Physica D 57 (1992) 1–30
North-Holland

PHYSICA D

# An analytic approach to practical state space reconstruction

John F. Gibson[1,2], J. Doyne Farmer[2], Martin Casdagli[3] and Stephen Eubank[2]

*Complex Systems Group, Los Alamos National Laboratory, Los Alamos, NM 87545, USA*
*and Santa Fe Institute, 1660 Old Pecos Trail, Suite A, Santa Fe, NM 87501, USA*

Received 29 October 1991
Revised manuscript received 14 February 1992
Accepted 14 February 1992
Communicated by P.E. Rapp

We study the three standard methods for reconstructing a state space from a time series: delays, derivatives, and principal components. We derive a closed-form solution to principal component analysis in the limit of small window widths. This solution explains the relationship between delays, derivatives, and principal components, it shows how the singular spectrum scales with dimension and delay time, and it explains why the eigenvectors resemble the Legendre polynomials. Most importantly, the solution allows us to derive a guideline for choosing a good window width. Unlike previous suggestions, this guideline is based on first principles and simple quantities. We argue that discrete Legendre polynomials provide a quick and not-so-dirty substitute for principal component analysis, and that they are a good practical method for state space reconstruction.

## 1. Introduction

State space reconstruction is the creation of a multidimensional, deterministic state space from a lower dimensional time series. It is a prerequisite step for analyzing a time series in the language of dynamical systems or for making predictive state space models. In the statistics literature, the idea of state space reconstruction is quite old [1]. It was introduced into dynamical systems theory independently by Packard et al. [2], Ruell[#1], and Takens [3]. The important contribution from dynamical systems was the demonstration that reconstructed state spaces can preserve geometrical invariants, such as the eigenvalues of a fixed point, the fractal dimension of an attractor, and the Lyapunov exponents of a trajectory. This was demonstrated numerically by Packard et al. and proven by Takens.

Takens proved that in the absence of noise it is always possible to embed a time series in a state space. When the dimension is sufficiently high, a reconstruction is almost always an embedding. The method of reconstruction is irrelevant, as are the values of free parameters of the reconstruction, such as the lag time.

However, in practical applications, the choice of parameters such as the lag time may have a significant effect on the quality of the results. For example, real time series are inevitably contaminated by noise, and different reconstructions behave differently in the presence of noise [4]. Consequently, in the last ten years much of the research in this area has focused on different methods of reconstruction, and on the problem of finding good reconstruction parameters [5–9]. The

[1] Author for correspondence.

[2] Present address: Prediction Company, 234 Griffin St., Santa Fe, NM 87501, USA.

[3] Present address: Tech Partners, 4 Stamford Forum, Stamford, CT 06901, USA.

[#1] Private communication.

0167-2789/92/$05.00

three methods of state space reconstruction in popular use are delays [3], derivatives [2], and principal components [9]. The relations between these methods have remained unclear, as has their dependence on the choice of parameters.

In this paper we analyze the reconstruction problem theoretically. The core of our analysis is a closed-form solution to principal component analysis in the limit of small window width. This solution is based on derivatives of the time series, and it gives insight into the relationship between delays, derivatives and principal components. It provides a theoretical understanding of the behavior of principal components, which makes it possible to compare them quantitatively to derivatives and delays. It also explains several properties of principal component analysis originally observed by Broomhead and King [9], such as the similarity of the eigenvectors to Legendre polynomials.

Perhaps most important, the small-window solution provides a quantitative understanding of the relationship between the free parameters and the reconstructed state space. This allows first-principles analysis of the problems surrounding the choice of parameters. We develop a theoretical guideline for choosing a generally good window width. This guideline is based on quantities that are simple to compute directly from the time series. Furthermore, it yields good results in preliminary numerical experiments.

This paper is closely related to a previously published paper of ours, ref. [4]. In that paper, we proposed a general framework for understanding how state space reconstructions and nonlinear coordinate transformations affect noise and estimation error. In this paper, we limit our attention to delays, derivatives, and principal components. There are interesting connections between the two papers which we have not had time to investigate.

This paper is organized as follows: In section 2, we review delay vectors and principal component analysis. In section 3, we derive a solution to principal component analysis and test it numerically. In section 4, we examine the implications of the solution towards practical problems of state space reconstruction. Section 5 contains a summary of results and open questions. Several mathematical issues are discussed in appendices.

## 2. Review

In this section, we review the work of Packard et al., Takens, and Broomhead and King. This section also serves as an introduction to notation.

### 2.1. Delay vectors

A *delay vector* $\boldsymbol{x}(t)$ for a univariate time series $x(t)$ is defined by

$$\boldsymbol{x}(t) = \big(x(t - m_{\mathrm{p}}\tau), x\big(t - (m_{\mathrm{p}} - 1)\tau\big), \ldots, x(t), \ldots, x(t + m_{\mathrm{f}}\tau)\big)^{\dagger}, \qquad (1)$$

where $\tau$ is the *lag time*, $m_{\mathrm{f}}$ is the number of future coordinates, $m_{\mathrm{p}}$ is the number of past coordinates, and $\dagger$ denotes the transpose. The dimension of a delay vector is $m = m_{\mathrm{p}} + m_{\mathrm{f}} + 1$. We take delay vectors to be column vectors. Usually, delay vectors are defined so that the future-most coordinates are first. For convenience in what follows, we have reversed the order.

Let $l$ be the dimension of the underlying dynamical system which generates $x(t)$[#2]. Takens [3] proved that in the absence of noise, if $m \geq 2l + 1$, then $m$-dimensional delay vectors generically form an embedding of the underlying state space[#3]. An embedding exists for generic $\tau$, so in

[#2] Or, as in Takens' proof, let $l$ be the dimension of a Euclidean manifold that contains the attractor of the dynamical system that generates $x(t)$. The time series $x(t)$ is presumed to be the output of a smooth measurement function on the $l$-dimensional state space.

[#3] Note that while $m \geq 2l + 1$ generically guarantees an embedding, in some cases there is an embedding for $l \leq m \leq 2l$.

the idealization of arbitrarily precise measurements of $x(t)$, the choice of $\tau$ is unimportant to the reconstruction. However, real data are necessarily noisy, and finite amounts of data cause estimation errors. These limitations make the choice of $\tau$ important, for theoretical reasons that we discussed in detail in ref. [4]. In typical practical applications, the dimension $l$ is unknown, so that both $m$ and $\tau$ must be chosen without guidance from Takens' theorem.

A single time series provides data for a sequence of delay vectors. Suppose we have $N$ samples of $x(t)$, sampled at the interval $\Delta t$.

$$\{x(i\Delta t)\}, \quad i \in [0, N-1]. \tag{2}$$

If we set the lag time to an integer multiple of the sampling time, $\tau = h\Delta t$, we can construct $N' = (N-(m_\mathrm{p}+m_\mathrm{f})h)$ delay vectors from the time series. The first delay vector is $\boldsymbol{x}(t_0)$ where $t_0 = m_\mathrm{p}\tau$, and the last is $\boldsymbol{x}(t_0+(N'-1)\Delta t)$. The *delay matrix* $X \in \mathbb{R}^{N' \times m}$ is a normalized sequence of all delay vectors,

$$X = N'^{-1/2} \begin{pmatrix} \boldsymbol{x}^\dagger(t_0) \\ \boldsymbol{x}^\dagger(t_0+\Delta t) \\ \vdots \\ \boldsymbol{x}^\dagger(t_0+(N'-1))\Delta t \end{pmatrix}. \tag{3}$$

Substituting with eq. (1) and $t_0 = m_\mathrm{p}\tau$ reveals that $X$ is invariant with respect to changes in $m_\mathrm{p}$ and $m_\mathrm{f}$ if $m = m_\mathrm{p} + m_\mathrm{f} + 1$ and $\tau$ are held constant:

$$X = N'^{-1/2} \begin{pmatrix} x(0), & \ldots, & x(h(m-1)\Delta t) \\ x(\Delta t), & \ldots, & x((h(m-1)+1)\Delta t) \\ \vdots & & \vdots \\ x(h(N-m+1)\Delta t), \ldots, & & x((N-1)\Delta t) \end{pmatrix}. \tag{4}$$

Thus the distinction between future and past coordinates of $\boldsymbol{x}(t)$ is meaningless to algorithms in which the delay matrix is the only input[#4].

The construction of the delay matrix is the first step of any state space reconstruction method. Alternate methods of reconstruction, such as derivatives and PCA, are really coordinate transformations on delay vectors. To be clear, we will call the construction of $X$ (essentially, the choice of the parameters $m$ and $\tau$) the *delay reconstruction*, and we will call subsequent operations *coordinate transformations*.

### 2.2. Principal component analysis

Principal component analysis (PCA, also known as *Karhunen–Loêve decomposition*, *principal value decomposition*, and *singular systems analysis*) is a general algorithm for decomposing multidimensional data into linearly independent coordinates. Broomhead and King [9] proposed using PCA as a coordinate transformation on delay reconstructions in order to eliminate linearly dependent coordinates and artificial symmetries. We give a brief review in order to provide a background for discussion.

#### 2.2.1. Methodology

The first step in PCA is the estimation of the covariance matrix[#5]. For delay vectors, we define the matrix $\Xi_x \in \mathbb{R}^{m \times m}$ by

$$\Xi_x = X^\dagger X. \tag{5}$$

From the definitions of $X$ and $\Xi_x$, it is easily shown that

$$(\Xi_x)_{ij} = \langle x_i(t)\, x_j(t) \rangle_t, \tag{6}$$

[#4] For example, consider an algorithm that attempts to predict the next value of the time series, $x(N\Delta t)$. If $m$ is held constant, it is equivalent to set $m_\mathrm{p} = 0$ and predict from $\boldsymbol{x}(h(N-m+1)\Delta t)$ or to set $m_\mathrm{f} = 0$ and predict from $\boldsymbol{x}(h(N-1)\Delta t)$.

[#5] PCA can also be formulated in terms of the singular value decomposition of the delay matrix $X$. This is a stabler form for numerical calculations when $X$ is ill-conditioned. See ref. [10].

where $x_i(t)$ denotes the $i$th coordinate[#6] of $x(t)$, $\langle\ \rangle_t$ denotes a time-average, and $t$ ranges from $t_0$ to $t_0+(N'-1)\Delta t$. Hereafter we will suppress the time-indices in averages, and we will assume that $N'\Delta t$ is large enough that eq. (6) is effectively an infinite-time average, in which case $\Xi_x$ approaches the covariance matrix of delay vectors. In this limit, the elements of $\Xi_x$ are given by the autocorrelation function.

$$(\Xi_x)_{ij}=A((i-j)\tau), \tag{7}$$

where

$$A(\tau)=(2T)^{-1}\lim_{T\to\infty}\int_{-T}^{T}x(t)\,x(t-\tau)\,\mathrm{d}t.$$

Throughout this paper, we use $\Xi$ to indicate a covariance matrix and a subscript to indicate its coordinate system.

The next step in PCA is the diagonalization of the covariance matrix. Since $\Xi_x$ is real symmetric it can be written as the product

$$\Xi_x=S\Sigma^2S^\dagger, \tag{8}$$

where $S$ is $m\times m$ orthonormal and $\Sigma^2$ is $m\times m$ diagonal. $S$: $\mathbb{R}^m\mapsto\mathbb{R}^m$ defines a rotation on delay vectors,

$$y^\dagger(t)=x^\dagger(t)\,S. \tag{9}$$

The components $y_j(t)$ of the vector $y(t)$ are called *principal components*.

Define the matrices $Y=XS$ and $\Xi_y=Y^\dagger Y$. It is easily shown that $\Xi_y$ is the covariance matrix of principal components,

$$(\Xi_y)_{ij}=\langle y_iy_j\rangle. \tag{10}$$

By the definitions of $Y$ and $\Xi_x$, eq. (8), and the orthonormality of $S$,

$$\Xi_y=S^\dagger\Xi_xS, \tag{11}$$

$$=\Sigma^2. \tag{12}$$

Thus the covariance matrix of principal components is diagonal, and the principal components are linearly independent.

Let $s_j$ be the $j$th column of $S$ and let $\sigma_j^2=\Sigma_{jj}$. Then $s_j$ is an eigenvector of $\Xi_x$ and $\sigma_j^2$ is the corresponding eigenvalue. By eq. (9), principal components are the projections of delay vectors onto the eigenvectors.

$$y_j(t)=x^\dagger(t)\cdot s_j. \tag{13}$$

By eqs. (10) and (12), the eigenvalues measure the variance of the principal components[#7]

$$\langle y_j^2\rangle=\sigma_j^2. \tag{14}$$

The set of $m$ eigenvalues, $\{\sigma_i^2\}$, $i\in[0,m-1]$, is called the *singular spectrum*. The eigenvalues and eigenvectors are ordered so that $\sigma_0^2\geq\sigma_1^2\geq\sigma_2^2$, etc.

Sauer et al. [11] extended Takens' proof to principal components, showing that generically $2l+1$ principal components form an embedding. In some cases, fewer principal components are needed. Let $q$ be the minimum number of principal components which form an embedding. Since we are interested in embeddings, we will assume that $m>q$.

### 2.2.2. Motivation

In some cases PCA can reduce noise. Here we have in mind setting the delay dimension $m$ large, and then projecting the delay reconstruction into $q<m$ principal components.

Suppose that the time series $x(t)$ is the sum of a smooth time series $\tilde{x}(t)$ and a Gaussian IID noise process $\eta(t)$,

$$x(t)=\tilde{x}(t)+\eta(t). \tag{15}$$

This induces isotropic Gaussian IID noise in delay coordinates, with variance $\langle\eta^2\rangle$. The projection of the noise in any direction also has variance $\langle\eta^2\rangle$; thus the signal-to-noise ratio of

[#6] Our convention is to start indices at zero: vectors begin with $i=0$ and matrices with $(i,j)=(0,0)$.

[#7] Note that for principal components, $\langle y_j\rangle=0$ for $j>0$, therefore $\langle y_j^2\rangle$ is the variance of $y_j$ for $j>0$.

any rotated coordinate is proportional to the square root of its variance. For example, the signal-to-noise ratio of $y_j(t)$ is $\sqrt{\langle y_j^2\rangle / \langle \eta^2 \rangle}$.

When the time series has noise in the form of eq. (15), PCA is the optimal linear coordinate transformation. This is because subsets of principal components have maximum variance, and consequently, maximum signal-to-noise ratios. Precisely speaking, for all orthonormal coordinate transformations $S'$: $\mathbb{R}^m \mapsto \mathbb{R}^m$ and $y'^{\dagger} = x^{\dagger} S'$, on a fixed set of delay vectors $X$, and for all values of $d \in [1, m]$,

$$\sum_{i=0}^{d-1} \sigma_i^2 = \sum_{i=0}^{d-1} \langle y_i^2 \rangle \geq \sum_{i=0}^{d-1} \langle y_i'^2 \rangle. \qquad (16)$$

For proof, see ref. [17]. Because they have maximum variance, the first $d$ principal components have the maximum signal-to-noise ratios of all $d$-dimensional projections of a fixed delay reconstruction. In this sense, PCA is the optimal linear coordinate transformation.

