predicting and explaining some of the many vagaries of the atmosphere than do currently available models.

Acknowledgments. Thanks are extended to A. Owens of DuPont and J. Perrett and J. van Stekelenborg of the Bartol Research Foundation at the University of Delaware, who introduced us to neural networks as a model for time-series prediction and supplied the initial code. Thanks are also given to R. L. Pfeffer of the Geophysical Fluid Dynamics Institute for his assistance and discussions concerning some of the data used in this study. Part of the research was supported by NOAA Grant NA-16RC0454-01 and Air Force Grant AFOSR 90-0009.

References

Anthes, R. A., 1984: Predictability of mesoscale meteorological phenomena. *Predictability of Fluid Motions,* G. Holloway and B. J. West, Eds., American Institute of Physics, 247–270.

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

Casdagli, M., 1989: Nonlinear prediction of chaotic time series. *Physica D.,* **35,** 335–356.

Elsner, J.B., and A.A. Tsonis, 1991: Do bidecadal Oscillations exist in the global temperature record? *Nature,* **353,** 551–553.

Elsner, J. B., 1991: Predicting time series using a neural network as a method to distinguish chaos from noise. *J. Phys. A.,* in press.

Essex, C., T. Lookman, and N. A. H. Nerenberg, 1987: The climate attractor over short time scales. *Nature,* **326,** 64–66.

Farmer, J. D., and J. J. Sidorowich, 1987: Predicting chaotic time series. *Phys. Rev. Lett.,* **59,** 845–848.

——, and ——, 1988: Exploiting chaos to predict the future and reduce noise. Theoretical Division and Center for Nonlinear Studies, Los Alamos National Laboratory, LA-UR-88-901.

Fraedrich, K., 1986: Estimating the dimensions of weather and climate attractors. *J. Atmos. Sci.,* **432,** 419–432.

Frison, T., 1990a: Predicting nonlinear and chaotic systems behavior using neural networks. *J. Neural Net. Comp.,* **2,** 31–39.

——, 1990b: A general discussion on matching problems to paradigms. *J. Neural Net. Comp.,* **2,** 45–55.

Ghil, M., and R. Vautard, 1991: Interdecadal oscillations and the warming trend in global temperature time series. *Nature,* **350,** 324–327.

Grassberger, P., and I. Procaccia, 1983: Characterization of strange attractors. *Phys. Rev. Lett.,* **50,** 346–349.

Katz, R. W., 1982: Statistical evaluation of climate experiments with general circulation models: A parametric time series modeling approach. *J. Atmos. Sci.,* **39,** 1445–1455.

Keeler, J. D., 1990: A dynamical systems view of cerebellar function. *Physica D.,* **42,** 396–410.

Lee, J., R. C. Weger, S. K. Sengupta, and R. M. Welch, 1990: A neural network approach to cloud classification. *IEEE Trans. Geosci. Remote Sens.,* **28,** 846–855.

Linsay, P. S., 1991: An efficient method of forecasting chaotic time series using linear interpolation. *Phys. Lett. A.,* **153,** 353–356.

Lorenz, E. N., 1963: Deterministic nonperiodic flow. *J. Atmos. Sci.,* **20,** 130–141.

——, 1969. Atmospheric predictability as revealed by naturally occurring analogues. *J. Atmos. Sci.,* **26,** 636–646.

Maddox, R. A., 1980: Mesoscale convective complexes. *Bull. Amer. Meteor. Soc.,* **61,** 1374–1387.

McAnelly, R. L., and W. R. Cotton, 1989: The precipitation life cycle of MCCs over the central U.S. *Mon. Wea. Rev.,* **117,** 784–808.

Nese, J. M., 1989: Quantifying local predictability in phase space. *Physica D.,* **35,** 237–250.

Nicolis, C., and G. Nicolis, 1984: Is there a climate attractor? *Nature,* **311,** 529–532.