However, there are two major qualifications: First, a set of $d$ principal components is optimal only for a *fixed* delay reconstruction. Changing the delay reconstruction (i.e. changing $m$ or $\tau$) generally changes the singular spectrum and thus the signal-to-noise ratios. Second, the variances of principal components are constrained by the variance of the time series. Because the trace of a matrix is invariant under similarity transformations, $\mathrm{Tr}\, \Xi_y = \mathrm{Tr}\, \Xi_x$, or equivalently,

$$\sum_{j=0}^{m-1} \sigma_j^2 = m \langle x^2 \rangle. \qquad (17)$$

The total variance of all $m$ principal components is the same as that of all $m$ delay coordinates. If the first few principal components are very large, the last principal components must be very small. The principal components with the smallest variance can actually have *worse* signal-to-noise ratios than delays. We can compare the signal-to-noise ratio of a given principal component $y_j(t)$ to that of a delay coordinate by comparing $\sigma_j^2$ with $\langle x^2 \rangle$. Again, let $q < m$ be the number of principal components ($y_0$ through $y_{q-1}$) which form an embedding. If

$$\sigma_{q-1}^2 > \langle x^2 \rangle, \qquad (18)$$

then $y_0$ through $y_{q-1}$ have better signal-to-noise ratios than an individual delay coordinate.

PCA can also detect noise-dominated coordinates. Broomhead and King noted that often the singular spectrum decreases until it hits a plateau, after which the eigenvalues are roughly equal. One possible (but not necessary) explanation for a plateau is the presence of Gaussian noise on the time series. If we assume the time series $x(t)$ is of the form of eq. (15), then the isotropic noise in delay coordinates imposes a lower bound of $\langle \eta^2 \rangle$ on each eigenvalue. Under this assumption, the height of the plateau indicates the variance of the noise, and an eigenvalue lying on the plateau represents a noise-dominated principal component. If $q$ principal components form an embedding and

$$\sigma_{q-1}^2 \gg \langle \eta^2 \rangle, \qquad (19)$$

then we say that the state space of $q$ principal components is *approximately deterministic*.

Of course, principal components are simply projections of delay coordinates. If the condition specified by eq. (19) is not met, the $m$-dimensional delay reconstruction effectively occupies a less-than-$q$-dimensional subspace of $\mathbb{R}^m$. Consequently, the delay reconstruction does not form an approximately deterministic state space, even if $m \geq 2l + 1$ (see ref. [9]). The advantage of PCA over delays is that it makes this problem apparent.

Another advantage of PCA is that it provides a rough characterization of the delay reconstruction. As noted in ref. [9], the delay vectors in the time series can be thought of as exploring, on average, an $m$-dimensional ellipsoid. The eigenvectors $\{s_j\}$ give the directions and the eigenvalues $\{\sigma_j^2\}$ give the lengths of the principal axes of this ellipsoid.

In the following section, we develop a theoretical understanding of PCA, which will provide us with a better understanding of these issues.

## 3. Small-window solution

Theoretical insight to PCA can be gained by studying its properties in appropriate limits. For example, if $\tau$ is held constant and $m$ tends to infinity, it can be shown that PCA becomes discrete Fourier analysis. For a good review, see Vautard and Ghil [12].

The main result of this paper is a solution to PCA in the limit of small window widths. In this section, we derive the small-window solution, and shows how it relates delays, derivatives, and principal components. The derivation is indirect: First we find a coordinate transformation that relates delay vectors to derivatives of the time series. Then we show that the coordinate transformation gives the covariance matrix a simple form that can be diagonalized to leading order in closed form.

We put the following restrictions on $x(t)$:

(1) $x(t)$ is analytic on $t \in \mathbb{R}$.

(2) $x(t)$ is bounded and its derivatives are bounded for $t \in \mathbb{R}$.

(3) $\lim_{T\to\infty} T^{-1}\int_{-T}^{T} x^2(t)\,\mathrm{d}t \neq 0$,

(4) $\lim_{T\to\infty} T^{-1}\int_{-T}^{T} (x^{(1)}(t))^2\,\mathrm{d}t \neq 0$,

where $x^{(i)} = \mathrm{d}^i x/\mathrm{d}t^i$. The generalization to $x(t)$ with additive Gaussian IID noise is straightforward (see ref. [9]). The solution can also be generalized to $\mathrm{C}^\infty$ functions by including error terms in Taylor expansions of $x(t)$.

Note that these restrictions exclude such functions as $x(t) = t^2$ and $x(t) = \exp(-t^2)$. An example of a function which satisfies all the restrictions is $x(t) = \sin t$. But we are primarily interested in smooth functions given by real-valued projections of trajectories on chaotic attractors of smooth dynamical systems.

A word about notation: The usual dimension parameter, $m$, is inconvenient for this analysis. For convenience, we set $m$ to be odd, i.e. $m = 2p+1$. Then the invariance of $X$ and $\Xi_x$ with respect to $m_{\mathrm{p}}$ and $m_{\mathrm{f}}$ if $m$ and $\tau$ are held constant allows us to express delay vectors symmetrically. For $\boldsymbol{x} \in \mathbb{R}^{m=2p+1}$, we let $m_{\mathrm{p}} = m_{\mathrm{f}} = p$, so that

$$\boldsymbol{x}(t) = (x(t-p\tau), \ldots, x(t), \ldots, x(t+p\tau)). \tag{20}$$

As a result, the small-window solution comes out in terms of integers $p$ representing odd values of $m$. Also, we will describe the time-scale of delay vectors with the *window width* $\tau_{\mathrm{w}} = (m-1)\tau$, instead of the lag time $\tau$.

### 3.1. Coordinate transformation

#### 3.1.1. Derivatives and discrete Legendre polynomials

First we derive a coordinate transformation which involves the derivatives of the time series. The $j$th-order derivative of $x(t)$ can be estimated by a discrete linear filter

$$w_j(t) = \sum_{n=-p}^{p} r_{j,p}(n)\, x(t+n\tau), \tag{21}$$

where the time series $x(t)$ is the input, $w_j(t)$ is the output, and $r_{j,p}(n)$ is an appropriate discrete convolution kernel, parameterized by the choice of $p$ and the order of the desired derivative, $j$.

Since $x(t)$ is analytic, we can expand it in a Taylor series, provided that the window width $\tau_{\mathrm{w}}$ is sufficiently small.

$$w_j(t) = \sum_{n=-p}^{p} r_{j,p}(n) \left( \sum_{i=0}^{\infty} \frac{(n\tau)}{i!} x^{(i)}(t) \right). \tag{22}$$

Assume that we can switch the order of summation to obtain

$$w_j(t) = \sum_{i=0}^{\infty} \frac{\tau^i}{i!} x^{(i)}(t) \left[ \sum_{n=-p}^{p} n^i r_{j,p}(n) \right]. \tag{23}$$

From eq. (23) it is clear that we can make $w_j(t)$ proportional to the $j$th derivative by causing the bracketed factor to vanish for $i<j$. This is done by choosing $r_{j,p}(n)$ so that it is orthogonal to $n^i$, i.e.

$$\sum_{n=-p}^{p} n^i r_{j,p}(n) = 0 \quad \text{for } i<j. \tag{24}$$

A kernel $r$ which satisfies this constraint leaves the $i=j$ term in eq. (23) as the leading-order term in $\tau_w$, so that $w_j(t)$ is approximately proportional to $x^{(j)}(t)$.

Many filters satisfy eq. (24). We restrict our attention to mutually orthonormal filters. This provides an additional constraint,

$$\sum_{n=-p}^{p} r_{i,p}(n)\, r_{j,p}(n) = \delta_{ij} \quad \text{for } i,j \le 2p. \tag{25}$$

It can be shown that the orthogonality constraints of eqs. (24) and (25) specify a unique set of $m$ kernel functions, which can be generated from the recurrence relation

$$r_{j,p}(n) = \frac{1}{c_j p^j}\left( n^j - \sum_{k=0}^{j-1} r_{k,p}(n) \sum_{l=-p}^{p} l^j r_{k,p}(l) \right)$$
$$\text{for } j \le 2p, \tag{26}$$

where $c_j$ is a normalization constant. It can be shown that $r_{j,p}(n)$ is an even or odd $j$th-degree polynomial in $n$ for even or odd $j$. The normalization constant $c_j$ can be determined from the requirement that $\Sigma_{n=-p}^{p} r_{j,p}^2(n) = 1$; this makes $c_j$ a function of $p$. It can be shown that $c_j(p)$ scales as $\sqrt{p}$ for large $p$. Formulae for the first six $c_j(p)$ and the first six $r_{j,p}(n)$ are given in appendix A.

The choice of normalization determines the constant of proportionality between $w_j(t)$ and $x^{(j)}(t)$. From eqs. (25) and (26), it can be shown that

$$\sum_{n=-p}^{p} n^j r_{j,p}(n) = p^j c_j(p). \tag{27}$$

Plugging eqs. (24) and (27) into eq. (23), and making the substitution $p\tau = \frac{1}{2}\tau_w$ gives

$$w_j(t) = \frac{c_j(p)\tau_w^j}{2^j j!} x^{(j)}(t) + \mathcal{O}(\tau_w^{j+2}). \tag{28}$$

The even/odd symmetry of $r_{j,p}(n)$ causes the order $\tau_w^{j+1}$ term to vanish. To leading order, the output $w_j(t)$ is proportional to the $j$th-order derivative $x^{(j)}(t)$, with a constant of proportionality determined by $j$, $p$, and $\tau_w$.

Discrete forms of continuous orthogonal functions are widely used in digital signal processing (see, for example, ref. [13]). In the limit $p \to \infty$, the kernels $r_{j,p}(n)$ reduce to the Legendre polynomials[#8], so we call them *discrete Legendre polynomials*. On the other hand, each $r_{j,p}(n)$ reduces to a standard finite-difference filter for estimating a derivative[#8] when $p$ takes on its smallest value (this value is different for each value of $j$). These relationships are examined in appendix A.

### 3.1.2. Discrete Legendre polynomials in $\mathbb{R}^m$

The discrete Legendre polynomials form an orthonormal basis in $\mathbb{R}^{m=2p+1}$, with basis vectors $\boldsymbol{r}_j$ defined by

$$\boldsymbol{r}_j = \left(r_{j,p}(-p), \dots, r_{j,p}(0), \dots, r_{j,p}(p)\right)^{\dagger}. \tag{29}$$

By eq. (21) the projection of a delay vector onto a basis vector $\boldsymbol{r}_j$ is

$$w_j(t) = \boldsymbol{x}^{\dagger}(t)\, \boldsymbol{r}_j. \tag{30}$$

We call $w_j(t)$ a *Legendre coordinate*, since it is the projection of a delay vector onto a discrete Legendre polynomial. We emphasize that the

[#8] Except for a difference of normalization.

Legendre *coordinate* $w_j(t)$ is a time-varying, state-dependent quantity, proportional to a derivative of the time series. This is opposed to the discrete Legendre *polynomial* $\boldsymbol{r}_j$, which is a fixed basis vector of an orthogonal coordinate system in $\mathbb{R}^m$. Legendre coordinates and discrete Legendre polynomials are related by eq. (30).

Taken together, the $m$ discrete Legendre polynomials define a transformation $R$: $\mathbb{R}^m \mapsto \mathbb{R}^m$, given by the $m \times m$ matrix

$$R = (\boldsymbol{r}_0, \boldsymbol{r}_1, \ldots, \boldsymbol{r}_{m-1}). \tag{31}$$

By eqs. (25), $\boldsymbol{r}_i^\dagger \cdot \boldsymbol{r}_j = \delta_{ij}$, and therefore $R$ is orthonormal.

Define the vector of Legendre coordinates by $\boldsymbol{w}(t) = (w_0(t), w_1(t), \ldots, w_{m-1}(t))^\dagger$. Then by eqs. (30) and (31),

$$\boldsymbol{w}^\dagger(t) = \boldsymbol{x}^\dagger(t)\, R. \tag{32}$$

Since $R$ is orthonormal, Legendre coordinates are a simple rotation of delay coordinates. When $\tau_\text{w}$ is small, the Legendre coordinates are proportional to derivatives; therefore the relationship between delays and derivatives consists of a rotation and a rescaling.

However, this is *not* to say that Legendre coordinates are equivalent to derivative coordinates obtained through the standard finite-difference estimators. First, the reduction of discrete Legendre polynomials to finite-difference filters takes place at a different value of $p$ for each $\boldsymbol{r}_j$. Therefore, the standard finite-difference filters ((1), $(-1,1)$, $(1,-2,1)$, etc.) correspond to $\boldsymbol{r}_j$'s of different dimensions. Embedded as vectors in $\mathbb{R}^m$, the finite difference filters are not orthonormal. For example, the first three finite-difference filters embedded as vectors in $\mathbb{R}^3$ are $(0,1,0)$, $(0,-1,1)$, and $(1,-2,1)$. Consequently, if we use finite-difference filters to form a state space of derivatives, the noise in the state space is nonisotropic, and the noise on different derivative-coordinates is correlated. Second, the Legendre coordinates are proportional, not equal, to derivatives. This is not a trivial difference, since for noisy $x(t)$ the prefactor in eq. (28) determines the signal-to-noise ratio of the Legendre coordinate. Generally, the signal-to-noise ratios of Legendre coordinates are better than those of finite-difference estimates of derivatives. These issues are discussed in detail in appendix A. From a practical point of view, Legendre coordinates are generally a better choice than finite-difference estimates of derivatives. Because of this, we will abandon discussion of derivatives in favor of the better-behaved Legendre coordinates.

When $\tau_\text{w}$ is small, the Legendre coordinates make it possible to quantitatively estimate the gross shape of an attractor in delay coordinates, and in particular, how the shape of the attractor varies with $\tau_\text{w}$. From appendix A, $\boldsymbol{r}_0^\dagger = (1,1,\ldots)/\sqrt{m}$, which corresponds to the identity line. According to eqs. (28) and (30), the projection of $\boldsymbol{x}$ onto this line is $w_0 = \mathcal{O}(1)$. Directions orthogonal to the identity line correspond to $\boldsymbol{r}_j$ with higher values of $j$; in these directions the projection is $w_j = \mathcal{O}(\tau_\text{w}^j)$. This explains the well-known fact that for small $\tau_\text{w}$, the attractor is extended along the identity line and squeezed in the directions perpendicular to it. Furthermore, it shows that for small $\tau_\text{w}$ the reconstructed attractor lies within a long, thin, ellipsoid. The principal axes of the ellipsoid range from $\langle w_0^2 \rangle = \mathcal{O}(1)$ to $\langle w_{m-1}^2 \rangle = \mathcal{O}(\tau_\text{w}^{2m-2})$ in length. The orientation of the principal axes is given by the discrete Legendre polynomials. Fig. 1 shows this for a three-dimensional reconstruction of the Lorenz $x(t)$. The attractor is extended along $\boldsymbol{r}_0$, narrower along $\boldsymbol{r}_1$, and very narrow along $\boldsymbol{r}_2$, as predicted by eq. (28).

Note that this description of the reconstruction as an ellipse is very similar to the one given by PCA (see section 2.2.2). In the next section we will show that PCA is closely connected to the discrete Legendre polynomials. Further, note that derivatives can be estimated from the time series, and that the other factors in eq. (28) are known. Thus eq. (28) allows us to make theoretical esti-

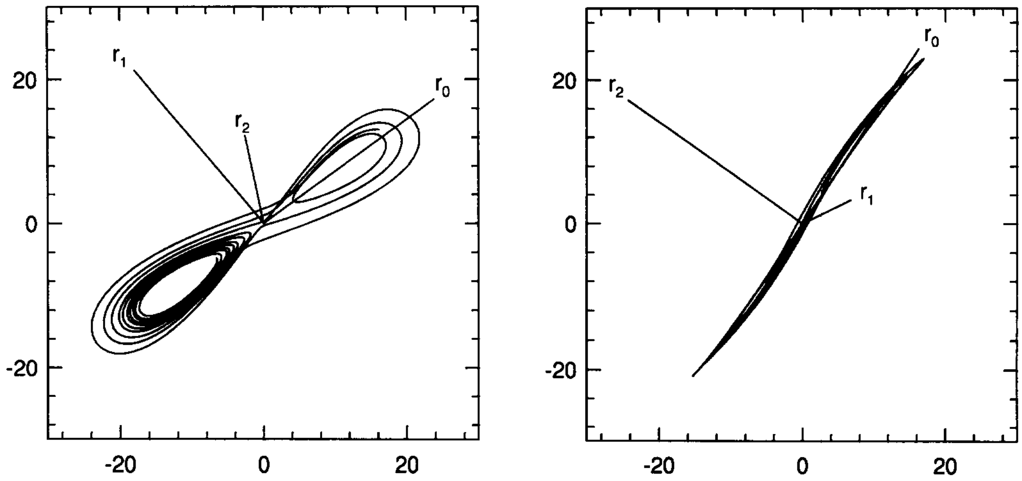

Fig. 1. Discrete Legendre polynomials and a Lorenz attractor reconstructed with delays. Here we plot two different projections of an attractor reconstructed from the Lorenz $x(t)$ and the orthogonal coordinate system defined by the discrete Legendre polynomials. The reconstruction was made with $m=3$, $\tau=0.04$ delay vectors. The three discrete Legendre polynomials for $\mathbb{R}^3$ were calculated from formulae in appendix A: $r_0^\dagger=(1,1,1)/\sqrt{3}$, $r_1^\dagger=(-1,0,1)/\sqrt{2}$, and $r_2^\dagger=(1,-2,1)/\sqrt{6}$. The Lorenz system is given by eqs. (62)–(64).

mates of the geometry of delay reconstructions when the time series is the only available information. We will return to this idea in section 4.2.

### 3.2. Diagonalizing the covariance matrix

We now return to the original problem of solving PCA. As stated in section 2.2, PCA is solved by finding a rotation on delay vectors which diagonalizes the covariance matrix. In this section, we show that the rotation from delays to Legendre coordinates almost diagonalizes the covariance matrix, and that the remaining rotation can be approximated in closed form.

#### 3.2.1. Covariance of Legendre coordinates

The matrix $R$ defines a transformation between delay coordinates and Legendre coordinates, $w^\dagger(t)=x^\dagger(t)R$. We carry out the same transformation on the delay matrix to define $W=XR$. Then the covariance matrix of Legendre coordinates is given by $\Xi_w=W^\dagger W=R^\dagger \Xi_x R$, with elements

$$(\Xi_w)_{ij}=\langle w_i w_j\rangle. \tag{33}$$

The relation between Legendre coordinates and derivatives allows us to calculate $\Xi_w$ explicitly. Substituting into eq. (33) with eq. (28) gives

$$(\Xi_w)_{ij}=\frac{\tau_w^{i+j}}{2^{i+j}}\frac{c_i c_j}{i!\,j!}\langle x^{(i)}x^{(j)}\rangle+\mathcal{O}\left(\tau_w^{i+j+2}\right). \tag{34}$$

Since $x(t)$ is bounded and has bounded derivatives, integration by parts shows that

$$\langle x^{(i)}x^{(j)}\rangle=\lim_{T\to\infty}\frac{1}{2T}\int_{-T}^{T}x^{(i)}(t)\,x^{(j)}(t)\,\mathrm{d}t \tag{35}$$

$$=\begin{cases}(-1)^{(i-j)/2}\left\langle\left(x^{((i+j)/2)}\right)^2\right\rangle & \text{for } i+j \text{ even,}\\ 0 \quad \text{for } i+j \text{ odd.}\end{cases} \tag{36}$$

Substituting into eq. (34) with eq. (36) gives

$$(\Xi_w)_{ij} = \begin{cases} (-1)^{(i-j)/2} \dfrac{\tau_\mathrm{w}^{i+j}}{2^{i+j}} \dfrac{c_i c_j}{i!j!} \left\langle \left( x^{((i+j)/2)} \right)^2 \right\rangle \\ \quad + \mathscr{O}\left( \tau_\mathrm{w}^{i+j+2} \right) & \text{for } i+j \text{ even}, \\ 0 & \text{for } i+j \text{ odd}. \end{cases} \tag{37}$$

The even/odd symmetry of the discrete Legendre polynomials causes $(\Xi_w)_{ij}$ to vanish to all orders of $\tau_\mathrm{w}$ when $i+j$ is odd. To simplify the equations we define

$$\kappa_i = \left\langle \left( x^{(i)} \right)^2 \right\rangle. \tag{38}$$

Then $\Xi_w$ written out in matrix form is

$$\Xi_w = \begin{pmatrix} c_0^2 \kappa_0 & 0 & -\dfrac{c_0 c_2 \tau_\mathrm{w}^2}{2^2 2!} \kappa_1 & 0 & \cdots \\ 0 & \dfrac{c_1^2 \tau_\mathrm{w}^2}{2^2} \kappa_1 & 0 & -\dfrac{c_1 c_3 \tau_\mathrm{w}}{2^4 3!} \kappa_2 & \\ -\dfrac{c_2 c_0 \tau_\mathrm{w}^2}{2^2 2!} \kappa_1 & 0 & \dfrac{c_2^2 \tau_\mathrm{w}^4}{2^4 2!^2} \kappa_2 & 0 & \\ 0 & -\dfrac{c_3 c_1 \tau_\mathrm{w}^4}{2^4 3!} \kappa_2 & 0 & \dfrac{c_3^2 \tau_\mathrm{w}^6}{2^6 3!^2} \kappa_3 & \\ & & & & \ddots \\ \vdots & & & & \end{pmatrix}$$

$$+ \left[ (i+j+1 \bmod 2) \mathscr{O}\left( \tau_\mathrm{w}^{i+j+2} \right) \right]. \tag{39}$$

The second, bracketed, term indicates a matrix whose elements are given by the enclosed formula.

Eq. (36) has a few interesting consequences; these are examined in appendix B. One consequence is that if $\kappa_0$ and $\kappa_1$ are nonzero (as required by the restrictions on $x(t)$), then $\kappa_i \neq 0$ for $i \geq 0$.

### 3.2.2. Diagonalization of Legendre coordinates

In a loose sense, the rotation from delay coordinates to Legendre coordinates diagonalizes the covariance matrix to leading order, because the $(0,0)$ element is the only order-1 element in $\Xi_w$. However, $\Xi_w$ is not diagonal in the sense that is important to PCA. PCA finds linearly independent coordinates, and linear dependence is measured by correlation. For $i+j$ even, the correlation between $w_i(t)$ and $w_j(t)$ is given by

$$\frac{(\Xi_w)_{ij}}{\sqrt{(\Xi_w)_{ii} (\Xi_w)_{jj}}} = \frac{\mathscr{O}\left( \tau_\mathrm{w}^{i+j} \right)}{\sqrt{\mathscr{O}\left( \tau_\mathrm{w}^{2i} \right) \mathscr{O}\left( \tau_\mathrm{w}^{2j} \right)}} = \mathscr{O}(1). \tag{40}$$

Since the correlation is of order 1, the Legendre coordinates are not linearly independent, and in this sense $\Xi_w$ is not diagonal. Therefore, a further rotation is needed to approximate PCA.

Because $(\Xi_w)_{ij} = 0$ for odd $i+j$, the diagonalization of $\Xi_w$ can be decomposed into two separate diagonalizations, one among the even coordinates of $\boldsymbol{w}$ and one among the odd. Define a $\frac{1}{2}(m+1) \times \frac{1}{2}(m+1)$ covariance matrix of the even coordinates, $\Xi_w^\mathrm{e}$, by

$$(\Xi_w^\mathrm{e})_{ij} = (\Xi_w)_{2i,2j}. \tag{41}$$

Then

$$\Xi_w^\mathrm{e} = \begin{pmatrix} c_0^2 \kappa_0 & -\dfrac{c_0 c_2 \tau^2}{2^2 2!} \kappa_1 & \dfrac{c_0 c_4 \tau_\mathrm{w}^4}{2^4 4!} \kappa_2 & \cdots \\ -\dfrac{c_0 c_2 \tau_\mathrm{w}^2}{2^2 2!} \kappa_1 & \dfrac{c_2^2 \tau_\mathrm{w}^4}{2^4 2!^2} \kappa_2 & -\dfrac{c_2 c_4 \tau_\mathrm{w}^6}{2^6 2! 4!} \kappa_3 & \\ \dfrac{c_0 c_4 \tau_\mathrm{w}^4}{2^4 4!} \kappa_2 & -\dfrac{c_2 c_4 \tau_\mathrm{w}^6}{2^6 2! 4!} \kappa_3 & \dfrac{c_4^2 \tau_\mathrm{w}^8}{2^8 4!^2} \kappa_4 & \\ & & & \ddots \\ \vdots & & & \end{pmatrix}$$

$$+ \left[ \mathscr{O}\left( \tau_\mathrm{w}^{2i+2j+2} \right) \right]. \tag{42}$$

If $\tau_\mathrm{w}$ is small enough that the decrease of $c_i c_j \tau_\mathrm{w}^{2i+2j} / (2^{2i+2j} (2i)!(2j)!)$ with $i$ and $j$ dominates possible increases in $\kappa_{i+j}$, then $\Xi_w^\mathrm{e}$ can be diagonalized to leading order in closed form by the method described in appendix C (at least to some finite $i+j$). In appendix C, we show that the approximation breaks down as $\tau_\mathrm{w}$ nears $2\sqrt{3\kappa_0/\kappa_1}$. We therefore define the *critical window width* $\tau_\mathrm{w}^*$ by

$$\tau_\mathrm{w}^* = 2\sqrt{\frac{3\kappa_0}{\kappa_1}}. \tag{43}$$

The following solution to PCA is valid for $\tau_\mathrm{w} \ll \tau_\mathrm{w}^*$.

The eigenvalues of $\Xi_w^e$ are the even eigenvalues of $\Xi_x$. By the results of appendix C, these are

$$\sigma_0^2 = c_0^2 \kappa_0 + \mathscr{O}(\tau_w^2), \tag{44}$$

$$\sigma_2^2 = \left(\frac{c_2 \tau_w^2}{2^2 2!}\right)^2 \left(\kappa_2 - \frac{\kappa_1^2}{\kappa_0}\right) + \mathscr{O}(\tau_w^6), \tag{45}$$

$$\sigma_4^2 = \left(\frac{c_4 \tau_w^4}{2^4 4!}\right)^2 \left(\kappa_4 - \frac{\kappa_2^2}{\kappa_0} - \frac{(\kappa_1\kappa_2 - \kappa_0\kappa_3)^2}{\kappa_0(\kappa_0\kappa_2 - \kappa_1^2)}\right) + \mathscr{O}(\tau_w^{10}). \tag{46}$$

The covariance matrix of odd Legendre coordinates can be diagonalized in the same way, giving

$$\sigma_1^2 = \left(\frac{c_1 \tau_w}{2}\right)^2 \kappa_1 + \mathscr{O}(\tau_w^4), \tag{47}$$

$$\sigma_3^2 = \left(\frac{c_3 \tau_w^3}{2^3 3!}\right)^2 \left(\kappa_3 - \frac{\kappa_2^2}{\kappa_1}\right) + \mathscr{O}(\tau_w^8), \tag{48}$$

$$\sigma_5^2 = \left(\frac{c_5 \tau_w^5}{2^5 5!}\right)^2 \left(\kappa_5 - \frac{\kappa_3^2}{\kappa_1} - \frac{(\kappa_2\kappa_3 - \kappa_1\kappa_4)^2}{\kappa_1(\kappa_1\kappa_3 - \kappa_2^2)}\right) + \mathscr{O}(\tau_w^{12}). \tag{49}$$

The diagonalization procedure in appendix C requires that $\tau_w$ be small enough that the eigenvalues of $\Xi_w$ decrease rapidly. Therefore the critical window width $\tau_w^*$ can also be derived retrospectively by setting $\sigma_0^2 = \sigma_1^2$, solving for $\tau_w$, and substituting with the limiting value of $c_0(p)/c_1(p)$.

Next we find the eigenvectors of $\Xi_x$, which are the columns of $S$, where $S^+ \Xi_x S = \Sigma$. Let $V$ be the $m \times m$ matrix whose columns are the eigenvectors of $\Xi_w$, i.e. $V^+ \Xi_w V = \Sigma$. Since $\Xi_w = R^+ \Xi_x R$, then $(RV)^+ \Xi_x RV = \Sigma$. Therefore $S = RV$. From appendix C,

$$V = \mathrm{I}_m + [\mathscr{O}(\tau_w^2)], \tag{50}$$

where $\mathrm{I}_m$ is the $m \times m$ identity matrix, and $[\mathscr{O}(\tau_w^2)]$ indicates a matrix whose elements are order-$\tau_w^2$. Substituting, eq. (50) in $S = RV$ gives

$$S = R + R[\mathscr{O}(\tau_w^2)], \tag{51}$$

$$= R + [\mathscr{O}(\tau_w^2)]. \tag{52}$$

We can make the substitution $R[\mathscr{O}(\tau_w^2)] = [\mathscr{O}(\tau_w^2)]$ because $R$ is orthonormal. Geometrically, eq. (52) means that the rotation from Legendre coordinates to principal components is small. Taking eq. (52) column by column,

$$s_j = r_j + \mathscr{O}(\tau_w^2). \tag{53}$$

Thus, to leading order, the eigenvectors of $\Xi_x$ are the discrete Legendre polynomials. Although this might seem to contradict the statement that discrete Legendre polynomials do not diagonalize $\Xi_x$ to leading order, this is not the case: Because the eigenvalues of $\Xi_x$ range from order-1 to order-$\tau_w^{2m}$, a small error in an high-order eigenvector can result in an eigenvalue error that is as large as or larger than the eigenvalue itself. Thus eq. (53) provides a way to calculate leading-order approximations to eigenvectors, but these approximations cannot be used to make leading-order approximations to eigenvalues or principal components.