Owens, A. J., and D. L. Filkin, 1989: Efficient training of the back-propagation network by solving a system of stiff ordinary differential equations. Preprints, *International Conference on Neural Networks,* Washington, D.C., **2,** 381–386.

Packard, N. H., 1990: A genetic algorithm for the analysis of complex data. *Complex Systems,* **4,** 56–67.

——, J. D. Farmer, and R. S. Shaw, 1980: Geometry from a time series. *Phys. Rev. Lett.,* **45,** 712–716.

Pandit, S. M., and S. M. Yu, 1983: *Time Series and System Analysis with Applications.* Wiley, 272 pp.

Perrett, J. C., and J. T. P. van Stekelenborg, 1990: A neural network as a model for the prediction of sunspot numbers. Bartol Research Foundation, University of Delaware, Neward, Delaware, 41 pp.

Pfeffer, R., G. Buzyna, and R. Kung, 1980a: Time-dependent modes of behavior of thermally driven rotating fluids. *J. Atmos. Sci.,* **37,** 2129–2149.

——, ——, and ——, 1980b: Relationships among eddy fluxes of heat, eddy temperature variances and basic-state temperature parameters in thermally driven rotating fluids. *J. Atmos. Sci., 37,* 2577–2599.

Ruddiman, W. F., M. Raymo, and A. McIntyre, 1986: Matuyama 41 000-year cycles: North Atlantic and Northern Hemisphere ice sheets. *Earth and Planet Sci. Lett.,* **80,** 117–129.

Rumelhart, D. E., G. E. Hinton, and R. J. Williams, 1986: Learning representations by back-propagating errors. *Nature,* **323,** 533–536.

Sharifi, M. B., K. P. Georgakakos, and I. Rodriguez-Iturbe, 1990: Evidence of deterministic chaos in the pulse of storm rainfall. *J. Atmos. Sci.,* **47,** 888–893.

Smolensky, P., 1988: On the proper treatment of connectionism. *Behavior and Brain Sciences,* **11,** 1–74.

Sugihara, G., and R. M. May, 1990: Nonlinear forecasting as a way of distinguishing chaos from measurement error in time series. *Nature,* **344,** 734–741.

Takens, F., 1981: Detecting strange attractors in turbulence. Dynamical Systems and Turbulence, *Lecture Notes in Math. 898.* Springer, 366–381.

Thompson, P. D., 1957: Uncertainty of initial state as a factor in the predictability of large-scale atmospheric flow patterns. *Tellus,* **9,** 275–295.

Tong, H., 1983: *Threshold Models in Nonlinear Time Series Analysis.* Springer-Verlag, 323 pp.

——, and K. S. Lim, 1980: Threshold autoregression, limit cycles and cyclic data. *J. Roy. Stat. Soc. B.,* **42,** 245–292.

Toth, Z., 1989: Long-range weather forecasting using an analog approach. *J. Climate,* **2,** 594–607.

Tsonis, A. A., and J. B. Elsner, 1988: The weather attractor over very short time scales. *Nature,* **333,** 545–547.

——, and ——, 1989: Chaos, strange attractors, and weather. *Bull. Amer. Meteor. Soc.,* **70,** 14–23.

van den Dool, H. M., 1989: A new look at weather forecasting through analogues. *Mon. Wea. Rev.,* **117,** 2230–2247.

Vautard, R., and M. Ghil, 1989: Singular spectrum analysis in nonlinear dynamics, with applications to paleoclimatic time series. *Physica D.,* **35,** 395–424.

Werbos, P. J., 1990: Back-propagation through time: What it does and how to do it. *Proc. IEEE,* **78,** 1550–1560.

Wiener, N., 1956: Nonlinear prediction and dynamics. *Proc. of the Third Berkeley Symposium,* J. Neyman, Ed., University of California Press, Berkeley, 247–252.

——, 1961: *Cybernetics.* The MIT Press, 212 pp.

Zwiers, F., and H. von Storch, 1990: Regime-dependent autoregressive time-series modeling of the Southern Oscillation. *J. Climate,* **3,** 1347–1363.