Formulae for principal components must be derived by other means. In appendix C.3, we show how principal components are a Gram–Schmidt orthogonalization of Legendre coordinates. This gives the following recurrence relation:

$$y_j(t) = w_j(t) - \sum_{i=0}^{j-1} y_i(t) \frac{\langle y_i w_j \rangle}{\langle y_i^2 \rangle} + \mathscr{O}(\tau_w^{j+2}). \tag{54}$$

By eq. (28), an alternative form of the recurrence is

$$y_j(t) = \frac{c_j \tau_w^j}{2^j j!} \left( x^{(j)}(t) - \sum_{i=0}^{j-1} y_i(t) \frac{\langle y_i x^{(j)} \rangle}{\langle y_i^2 \rangle} \right) + \mathscr{O}(\tau_w^{j+2}). \tag{55}$$

Because the even derivatives of $x(t)$ are uncorrelated with the odd, the terms in the sum with $i+j$ odd vanish, for both recurrence relations. We represent them in the sums anyway, for ease of expression.

Iterating the recurrence relation gives

$$y_0(t) = c_0\left[x^{(0)}(t)\right] + \mathscr{O}\left(\tau_{\mathrm{w}}^2\right), \tag{56}$$

$$y_1(t) = \frac{c_1\tau_{\mathrm{w}}}{2}\left[x^{(1)}(t)\right] + \mathscr{O}\left(\tau_{\mathrm{w}}^3\right), \tag{57}$$

$$y_2(t) = \frac{c_2\tau_{\mathrm{w}}^2}{2^2 2!}\left[x^{(2)}(t) + x^{(0)}(t)\left(\kappa_1/\kappa_0\right)\right] + \mathscr{O}\left(\tau_{\mathrm{w}}^4\right), \tag{58}$$

$$y_3(t) = \frac{c_3\tau_{\mathrm{w}}^3}{2^3 3!}\left[x^{(3)}(t) + x^{(1)}(t)\left(\kappa_2/\kappa_1\right)\right] + \mathscr{O}\left(\tau_{\mathrm{w}}^5\right), \tag{59}$$

$$y_4(t) = \frac{c_4\tau_{\mathrm{w}}^4}{2^4 4!}\left(x^{(4)}(t) + x^{(2)}(t)\frac{\kappa_1\kappa_2 - \kappa_0\kappa_3}{\kappa_1^2 - \kappa_0\kappa_2} + x^{(0)}(t)\frac{\kappa_2^2 - \kappa_1\kappa_3}{\kappa_1^2 - \kappa_0\kappa_2}\right) + \mathscr{O}\left(\tau_{\mathrm{w}}^6\right), \tag{60}$$

$$y_5(t) = \frac{c_5\tau_{\mathrm{w}}^5}{2^5 5!}\left(x^{(5)}(t) + x^{(3)}(t)\frac{\kappa_2\kappa_3 - \kappa_1\kappa_4}{\kappa_2^2 - \kappa_1\kappa_3} + x^{(1)}(t)\frac{\kappa_3^2 - \kappa_2\kappa_4}{\kappa_2^2 - \kappa_1\kappa_3}\right) + \mathscr{O}\left(\tau_{\mathrm{w}}^7\right). \tag{61}$$

Squaring and averaging over time shows that these principal components are consistent with the eigenvalues given by eqs. (44)–(49).

Note that the recurrence relation implies $y_j = \mathscr{O}(\tau_{\mathrm{w}}^j)$. Thus, for a $d$-dimensional reconstruction with coordinates $y_0$ through $y_{d-1}$, the noisiest coordinate is $y_{d-1}$, with signal-to-noise scaling as $\tau_{\mathrm{w}}^{d-1}$. In ref. [4] we defined the *distortion* of a reconstruction, which measures the detrimental effect of noise on a reconstruction, and we showed that for small window widths, the distortion scales asymptotically as $\tau_{\mathrm{w}}^{d-1}$. This is closely related to the scaling of $y_{d-1}$. In fact, eqs. (56)–(61) provide a method of calculating the distortion of a reconstruction explicitly, if the map from the original coordinates to the time series' derivatives is known.

### 3.3. *Numerical tests*

In this section, we present numerical tests of these results. The data used in this section were obtained from numerical integration of the Lorenz equations,

$$x^{(1)} = sx - su, \tag{62}$$

$$u^{(1)} = rx - u - xv, \tag{63}$$

$$v^{(1)} = -bv + xu. \tag{64}$$

We used the parameter values $s = 10$, $r = 28$, $b = \frac{8}{3}$ and a fourth-order Runge–Kutta algorithm, with a fixed integration step $\mathrm{d}t = 0.001$. The time series $\{x(\mathrm{i}\Delta t)\}$ consisted of 50 000 values of $x(t)$ sampled at the interval $\Delta t = 0.01$.

To predict the behavior of principal component analysis, we needed to compute the derivatives $x^{(i)}$ and their average squared values, $\kappa_i$. Differentiating eq. (62) with respect to time and substituting with eqs. (62)–(64) gives an explicit function for $x^{(2)}$ in terms of $(x, u, v)$. We iterated the differentiation to obtain functions for $x^{(3)}$, $x^{(4)}$, and $x^{(5)}$. We evaluated these functions over the 50 000 point trajectory of $(x, u, v)$ to estimate $\kappa_j = \langle (x^{(j)})^2 \rangle$ and to obtain a time series of derivatives. Note that the derivatives and the $\kappa$'s could also be estimated directly from the time series of $x(t)$, for example, by using the discrete Legendre polynomials and eqs. (21) and (28).

Fig. 2a shows a segment of $x(t)$ in the time units of eqs. (62)–(64). Fig. 2b shows a numerical estimate of the autocorrelation function of $x(t)$, $A(\tau) = \langle x(t)\,x(t-\tau)\rangle$. For the purpose of visual comparison, the critical window width $\tau_{\mathrm{w}}^*$ is indicated in both (a) and (b). We estimated $\tau_{\mathrm{w}}^*$ using eq. (43) and numerical estimates of $\kappa_0$ and $\kappa_1$. This gave the value $\tau_{\mathrm{w}}^* \approx 0.63$, which falls near the first minimum of the autocorrelation function, at $\tau \approx 0.69$. For simple functions like the Lorenz $x(t)$, such correspondence is probably not coincidental, since $\tau^*$ and the first minimum of the autocorrelation function are both related to the "period" of the system. But in general we do not expect such correspondence, since the auto-

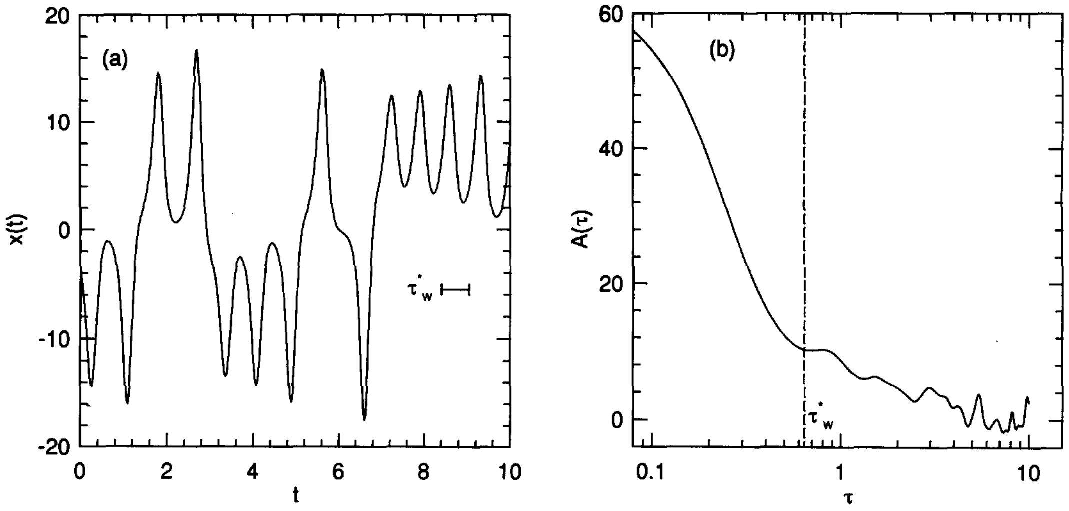

Fig. 2. (a) A sample of the Lorenz $x(t)$. The relative size of the critical window width $\tau_w^* \approx 0.63$ is indicated by a line segment. (b) The autocorrelation function of the Lorenz $x(t)$. This was estimated from a 50000 point time series, with $\Delta t = 0.01$. The critical window width is indicated by a dashed line.

correlation function need not have a minimum at finite $\tau$, whereas $\tau_w^*$ is finite for any analytic $x(t)$ with a spectrum of finite non-zero $\kappa_i$'s.

For numerical principal component analysis, we formed delay matrices by eq. (4), from a 10000 point subset of the time series, using $m = 9$ and various values of $\tau$. For each delay matrix, we made numerical calculations of the covariance matrix and its singular value decomposition (eqs. (5) and (8)). This gave numerical values for the eigenvectors $s_j$ and the eigenvalues $\sigma_j^2$.

Fig. 3 compares the numerical eigenvalues to those predicted from eqs. (44)–(49). The predictions agree well with the numerics for small $\tau_w$. Each eigenvalue $\sigma_j^2(\tau_w)$ exhibits $\tau_w^{2j}$ power-law scaling with $\tau_w$. For fixed $\tau_w$, the $\sigma_j^2$'s decrease exponentially with $j$. The latter effect is indicated by the roughly equal vertical spacing of eigenvalues at a fixed value of $\tau_w$. Both the power-law scaling and the exponential decrease are consequences of eqs. (44)–(49).

The domain of validity for the small-window solution is well-characterized by the critical window width, $\tau_w^*$. For the Lorenz $x(t)$, $\tau_w^* \approx 0.63$. The predicted eigenvalues in fig. 3 are fairly accurate until $\tau_w \approx \frac{1}{2}\tau_w^*$. The power-law scaling begins to break down here, and higher-order effects emerge in the eigenvectors (see fig. 4). For $\tau_w \geq \tau_w^*$, PCA is outside the domain of validity of the small-window solution. As $\tau_w$ becomes large, the numerical eigenvalues converge on $\langle x^2 \rangle$. This happens because, for the Lorenz $x(t)$, $\lim_{\tau \to \infty} A(\tau) = 0$: As $\tau$ increases, the off-diagonal elements $(\Xi_x)_{ij} = A((i-j)\tau)$ vanish, but the diagonal elements remain constant at $A(0)$. $\Xi_x$ approaches diagonal form and its eigenvalues approach its diagonal elements, $A(0) = \langle x^2 \rangle$[#9].

Thus there are three main regimes in fig. 3:

– *Small-window regime*, $\tau_w \ll \tau_w^*$. The eigenvalues are well-predicted by the small-window solution. Higher-order eigenvalues are small due to the exponential decrease with $j$, but they increase rapidly with $\tau_w$, scaling as $\tau_w^{2j}$.

– *Moderate-window, transition regime*, $\tau_w \approx \tau_w^*$. The eigenvalues diverge from the small-window solution, but they still decrease with $j$ roughly exponentially.

[#9] Note that the large-window behavior of PCA depends on how the limit $\tau_w \to \infty$ is reached: If $m$ is held constant and $\tau \to \infty$, the eigenvalues converge on $\langle x^2 \rangle$. If $\tau$ is held constant and $m \to \infty$, PCA becomes a discrete Fourier analysis, as noted in [12].

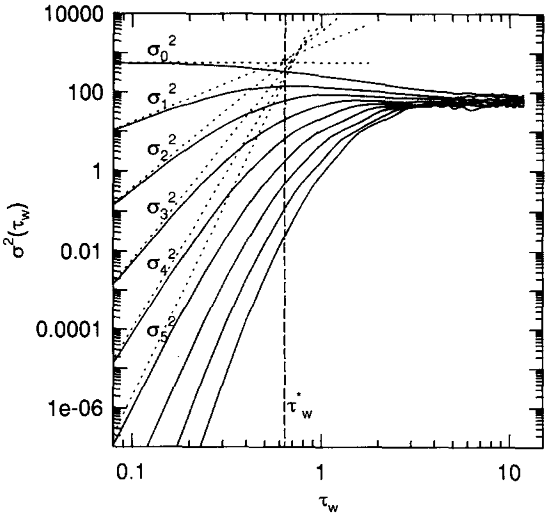

Fig. 3. Predicted and numerical eigenvalues as a function of $\tau_w$ for the Lorenz $x(t)$. Here $p = 4$ ($m = 9$). We plot all nine numerical eigenvalues with solid lines and six predicted eigenvalues with dotted lines. The critical window width is indicated by a vertical dashed line. The eigenvectors plotted in fig. 4 correspond to the eigenvalues $\sigma_0^2$, $\sigma_1^2$, and $\sigma_2^2$ in this plot in the range $0.08 \leq \tau_w \leq 1.28$. The time-axis of this plot coincides with the time-axis of the autocorrelation function in fig. 2b.

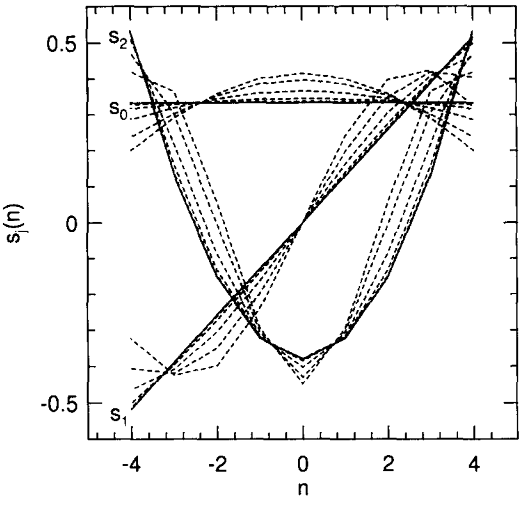

Fig. 4. Numerical eigenvectors and leading-order predictions for the Lorenz $x(t)$. We plot the first three eigenvectors $s_j(n)$ for $m = 9$ as a function of their coordinate-index $n$, using dashed lines for numerical eigenvectors and solid lines for predictions. By eq. (53), $s_j(n)$ is approximated to leading order by a discrete Legendre polynomial. Thus for predicted eigenvectors, we simply plotted discrete Legendre polynomials from the formulae in appendix A, which were derived from the recurrence relation of eq. (26). The numerical eigenvectors were calculated for $\tau_w = 0.08$, 0.16, 0.32, 0.64, and 1.28. The numerical eigenvectors for $\tau_w = 0.08$ are nearly indistinguishable from the discrete Legendre polynomials. For increasing $\tau_w$ there is increasing dicrepancy.

- *Large-window regime*, $\tau_w \gg \tau_w^*$. The eigenvalues converge on $\langle x^2 \rangle$.

Fig. 4 compares the first three numerical eigenvectors to leading-order predictions. By eq. 53, the order-1 term of the eigenvector $s_j(n)$ is the discrete Legendre polynomial $r_{j,p}(n)$. For small window widths, this term dominates and the numerical eigenvectors resemble discrete Legendre polynomials. For $\tau_w = 0.08$, the approximation $s_j(n) \approx r_{j,p}(n)$ is good. As the window width increases towards $\tau_w^*$, the higher-order terms in $\sigma_j^2(n)$ become significant, and the appearance of the eigenvectors is more complicated.

Broomhead and King originally noticed the resemblance between eigenvectors and Legendre polynomials in their application of PCA to numerical Lorenz data [9]. It is also noticeable in Vautard and Ghil's application of PCA to global surface air temperature data [14], though in the latter case the window width is large enough that second-order effects are significant.

Fig. 5 compares numerical and predicted phase portraits. The numerical portrait was obtained from projecting delay vectors onto the numerically calculated eigenvectors. The predicted portrait was obtained from eqs. (56) and (57). The agreement is good. We obtained similar results for portraits of other principal components.

## 4. Practical consequences

In section 2.2.2, we discussed how the delay reconstruction is characterized roughly by PCA. In this section, we use the theoretical understanding of PCA to put this characterization into analytic form, thereby obtaining a framework for

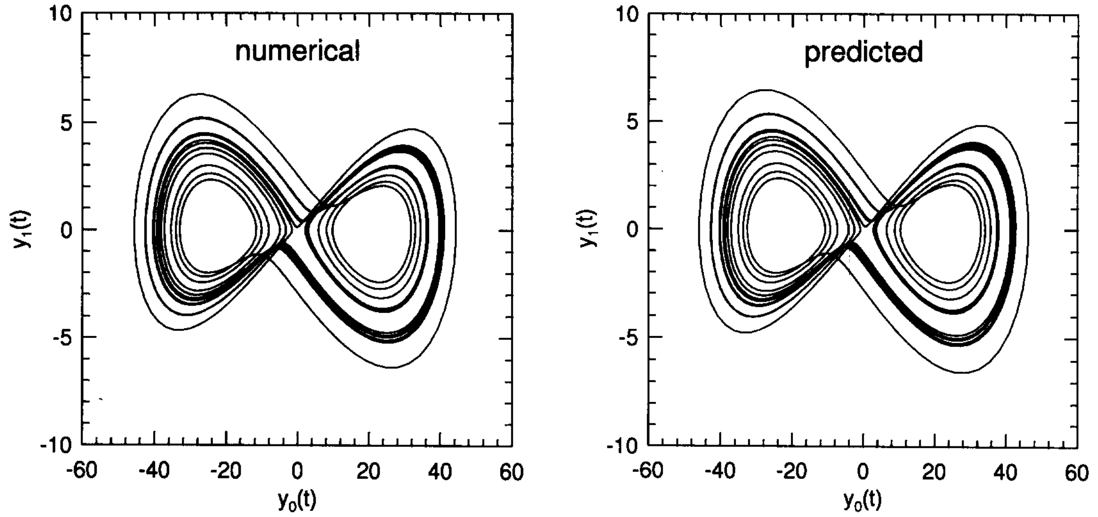

Fig. 5. Numerical and predicted phase portraits of principal components. We plot $y_1(t)$ vs. $y_0(t)$ for $p = 3$, $\tau_w = 0.06$. Note that the vertical axes are expanded relative to the horizontal.

choosing good delay reconstructions. In particular, we derive a guideline for choosing a good window width, and we re-examine the conditions under which principal components are good coordinates. We also show why PCA does not estimate dimension.

As a first step, we show that Legendre coordinates are close to principal components, in a precise sense. This allows us to phrase further discussions in terms of the simpler Legendre coordinates.

### *4.1. Closeness of Legendre coordinates to principal components*

In section 2.2.2, we discussed how PCA gives the optimal linear coordinate transformation for a fixed delay reconstruction, in terms of signal-to-noise ratios, because principal components have the maximum total variance of all projections from $m$ to $d < m$ coordinates. However, when $\tau_w$ is small, the Legendre coordinates are close to optimal, in the following sense: For a fixed set of delay vectors, consider the principal components $y^\dagger = x^\dagger S$ and the Legendre coordinates $w^\dagger = x^\dagger R$. Because $\Xi_y$ and $\Xi_w$ are related by a similarity transformation, their traces are equal.

$$\mathrm{Tr}\,\Xi_y = \sum_{i=0}^{m-1} \langle y_i^2 \rangle = \mathrm{Tr}\,\Xi_w = \sum_{i=0}^{m-1} \langle w_i^2 \rangle. \qquad (65)$$

For small $\tau_w$, $\langle y_i^2 \rangle = \mathscr{O}(\tau_w^{2i})$ and $\langle w_i^2 \rangle = \mathscr{O}(\tau_w^{2i})$. Therefore, for $1 \le d \le m - 1$,

$$\sum_{i=0}^{m-1} \langle y_i^2 \rangle = \sum_{i=0}^{d-1} \langle y_i^2 \rangle + \mathscr{O}\left(\tau_w^{2d}\right), \qquad (66)$$

$$\sum_{i=0}^{m-1} \langle w_i^2 \rangle = \sum_{i=0}^{d-1} \langle w_i^2 \rangle + \mathscr{O}\left(\tau_w^{2d}\right). \qquad (67)$$

The order-$\tau_w^2$ terms can be viewed as the variance lost in projecting from $m$-dimensional delays to $d$ principal components or $d$ Legendre coordinates. By transitivity of eqs. (65)–(67),

$$\sum_{i=0}^{d-1} \langle y_i^2 \rangle = \sum_{i=0}^{d-1} \langle w_i^2 \rangle + \mathscr{O}\left(\tau_w^{2d}\right). \qquad (68)$$

The difference in variance between projections of $d$ Legendre coordinates and $d$ principal components is on the same order as the variance lost in projecting from $m$-dimensional delays. It is two orders of $\tau_w$ higher than the variance of the smallest of the coordinates in the projection,

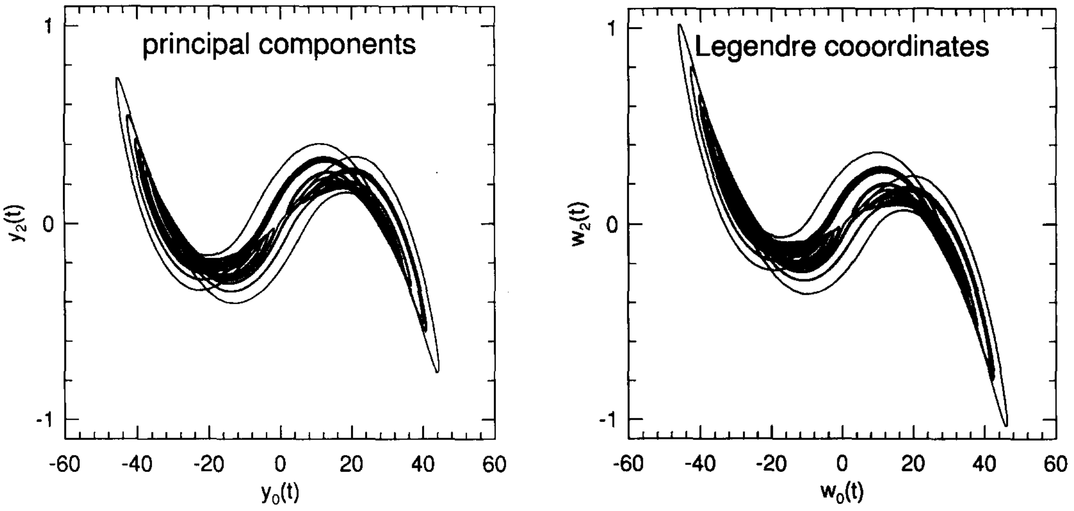

Fig. 6. Principal components and Legendre coordinates. We plot the principal components $y_2(t)$ versus $y_0(t)$ on the left. On the right are the corresponding Legendre coordinates, $w_2(t)$ versus $w_0(t)$. The portraits are similar, but they are not identical. They are related by a well-conditioned invertible linear transformation. The principal components were calculated numerically from $p = 3$, $\tau_w = 0.06$, delay vectors. The Legendre coordinates were obtained by projecting the same delay vectors onto discrete Legendre polynomials, given in appendix A.

$\langle y_{d-1}^2 \rangle = \mathscr{O}(\tau_w^{2d-2})$. Thus the Legendre coordinates $w(t)$ are close to optimal because they have nearly maximal variance.

The closeness of Legendre coordinates to principal components can also be seen from the recurrence relation of eq. (54). Principal components equal Legendre coordinates withcomponents of only lower-order Legendre coordinates subtracted off. Therefore, not only are all $m$ principal components and $m$ Legendre polynomials related by an invertible linear transformation, but so too are projections; i.e. the first $d < m$ principal components and the first $d < m$ Legendre coordinates are related by an invertible (and generally well-conditioned) linear transformation. Therefore attractors reconstructed from $d$ Legendre coordinates are very similar to attractors reconstructed with $d$ principal components. See fig. 6 for an illustration.

Legendre coordinates are a quick and not-so-dirty substitute for principal components when the window width is small. They may actually be superior in some situations: Principal components must be estimated numerically, which for small data sets may introduce substantial estimation errors. In contrast, the discrete Legendre polynomials do not need to be estimated. As a result, for short data sets in high dimensions, numerical PCA may actually be inferior to the alternative provided by discrete Legendre polynomials. We have not had the opportunity to investigate this in detail.

### 4.2. Choosing the window width

As first pointed out in ref. [2], the choice of the lag time $\tau$ requires a balance of two effects. Small lags cause the reconstruction to be stretched out along the identity line, which amplifies noise. On the other hand, for chaotic systems, large lags cause overly complicated reconstructions, which cause estimation error. The small-window solution to PCA provides insight towards the problem of balancing these effects.

The problems with small lags have been explained already. In section 3.1.2, we showed why small lags cause stretched-out attractors, and in section 2.2.2, we showed why stretched-out at-

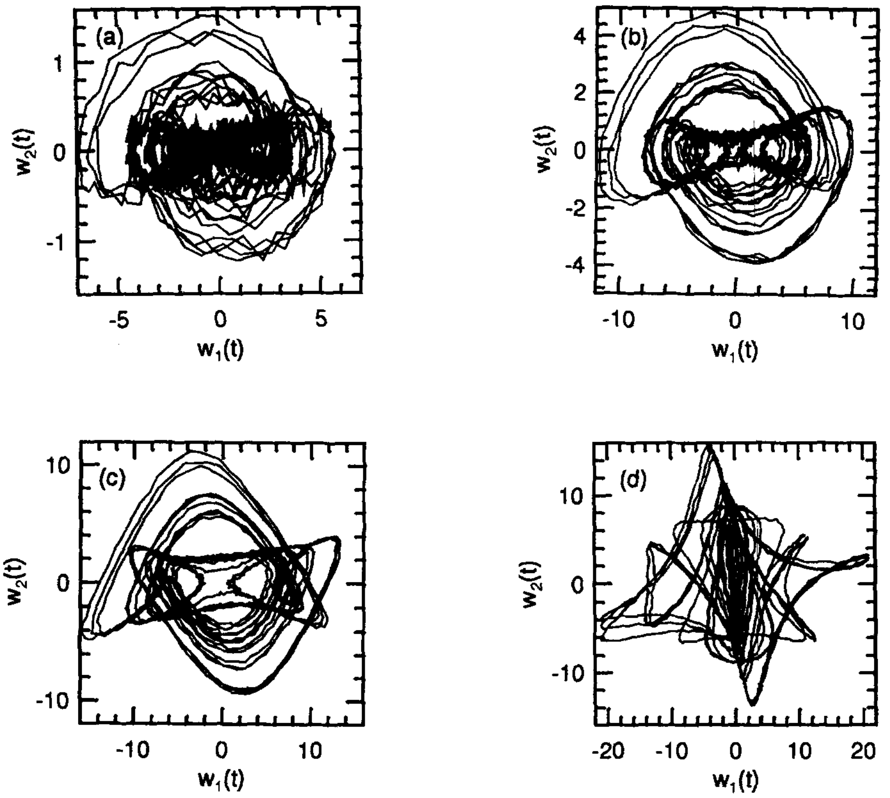

Fig. 7. Phase portraits of Legendre coordinates for varying window widths. The time series the Lorenz $x(t)$ with 1% additive Gaussian IID noise. In all four plots, $p = 1$ ($m = 3$), and we show $w_2(t)$ versus $w_1(t)$. In (a) $\tau_w \approx \frac{1}{8}\tau_w^*$, (b) $\tau_w \approx \frac{1}{4}\tau_w^*$, (c) $\tau_w \approx \frac{1}{2}\tau_w^*$, (d) $\tau_w = 0.64 \approx \tau^*$. As $\tau_w$ increases, signal-to-noise ratios increase and the geometry stays constant until $\tau_w$ reaches $\frac{1}{2}\tau_w^*$. Past this value, signal-to-noise ratios are nearly constant and the geometry becomes increasingly complex.

tractors have poor signal-to-noise ratios. The problems with large lags may be seen by imagining a delay reconstruction with a fixed dimension and a variable window width. As the window width is increased, the relation between the first and last coordinates of a delay vector is governed by an increasingly longer-term iteration of the dynamics. Consequently, if the dynamics are chaotic, the delay reconstruction acquires increasing complexity, and any numerical analysis on the reconstruction, such as dimension estimation or modeling dynamics, requires an increasing number of datapoints to maintain a given accuracy [4].

Of course, the optimal balance between noise and complexity depends how the reconstruction is used. For example, a dimension calculation might be more sensitive to noise and less sensitive to the complexity of the reconstruction than nonlinear predictive modeling. However, it is possible to discuss the balance in general, context-independent manner, and to derive a balance which is generally good. For simplicity, we will phrase the discussion in terms of Legendre coordinates, instead of principal components. We will also temporarily assume that the sampling time can be made arbitrarily small. The ramifications of a minimum sampling time will be discussed in section 4.3.

Consider how signal-to-noise ratios of Legendre coordinates vary with $\tau_w$: As with principal components, the signal-to-noise ratio of a Legendre coordinate is proportional to the square root of its variance. For $\tau_w \ll \tau_w^*$, eq. (28) and substituting with eq. (38) gives

$$\langle w_j^2 \rangle = \left( \frac{c_j(p)\tau_w^j}{2^j(j!)} \right)^2 \kappa_j + \mathscr{O}\left(\tau_w^{2j+2}\right). \qquad (69)$$

If we begin with $\tau_w \ll \tau_w^*$, increasing $\tau_w$ increases variances and therefore improves signal-to-noise ratios. As $\tau_w$ nears $\tau_w^*$, the small-window analysis breaks down, which causes the variance of $w_j$ to break away from scaling according to $\tau_w^{2j}$, and the improvement in signal-to-noise ratios begins to taper off.

On the other hand, consider how changing $\tau_w$ changes the complexity of the attractor: For $\tau_w \ll \tau_w^*$, the geometry of the delay reconstruction is determined by eq. (28). In this regime, the reconstruction is decomposed into a prefactor that depends on $\tau_w$, and the derivatives $x^{(j)}(t)$, which do not. The prefactor determines the relative scale of each Legendre coordinate and its signal-to-noise ratio, as described above, but the derivatives determine the scale-independent nonlinear structure of the attractor. Since the derivatives are independent of $\tau_w$, the scale-independent structure is constant for $\tau_w \ll \tau_w^*$. As $\tau_w$ nears $\tau_w^*$, the higher-order terms in eq. 28 become significant. These terms represent the higher-order derivatives of $x(t)$, whose functional relationships are more complicated. Because of this, the reconstruction is relatively simple until $\tau_w \approx \tau_w^*$ and increasingly complicated thereafter.

Thus the small-window solution provides insight towards both sides of the balance: Increasing $\tau_w$ towards $\tau_w^*$ increases the signal-to-noise ratios, while the complexity of the reconstruction remains approximately constant. As $\tau_w$ nears $\tau_w^*$, the small-window solution breaks down, and the signal-to-noise ratios increase less rapidly, while the complexity begins to increase. Thus a good balance between signal-to-noise ratios and complexity can be obtained by using a window width less than but near to $\tau_w^*$. In other words, good delay reconstructions sit on the upper edge of the small-window solution.

This balance is illustrated in fig. 7. In this figure, we show phase portraits of Legendre coordinates reconstructed from a time series of the Lorenz $x(t)$ with 1% additive Gaussian IID noise. For the Lorenz $x(t)$, $\tau_w^* \approx 0.63$. Fig. 7 shows a phase portrait of $w_2(t)$ vs. $w_1(t)$ for $p = 1$ $(m = 3)$. In (a) $\tau_w = 0.08 \ll \tau_w^*$, and the reconstruction is very noisy. In (b), increasing $\tau_w$ to 0.16 increases the signal-to-noise ratios of both coordinates, without changing the geometry of the reconstructed attractor. Note that the scales of the plot have changed. In (c) $\tau_w = 0.32 \approx \frac{1}{2}\tau_w^*$, and again the signal-to-noise ratios are better. At this point, the geometry of the reconstructed attractor begins to change, but it is still close to the simple object seen in (a) and (b). In (d), $\tau_w = 0.64 \approx \tau_w^*$. The signal-to-noise ratios are not much better, because the small-window analysis has broken down and $w_j(t)$ no longer scales with $\tau_w^j$. The geometry of the attractor is no longer governed by its derivatives, so the attractor has become complicated. It is clear that $\tau_w = \frac{1}{2}\tau_w^*$ as shown in (c) is a good value for $\tau_w$. As predicted, it is less than but near to $\tau_w^*$.

In general, the precise value of $\tau_w$ which gives the best balance depends on the application, and the best way to choose $\tau_w$ is with a numerical optimization. For example, for predictive modeling, one would minimize the prediction error over $\tau_w$. Since numerical optimizations can be hastened by a good starting point, we recommend starting the search at a window width less than but near to $\tau_w^*$.

Numerical optimization is not always an option. For example, in dimension calculations, there is no known error function to minimize. As an intermediate solution for such cases, we recommend making a log–log plot of $\{\langle w_j^2 \rangle\}$ versus $\tau_w$ (a Legendre-coordinate version of fig. 3). Signal-to-noise ratios can be ascertained from the magnitudes of $\{\langle w_j^2 \rangle\}$; the emergence of higher-order effects are indicated by the deviations of

$\{\langle w_j^2\rangle\}$ from $\tau_w^{2j}$ scaling. The desired balance between the effects can be judged by eye.

Lastly, when only a swift ball-park estimate is desired, we recommend setting

$$\tau_w = \mu\tau_w^* = 2\mu\sqrt{\frac{3\kappa_0}{\kappa_1}} = 2\mu\sqrt{\frac{3\langle x^2\rangle}{\langle (\mathrm{d}x/\mathrm{d}t)^2\rangle}}\,, \tag{70}$$

where $\mu$ is a fixed constant less than but on the order of 1. For example, $\mu = \frac{1}{2}$ gave good results in fig. 7. Note that the derivatives in eq. (70) can be computed by a number of methods. We recommend using the discrete Legendre polynomials because of their ability to average out noise[#10].

Once the window width is fixed, we must choose values of $\tau$ and $p$ which produce the given window. By eq. (69) and the scaling of $c_j(p)$, the variances of the principal components scale as $p$. Therefore in the idealized case of an infinite sampling rate, arbitrarily large signal-to-noise ratios can be obtained by letting $p \to \infty$ and $\tau \to 0$, keeping $\tau_w = 2p\tau$ fixed.

We then recommend projecting the delay reconstruction onto the first $q < m$ discrete Legendre polynomials, where $q$ is the smallest number of derivatives which form an embedding. Unfortunately, there is no simple rule for determining $q$ (see section 4.4).

### 4.3. Ramifications of minimum sampling time and finite noise

Of course, letting $p \to \infty$ with a fixed window is not a practical recommendation, since we cannot decrease the lag time $\tau$ below the sampling time $\Delta t$. This limits the variance that can be gained by increasing $p$. In this case, we should set $\tau = \Delta t$ and $p = \tau_w/2\tau$, where $\tau_w$ has been chosen by the methods of section 4.2. The question then arises whether these parameters result in an approximately deterministic state space.

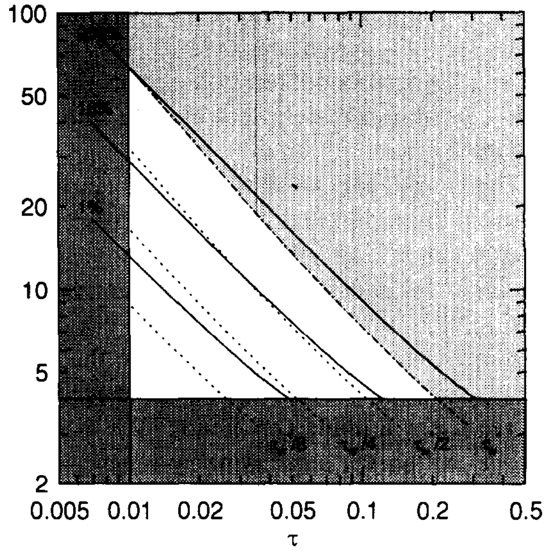

Fig. 8. The parameter space of a Lorenz $x(t)$ reconstruction, delay dimension $m$ versus lag time $\tau$. Dark grey regions represent restricted parameters ($\tau < \Delta t = 0.01$ or $m < q = 4$). The light grey region represents parameters in the large window regime ($(m-1)\tau > \tau_w^* \approx 0.63$). The white region represents the set of small-window parameters which form embeddings. Dashed lines indicate constant window widths, and solid lines indicate parameters at which the signal-to-noise ratio of $w_{3=q-1}$ is unity, for three different noise levels, calculated from eq. (71).

By eq. (19), if $\sigma_{q-1}^2 \gg \langle\eta^2\rangle$, then a state space of $q$ principal components is approximately deterministic. As a rough approximation, we can replace $\sigma_{q-1}^2$ with $\langle w_{q-1}^2\rangle$ if $\tau_w \ll \tau_w^*$. Then substituting with eq. (69) gives

$$\tau_w^{q-1}c_{q-1}(p) \gg \frac{2^{q-1}}{(q-1)!}\sqrt{\frac{\langle\eta^2\rangle}{\kappa_{q-1}}}\,. \tag{71}$$

From this inequality it is possible to determine, for a given noise level, which parameters result in approximately deterministic state spaces. This is illustrated in fig. 8, where we plot the parameter space $(m, \tau)$ of the delay reconstruction for a Lorenz $x(t)$ sampled at $\Delta t = 0.01$. The solid, diagonal lines represent parameters for which the two sides of eq. (71) are equal for a given noise

[#10] This might require iteration, since the discrete Legendre polynomials must be applied over a window width. We recommend starting with a small window, then increasing the window towards $\tau_w^*$ as the estimate of $\tau_w^*$ becomes more precise.

level[#11] and $q = 4$[#12]. Thus each solid line represents, for the given noise level, the boundary between noise-dominated state spaces (below the line) and approximately deterministic state spaces (above the line).

The shaded regions in fig. 8 represent parameters which are restricted by other considerations. The region $\tau < \Delta t = 0.01$ is excluded because the lag time cannot decrease below the sampling time. The region $m < q = 4$ is excluded because the minimum embedding dimension for the Lorenz $x(t)$ is four. The region $\tau_w < \tau_w^*$ is shaded lightly to represent a milder restriction: We would like to stay within the small window regime in order to get a simple reconstruction. Thus the white region represents the accessible set of small-window parameters which will form embeddings.

Putting all the restrictions together, we see that as the noise level increases, less and less of the white region is available for approximately deterministic reconstructions. For example, a 1% noise level covers the bottom corner of the white region, so that reconstructions with $\tau_w \ll \frac{1}{4}\tau_w^*$ are noise-dominated, but reconstructions with $\frac{1}{4}\tau_w^* < \tau_w < \tau_w^*$ are approximately deterministic. (Dashed lines indicate parameters $(m, \tau)$ with constant window width.) Raising the noise level to 16% reduces the range of good window widths to $\frac{1}{2}\tau_w^* < \tau_w < \tau_w^*$.

Note that the constant window-width lines are not exactly parallel to the noise-level lines. For example, the $\frac{1}{2}\tau_w^*$ line lies below the 16% noise line at $(m, \tau) = (4, 0.1)$, but crosses over it as $\tau$ decreases and $m$ increases. At $(m, \tau) = (30, 0.01)$, the $\frac{1}{2}\tau_w^*$ line reaches its maximum extension above the 16% noise line. This is a result of the $\sqrt{p}$ scaling of $c_4(p)$, and it is why we recommend setting reconstruction parameters with the minimum lag time and maximum dimension for a given window. The 256% noise line, however, lies entirely above the region of accessible small-window parameters; therefore it is not possible to reconstruct an approximately deterministic state space from a Lorenz $x(t)$ with this noise level and this sampling time by using Legendre coordinates.

In our example, the sampling time is roughly two orders of magnitude smaller than the critical window width. In practical situations, we might have a coarser sampling of the time series, with only one order of magnitude difference, or less. If the sampling time in our example were increased to $\Delta t = 0.1$, this would shift the $\tau$ lower bound from 0.01 to 0.1, the white region of accessible small-window parameters would be much smaller, and slightly lower noise levels would obscure reconstructions with the same window width. If $\Delta t$ reached 0.2, the white region would vanish, and approximately deterministic small-window reconstructions would be impossible at any noise level. Since Legendre coordinates require small windows, this would effectively rule out a Legendre $\langle \eta^2 \rangle$ coordinate reconstruction.

Note that this analysis relies only on the variance of the noise, $\langle \eta^2 \rangle$; the minimum embedding dimension for derivatives, $q$; the critical window width, $\tau_w^*$; and $\kappa_{q-1}$. If these quantities are known or can be estimated for a given time series, the parameter space can be mapped out as in fig. 8.

### 4.4. PCA and dimension estimation

The possibility of a relationship between the singular spectrum and the dimension of the underlying dynamical system has been discussed in the literature [10, 12, 15]. As discussed in section 2.2.2, the singular spectrum often reaches a plateau, which can be attributed to noise on the time series, of the form of eq. (15). The eigenvalues above this plateau are called *significant eigenvalues*. Broomhead and King [9] claimed that the number of significant eigenvalues reflects the number of linear modes in $x(t)$ that lie above the noise level. This is consistent with our results

[#11] By 1% noise, for example, we mean $\sqrt{\langle \eta^2 \rangle / \langle x^2 \rangle} = 0.01$.

[#12] In the small-window limit, the minimum embedding dimension for the Lorenz $x(t)$ is four. This can be seen by calculating the function $(x^{(0)}, x^{(1)}, x^{(2)}, x^{(3)}) = f(x, u, v)$ from the Lorenz equations (62)–(64), and then inverting $f$. Note that this casts doubt on whether the $m = 3$ plots in fig. 7 are embeddings.

(but we note that this number is dependent on the choice of window width). A stronger claim has been discussed in the literature, namely, that the number of significant eigenvalues reflects the dimension of the manifold which embeds the dynamical system. The stronger claim has been rejected on grounds of genericity [12] and by counter-example [10, 15].

The small-window solution provides a convenient framework in which to discuss the stronger claim analytically. Let us consider noiseless time series, ignoring at first the higher-order terms of the small-window solution. Suppose that the singular spectrum reaches zero at finite $k$. That is, $\sigma_k^2 = \langle y_k^2 \rangle = 0$. For a stationary time series, this means that $y_k(t)$ must be identically zero. But by eq. (55), $y_k(t) = 0$ implies that

$$x^{(k)}(t) = \sum_{i=0}^{k-1} a_i x^{(i)}(t) \tag{72}$$

for some set of $a_i$'s. Eq. (72) represents $k$-dimensional linear dynamics. Therefore, the singular spectrum vanishes at finite $k$ only for time series from linear dynamics. For nonlinear systems, no derivative is identically a linear combination of lower-order derivatives, so no eigenvalue can vanish. Geometrically, this means a delay-vector trajectory from a linear dynamical system occupies a fixed-dimensional linear subspace of $\mathbb{R}^m$, while nonlinear systems produce trajectories that span $\mathbb{R}^m$, regardless of the choice of $m$. The same is true when the higher-order terms in eq. (55) are included, because these terms are composed of higher-order derivatives of $x(t)$, which are subject to the same argument.

When noise is included in the time series, it induces a lower bound on the singular spectrum, as discussed in section 2.2.2. The eigenvalues decrease exponentially as $\tau_w^{2j}$, and the number of significant eigenvalues is determined by the intersection of the decreasing part and the noise floor.

In ref. [4], we gave a geometrical description which illustrated the complications of estimating the minimum embedding dimension. In general, the minimum embedding dimension must be estimated by a nonlinear algorithm.

## 5. Conclusion

### 5.1. Open questions

Several problems are left outstanding: We recommend Legendre coordinates for algorithms such as dimension estimation, but Legendre coordinates may be used as they are, or they can be rescaled so each has the same variance. Straight Legendre coordinates have the advantage of isotropic noise, but rescaled Legendre coordinates seem more appropriate for local analysis techniques. It is unclear which is better.

As stated earlier, the good delay reconstructions sit on the upper edge of the small-window solution. The small-window solution could be extended towards this edge by quantifying the higher-order effects in Legendre coordinates or principal components. This would also shed more light on reconstructions which are forced into the moderate-window regime by sampling limitations.

There are interesting but undeveloped connections between this paper and ref. [4]. For example, the signal-to-noise ratios discussed in this paper are clearly related to the distortion defined in ref. [4].

### 5.2. Summary

The small-window solution to PCA explains several known characteristics of PCA: the resemblance of eigenvectors to Legendre polynomials, the exponential decrease of the singular spectrum, and the relative insensitivity of the singular spectrum to changes in $m$ and $\tau$ if $\tau_w = (m-1)\tau$ is fixed. Because PCA becomes equivalent to Fourier analysis in the limit of large windows, we also have the interesting result that as the window width tends from zero to infinity with a small fixed $\tau$, the eigenvectors of PCA go from Legendre polynomials to trigonometric functions.

We have clarified the relationships between delays, derivatives, and principal component analysis, and we have shown why the number of significant eigenvalues is unrelated to the dimension of the underlying system. We have shown that principal component analysis is a useful coordinate transformation, and we have derived explicit criteria that predict when principal components are above the noise floor and better than delays.

We have derived a set of discrete Legendre polynomials which are useful both for state space reconstruction and, in a more general context, for stable estimates of derivatives of discretely sampled functions. We have described analytically the counteracting effects one must balance when choosing a window width. We have outlined a procedure for choosing a window width that gives a good balance. For situations that require only a rough estimate of the best window width, we have given a simple, analytic formula.

## Appendix A. Discrete and continuous Legendre polynomials

The first six discrete Legendre polynomials for $n \in [-p, p]$ (i.e. $\boldsymbol{r}_j \in \mathbb{R}^{m=2p+1}$) are

$$r_{0,p}(n) = \frac{1}{c_0(p)},$$

$$r_{1,p}(n) = \frac{1}{pc_1(p)}n,$$

$$r_{2,p}(n) = \frac{1}{p^2c_2(p)}\left(n^2 - \frac{p(p+1)}{3}\right),$$

$$r_{3,p}(n) = \frac{1}{p^3c_3(p)}\left(n^3 - n\frac{3p^2+3p-1}{5}\right),$$

$$r_{4,p}(n) = \frac{1}{p^4c_4(p)}\left(n^4 - n^2\frac{6p^2+6p-5}{7} + \frac{3p(p^2-1)(p+2)}{35}\right),$$

$$r_{5,p}(n) = \frac{1}{p^5c_5(p)}\left(n^5 - n^3\frac{5(2p^2+2p-3)}{9} + n\frac{15p^4+30p^3-35p^2-50p+12}{63}\right). \tag{73}$$

These formulae were generated from the recurrence relation for discrete Legendre polynomials, given by eq. (26). Discrete Legendre polynomials for $n \in [-p+1, p]$ (i.e. $\boldsymbol{r}_j \in \mathbb{R}^{m=2p}$) can be obtained by making the appropriate alteration of recurrence relation.

The normalization constants $c_j(p)$ are

$$\begin{aligned}
c_0(p) &= (2p+1)^{1/2},\\
c_1(p) &= [(2p+1)(p+1)/3p]^{1/2},\\
c_2(p) &= \tfrac{1}{3}[(4p^2-1)(p+1)(2p+3)/5p^3]^{1/2},\\
c_3(p) &= \tfrac{1}{5}[(4p^2-1)(p^2-1)(2p+3)\\
&\quad\times(p+2)/7p^5]^{1/2},\\
c_4(p) &= \tfrac{2}{35}[(4p^2-1)(p^2-1)(4p^2-9)(p+2)\\
&\quad\times(2p+5)/9p^7]^{1/2},\\
c_5(p) &= \tfrac{2}{63}[(4p^2-1)(p^2-1)(4p^2-9)(p^2-4)\\
&\quad\times(2p+5)(p+3)/11p^9]^{1/2}.
\end{aligned} \tag{74}$$

The normalization constants were derived from the condition $\Sigma_{n=-p}^{p} r_{j,p}^2(n) = 1$.

In this appendix, we prove that $r_{j,p}(n)$ approaches the $j$th Legendre polynomial in the limit $p \to \infty$, except for a difference of normalization. We also demonstrate that $r_{j,p}(n)$ reduces to a finite differencing filter when $p$ takes on the lowest value allowed.

First we prove the continuous-limit equivalence by induction. The discrete Legendre polynomials are normalized to have unit length in $\mathbb{R}^{2p+1}$, whereas the Legendre polynomials $P_j(\chi)$ are normalized to reach unity at $\chi = \pm 1$. To start the induction, we put $r_{j,p}(n)$ the normalization of $P_j(\chi)$: Define $r'_{j,p}(n) = r_{j,p}(n)/r_{j,p}(p)$, so that

$r'_{j,p}(\pm p) = 1$. Then by eqs. (73) and (74),

$$r'_{0,p}(n) = 1,$$

$$r'_{2,p}(n) = \frac{3n^2 - p^2 - p}{2p^2 - p}, \qquad (75)$$

$$r'_{1,p}(n) = \frac{n}{p},$$

$$r'_{3,p}(n) = \frac{5n^3 - 3p^2 n - 3pn + n}{2p^3 - 3p^2 + p} \qquad (76)$$

Letting $\chi = n/p$, taking the limit $p \to \infty$, and writing $r'$ as a function of $\chi$, we get

$$\lim_{p\to\infty} r'_{0,p}(\chi) = 1,$$

$$\lim_{p\to\infty} r'_{2,p}(\chi) = \tfrac{1}{2}(3\chi^2 - 1), \qquad (77)$$

$$\lim_{p\to\infty} r'_{1,p}(\chi) = \chi,$$

$$\lim_{p\to\infty} r'_{3,p}(\chi) = \tfrac{1}{2}(5\chi^3 - 3\chi). \qquad (78)$$

By inspection, $\lim_{p\to\infty} r'_{j,p}(\chi) = P_j(\chi)$ for $0 \le j \le 3$, where $P_j(\chi)$ is the $j$th Legendre polynomial.

To continue the induction, we will show the equivalence of the recurrence relations for $r_{j,p}(n)$ and $P_j(\chi)$. This is more convenient using the unit-length normalization: Define

$$P'_j(\chi) = \frac{1}{\sqrt{\int_{-1}^{1} P_j^2(\xi)\,\mathrm{d}\xi}} P_j(\chi). \qquad (79)$$

so that $\int_{-1}^{1} P_j'^2(\chi)\,\mathrm{d}\chi = 1$. From real analysis (see, for example, ref. [16]), any polynomial of $j$th degree can be decomposed into a linear combination of the first $j+1$ Legendre polynomials. For the $j$th degree polynomial $\chi^j$, the decomposition is

$$\chi^j = \sum_{i=0}^{j} P_i(\chi) \frac{\int_{-1}^{1} \xi^j P_i(\xi)\,\mathrm{d}\xi}{\int_{-1}^{1} P_i^2(\xi)\,\mathrm{d}\xi}, \qquad (80)$$

$$= \sum_{i=0}^{j} P'_i(\chi) \int_{-1}^{1} \xi^j P'_i(\xi)\,\mathrm{d}\xi. \qquad (81)$$

Rearranging terms gives a recurrence relation for renormalized Legendre polynomials,

$$P'_j(\chi) = \frac{1}{\int_{-1}^{1} \xi^j P'_j(\xi)\,\mathrm{d}\xi} \times \left( \chi^j - \sum_{i=0}^{j-1} P'_i(\chi) \int_{-1}^{1} \xi^j P'_i(\xi)\,\mathrm{d}\xi \right). \qquad (82)$$

This is the continuum limit of eq. 26, the recurrence relation for $r_{j,p}(n)$. Therefore, the discrete Legendre polynomials approach the Legendre polynomials in the limit of large $p$, except for a difference of normalization.

On the other hand, consider $r'_{j,p}(n)$ with $p$ as small as possible. By eq. (26), $r_{j,p}(n)$ exists only for $2p \ge j$. Thus for $j = 0, 2, 4$, the minimum $p$'s are $p = 0, 1, 2$, respectively. In vector form, the renormalized discrete Legendre polynomials for these $(j, p)$ pairs are

$$r'_{0,0}(n) = (1)^{\dagger}, \qquad (83)$$

$$r'_{2,1}(n) = (1, -2, 1)^{\dagger}, \qquad (84)$$

$$r'_{4,2}(n) = (1, -4, 6, -4, 1)^{\dagger}. \qquad (85)$$

By inspection these are finite differencing filters for the zeroth, second, and fourth derivatives. (For $r'_{j,p}(n)$ with odd $j$, this reduction occurs when $m$ is even.)

We can think of finite-differencing as a special case of the discrete Legendre polynomials. This is helpful because it shows why finite-differencing is generally not the best method for estimating derivatives of discretely sampled, noisy functions: Suppose that $x(t)$ is sampled at intervals $\Delta t$, and that $\Delta t \ll \tau_w^*$. Then the filters given by eqs. (83), (84) will provide estimates of $x$, $x^{(2)}$, and $x^{(4)}$, according to eq. (28), with $(m, \tau_w)$ equalling $(1, 0)$, $(3, 2\Delta t)$ and $(5, 4\Delta t)$, respectively. By eq. (69), the signal-to-noise ratios of these estimates scale as $m^{1/2}\tau_w^j$. Compare this to the $m = 5$ renormalized discrete Legendre polynomials for $x$, $x^{(2)}$, and $x^{(5)}$,

$$r'_{0,2}(n) = (1,1,1,1,1)^{\dagger}, \tag{86}$$

$$r'_{2,2}(n) = (1, -\tfrac{1}{2}, -1, -\tfrac{1}{2}, 1)^{\dagger}, \tag{87}$$

$$r'_{4,2}(n) = (1, -4, 6, -4, 1)^{\dagger}. \tag{88}$$

In each of these estimates, $(m, \tau_w) = (5, 4\Delta t)$. The signal-to-noise ratios of these estimates are better than those of the finite-difference estimates by factors of $\sqrt{5}$, $2\sqrt{\frac{5}{3}}$, and 1, respectively. Further, eqs. (86)–(88) give estimates with uncorrelated noise. Therefore, the general discrete Legendre polynomials generally provide better estimates of derivatives than finite-difference estimators[#13].

As the discreteness parameter $p$ ranges between its lowest allowed values and infinity, the $r_{j,p}(n)$'s form a bridge between finite differencing and continuous Legendre polynomials, retaining the advantages of each. The discrete Legendre polynomials estimate derivatives from a discretely sampled functions, like finite-differencing, but with the noise-reductive averaging that one would get from projecting continuous functions onto continuous Legendre polynomials. The discrete Legendre polynomials estimate derivatives more accurately than discretely sampled continuous Legendre polynomials, since the latter are not exactly orthogonal when sampled discretely (consider how poorly $(1, -0.5, 1)$, a discrete sample of $P_2(\chi)$, would approximate a second derivative). Fig. 9 shows the transition of $r'_{2,p}(n)$ from finite differencing to the second Legendre polynomial as $p$ increases.

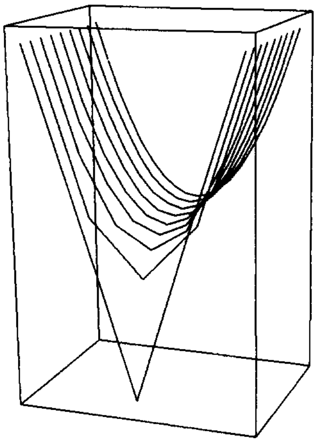

Fig. 9. The second discrete Legendre polynomial. We plot $r'_{2,p}(n)$ vertically, $n/p$ horizontally, and the discreteness parameter $p$ increasing towards the back. For $p = 1$, $r'_{2,p}(n) = (1, -2, 1)$, which is the finite-difference filter for the second derivative. In the continuous limit, $p \to \infty$, $r'_{2,p}(n)$ approaches the second Legendre polynomial.

## Appendix B. Rate of increase of $\langle (x^{(i)})^2 \rangle$ with $i$

Eq. (36) has an interesting implication on the average squared values of derivatives of bounded analytic functions. The Schwarz inequality for random variables $\xi$ and $\chi$ is

$$(\langle \xi\chi \rangle)^2 \le \langle \xi^2 \rangle \langle \chi^2 \rangle. \tag{89}$$

If we consider $x^{(i)}(t)$ and $x^{(i+2)}(t)$ as random variables whose distributions are defined by the function $x(t)$, the Schwarz inequality and eq. (36) (setting $j = i + 2$) yield

$$\frac{\left\langle \left(x^{(i+1)}\right)^2 \right\rangle}{\left\langle \left(x^{(i)}\right)^2 \right\rangle} \le \frac{\left\langle \left(x^{(i+2)}\right)^2 \right\rangle}{\left\langle \left(x^{(i+1)}\right)^2 \right\rangle} \quad \text{for } i \ge 0, \tag{90}$$

provided that the denominators are non-zero and that $x(t)$ is a bounded analytic function with bounded derivatives. Note that if $\langle (x^{(0)})^2 \rangle$ and $\langle (x^{(1)})^2 \rangle$ are non-zero, then by induction $\langle (x^{(i)})^2 \rangle$ is non-zero for all $i > 0$. Therefore, for bounded analytic functions $x(t)$, with non-zero variance in $x(t)$ and $x^{(1)}(t)$, the variance of all higher-order derivatives is non-zero, and the ratio between the

[#13] Three caveats: (1) The $m^{1/2}\tau_w^j$ scaling breaks down as $\tau_w$ nears $\tau_w^*$. (2) Discrete Legendre polynomials are symmetric filters, so if the value of the derivative at either end of the window is needed, it may be better to keep the window as small as possible for a given derivative. (3) If one is concerned with the accuracy of estimates of derivatives at a single point $t_0$, then the window width recommended for best results on average may not be appropriate.

variances of the $(i+1)$th and $i$th derivatives is monotonically non-decreasing with $i$.

In terms of $\{\kappa_i\}$, eq. (90) is

$$\frac{\kappa_{i+1}}{\kappa_i} \le \frac{\kappa_{i+2}}{\kappa_{i+1}}. \tag{91}$$

## Appendix C. Diagonalization of $\Xi_w^e$

We break the diagonalization of $\Xi_w^e$ into two parts: First, we show how to diagonalize a real symmetric matrix with exponentially decreasing elements. Second, we examine the application of this algorithm to $\Xi_w^e$.

### *C.1. The symmetric eigenvalue problem for exponentially decreasing matrices*

Let $A$ be an $m \times m$ real symmetric matrix of the form

$$A_{ij} = a_{ij}\epsilon^{i+j} + \mathscr{O}(\epsilon^{i+j+1}), \tag{92}$$

where $a_{ij} = \mathscr{O}(1)$ and $\epsilon \ll 1$. In this appendix we show that $A$ can be diagonalized to leading order by one sweep of the cyclic Jacobi method, and that leading-order approximations to its eigenvalues and eigenvectors can be calculated in closed form.

The cyclic Jacobi method is a numerical algorithm for diagonalizing real symmetric matrices [17]. Generally, it consists of a series of similarity transformations,

$$A \to A^{(1)} = J_0^\dagger A J_0 \to A^{(2)} = J_1^\dagger A^{(1)} J_1 \quad \text{etc.} \tag{93}$$

each of which zeroes a single off-diagonal element. Successive transformations generally undo previously set zeroes, so the numerical algorithm sweeps through the matrix repeatedly, zeroing and rezeroing the off-diagonal elements, in a fixed order, until the matrix is diagonalized to the required precision (the method can be shown to converge [17]). Each $J_n$ in the cyclic Jacobi method is an $m \times m$ Givens rotation $J(k,l,\theta)$,

$$J(k,l,\theta) = \begin{array}{cc} \begin{pmatrix} 1 & & & & \\ & \cos\theta & \cdots & \sin\theta & \\ & \vdots & & \vdots & \\ & -\sin\theta & \cdots & \cos\theta & \\ & & & & 1 \end{pmatrix} & \begin{array}{c} \\ k \\ \\ l \\ \\ \end{array} \\ \begin{array}{ccccc} \quad & k & \qquad & l & \quad \end{array} & \end{array}. \tag{94}$$

The diagonal elements of a Givens rotation are unity, except $[J(k,l,\theta)]_{kk} = [J(k,l,\theta)]_{ll} = \cos\theta$, and the off-diagonal elements are zero, except $[J(k,l,\theta)]_{kl} = -[J(k,l,\theta)]_{lk} = \sin\theta$.

By the definition of the $J(k,l,\theta)$, the elements of $A^{(1)} = J^\dagger(k,l,\theta)\,AJ(k,l,\theta)$ are given by

$$\begin{aligned} A_{ij}^{(1)} &= A_{ij} \quad \text{for } i \ne k,l \text{ and } j \ne k,l, \\ A_{jk}^{(1)} &= A_{kj}\cos\theta - A_{lj}\sin\theta \quad \text{for } j \ne k,l, \\ A_{il}^{(1)} &= A_{ik}\cos\theta + A_{il}\sin\theta \quad \text{for } i \ne k,l, \\ A_{kk}^{(1)} &= A_{kk}\cos^2\theta + A_{ll}\sin^2\theta - 2A_{kl}\sin\theta\cos\theta, \\ A_{ll}^{(1)} &= A_{ll}\cos^2\theta + A_{ll}\sin^2\theta + 2A_{kl}\sin\theta\cos\theta, \\ A_{kl}^{(1)} &= A_{kl}(\cos^2\theta - \sin^2\theta) \\ &\quad + (A_{kk} - A_{ll})\sin\theta\cos\theta. \end{aligned} \tag{95}$$

$A^{(1)}$ is symmetric, so elements not listed here can be found from their symmetric counterparts; for example $A_{li}^{(1)} = A_{il}^{(1)}$.

Generally, $A_{kl}^{(1)}$ is zeroed by setting $\theta$ so that $\sin\theta\cos\theta/(\cos^2\theta - \sin^2\theta) = A_{kl}/(A_{ll} - A_{kk})$. However, the exponentially decreasing form of $A$ (eq. (92)) allows us to make a simplifying approximation. Taking $k < l$, define

$$\begin{aligned} \theta_{kl} &= -(a_{kl}/a_{kk})\epsilon^{l-k} \\ &= -A_{kl}/A_{kk} + \mathscr{O}(\epsilon^{l-k+1}). \end{aligned} \tag{96}$$

Then, because $\epsilon \ll 1$, leading-order approximations to eqs. (95) for $\theta = \theta_{kl}$ can be obtained by substituting with eq. (92) and expanding the trigonometric functions with Taylor series about $\epsilon = 0$. The six equations for elements of the general transformation reduce to two, giving

$$A_{ij}^{(1)} = a_{ij}^{(1)}\epsilon^{i+j} + \mathscr{O}(\epsilon^{i+j+1}), \tag{97}$$

$$a_{ij}^{(1)} = \begin{cases} a_{ij} & \text{for } i, j \neq l \\ a_{il} - a_{ik}a_{kl}/a_{kk} & \text{for } j = l, \text{ all } i. \end{cases} \tag{98}$$

Substituting $i = k$ and $j = l$ in eq. (98) confirms that $a_{kl}^{(1)} = 0$. Note that the higher-order terms of the approximations are absorbed into the higher-order term for the matrix element $A_{ij}^{(1)}$. Also note that $A^{(1)}$ is of the same exponential form as $A$: their elements are on the same orders of $\epsilon$ (except for the element which is zeroed). Therefore the transformation can be iterated. A single step in the iteration is $A^{(n+1)} = J^{\dagger}(k,l,\theta_{kl})\,A^{(n)}J(k,l,\theta_{kl})$, where

$$A_{ij}^{(n+1)} = a_{ij}^{(n)}\epsilon^{i+j} + \mathscr{O}(\epsilon^{i+j+1}), \tag{99}$$

$$a_{ij}^{(n+1)} = \begin{cases} a_{ij}^{(n)} & \text{for } i, j \neq l \\ a_{il}^{(n)}a_{ik}^{(n)}a_{kl}^{(n)}/a_{kk}^{(n)} & \text{for } j = l, \text{ all } i. \end{cases} \tag{100}$$

Note that a transformation of this type causes changes *only in the $l$th column and $l$th row* of $A$.

The simplified form of the similarity transformation makes one sweep across the matrix elements sufficient for leading-order diagonalization. We prove this by induction:

*Proof.* Suppose $n$ transformations on $A$ diagonalize its $l \times l$ upper-left block, i.e.

$$a_{ij}^{(n)} = \begin{cases} 0 & \text{for } i \neq j,\ i < l,\ j < l, \\ \mathscr{O}(1) & \text{otherwise}. \end{cases} \tag{101}$$

We will show that the diagonal block can be extended by one row and one column by $l$ transformations

$$A^{(n+l)} = J_{l-1,l}^{\dagger} \cdots J_{0,l}^{\dagger}A^{(n)}J_{0,l} \cdots J_{l-1,l}, \tag{102}$$

$$= \left(\prod_{k=l-1}^{0} J_{kl}^{\dagger}\right)A^{(n)}\prod_{k=0}^{l-1} J_{kl}. \tag{103}$$

where $J_{kl} = J(k,l,\theta_{kl})$ with $\theta_{kl}$ given by

$$\theta_{kl} = -\frac{a_{kl}^{(n)}}{a_{kk}^{(n)}}\epsilon^{l-k}. \tag{104}$$

In this proof we take $l$ to be fixed, and $k$ to range between 0 and $l-1$.

To show that the diagonal block can be extended by eq. (103), we must verify first that the similarity transformations do not undo zeroes within the $l \times l$ upper-left diagonal block, and second that these transformations introduce zeroes at $A_{kl}^{(n+l)}$ for $k \in [0, l-1]$. The first verification is straightforward: Givens similarity transformations on matrices of the form of $A$ with $\theta_{kl} = \mathscr{O}(\epsilon^{l-k})$ and $k < l$ alter elements only on the $l$th row and column. Therefore the zeroes in upper-left $l \times l$ diagonal block of $A^{(n)}$ are preserved.

To prove the second, consider the innermost transformation ($k = 0$) in eq. (103), $A^{(n+1)} = J_{0,l}^{\dagger}A^{(n)}J_{0,l}$. Because the $l \times l$ upper-left block of $A^{(n)}$ is diagonal to leading order, $a_{i0}^{(n)} = 0$ for $i \neq k = 0$, $i < l$. Thus eq. (100), which gives the transformation of elements in the $l$th row and column, becomes

$$a_{il}^{(n+1)} = \begin{cases} 0 & \text{for } i = k = 0, \\ a_{il}^{(n)} & \text{for } 0 < i < l, \\ a_{il}^{(n)} - a_{i0}^{(n)}a_{0l}^{(n)}/a_{00}^{(n)} & \text{for } i \geq l. \end{cases} \tag{105}$$

The transformation zeroes $A_{0,l}^{(n+1)}$, but without changing any of the other elements in the $l$th row and column which are to be zeroed later in this sequence of transformations (that is, without changing elements $A_{il}^{(n)}$ for $i \neq 0$, $i < l$).

The following transformation in eq. (103) is for $k = 1$. Because the $k = 0$ transformation changed neither $a_{1,l}^{(n+1)}$ nor $a_{11}^{(n+1)}$ from its previous value, the rotation angle $\theta_{kl}$ defined by eq. (104) will zero $A_{1,l}^{(n+2)}$, even though it is defined in terms of $a_{1,l}^{(n)}$ and $a_{11}^{(n)}$. Applying eq. (100) again we get

$$a_{il}^{(n+2)} = \begin{cases} 0 & \text{for } i \leq k = 1, \\ a_{il}^{(n)} & \text{for } 1 < i < l, \\ a_{il}^{(n)} - a_{i0}^{(n)} a_{0l}^{(n)} / a_{00}^{(n)} & \\ \quad - a_{i1}^{(n)} a_{1l}^{(n)} / a_{11}^{(n)} & \text{for } i \geq l. \end{cases} \tag{106}$$

Now *two* elements in the $l$th column have been zeroed, and the others to be zeroed are unchanged.

This generalizes for each transformation $k < l$, and iterating eq. (100) $l$ times gives

$$a_{il}^{(n+l)} = \begin{cases} 0 & \text{for } i < l \\ a_{il}^{(n)} - \sum_{k=0}^{l-1} a_{il}^{(n)} a_{kl}^{(n)} / a_{kk}^{(n)} & \text{for } i \geq l. \end{cases} \tag{107}$$

Thus the sequence of transformations given by eqs. (103) and (104) extends the $l \times l$ diagonal block of $A^{(n)}$ to an $(l+1) \times (l+1)$ diagonal block.

The induction is started with $A^{(1)} = J_{01}^{\dagger} A J_{01}$, which gives a $2 \times 2$ diagonal block. Iterating eq. (107) $(m-2)$ times extends the diagonal block to cover the entire matrix, at which point the eigenvalues $\lambda_i$ are given by the diagonal elements, $\lambda_i = a_{ii}^{(\cdots)} \epsilon^{2i} + \epsilon^{2i+1}$. QED

To get all $m$ approximate eigenvalues in closed form, one must iterate eq. (107) through the entire matrix. However, since upper-left blocks of $A$ stay constant once they have been diagonalized, the first few eigenvalues can be obtained by a few iterations. The first three eigenvalues of $A = a_{ij} \epsilon^{i+j} + [\mathscr{O}(\epsilon^{i+j+1})]$ are

$$\lambda_0 = a_{00} + \mathscr{O}(\epsilon), \tag{108}$$

$$\lambda_1 = \left( a_{11} - \frac{a_{01}^2}{a_{00}} \right) \epsilon^2 + \mathscr{O}(\epsilon^3), \tag{109}$$

$$\lambda_2 = \left( a_{22} - \frac{a_{02}^2}{a_{00}} - \frac{(a_{00} a_{12} - a_{01} a_{02})^2}{a_{00}(a_{00} a_{11} - a_{01}^2)} \right) \epsilon^4 + \mathscr{O}(\epsilon^5). \tag{110}$$

Approximations to the eigenvectors of $A$ can be obtained in the following manner: Define $V$ as the matrix whose columns are the eigenvectors of $A$ in conventional order, i.e. $V^{\dagger} A V = \operatorname{diag}(\lambda_0, \lambda_1, \ldots)$. Then an approximation to $V$ is given by

$$\hat{V} = \prod_{l=1}^{m-1} \prod_{k=0}^{l-1} J_{kl}. \tag{111}$$

To find the order to which $\hat{V}$ is accurate, note that because $(\hat{V}^{\dagger} A \hat{V})_{ij} = \delta_{ij} \mathscr{O}(\epsilon^{2i}) + \mathscr{O}(\epsilon^{i+j+1})$, the further Givens transformations needed to diagonalize $A$ to all orders have $\theta_{ij} = \mathscr{O}(\epsilon^{j-i+1})$ (taking $i < j$). The largest of these is for $j - i = 1$, giving $\theta = \mathscr{O}(\epsilon^2)$. The largest higher-order term in these rotations is $\sin\theta = \mathscr{O}(\epsilon^2)$, so the true eigenvectors of $A$ are

$$V = \hat{V}\left(\mathrm{I}_m + [\mathscr{O}(\epsilon^2)]\right). \tag{112}$$

where $\mathrm{I}_m$ is the $m \times m$ identity matrix. Thus the accuracy of $\hat{V}$ is

$$V = \hat{V} + [\mathscr{O}(\epsilon^2)]. \tag{113}$$

Alternatively, since the largest rotation on the right-hand side of eq. (111) is $\mathscr{O}(\epsilon)$, we can approximate $V$ with the identity matrix, giving

$$V = \mathrm{I}_m + [\mathscr{O}(\epsilon)]. \tag{114}$$

### *C.2. Application to covariance matrix of Legendre coordinates*

In this appendix, we show how to apply the diagonalization procedure of appendix C.1 to the covariance matrix of Legendre coordinates. This gives an estimate for the value of $\tau_\mathrm{w}$ at which the small-window solution breaks down.

Appendix C gives formulae for the eigenvalues and eigenvectors of a matrix $[A_{ij}] = a_{ij}\epsilon^{i+j} + \mathscr{O}(\epsilon^{i+j+1})$, where $a_{ij} = \mathscr{O}(1)$ and $\epsilon \ll 1$. By eq. (41) the covariance matrix of even Legendre coordinates, $\Xi_w^\mathrm{e}$, has the form

$$(\Xi_w^\mathrm{e})_{ij} = (-1)^{i+j} \frac{c_{2i}c_{2j}\kappa_{i+j}}{2^{2i+2j}(2i)!(2j!)} \tau_\mathrm{w}^{2i+2j} + \mathscr{O}(\tau_\mathrm{w}^{2i+2j+2}). \qquad (115)$$

In order to apply the diagonalization procedure, we must factor the elements of $\Xi_w^\mathrm{e}$ in a form which fits the form of $A$.

The small parameter $\epsilon$ in $A$ should be proportional to the small parameter $\tau_\mathrm{w}^2$ in $\Xi_w^\mathrm{e}$, but the constant of proportionality is not immediately apparent. The constant can be determined by requiring that successive diagonal elements of $\Xi_w^\mathrm{e}$ decrease rapidly. By the definition of $\Xi_w^\mathrm{e}$, its diagonal elements are alternating diagonal elements of $\Xi_w$, so this requirement is met if

$$(\Xi_w)_{i,i} \gg (\Xi_w)_{i+1,i+1} \quad \text{for } i \in [0, m-1]. \qquad (116)$$

Substituting with eq. (37) gives

$$\frac{\tau_\mathrm{w}^2}{4(i+1)^2} \frac{c_{i+1}^2}{c_i^2} \frac{\kappa_{i+1}}{\kappa_i} \ll 1 \quad \text{for } i \in [0, m-1]. \qquad (117)$$

For simplicity's sake, we reduce eq. (117) to the instance $i = 0$[#14]. To eliminate the dependence on $p$, we replace $c_0^2(p)/c_1^2(p)$ with its limiting value, $\lim_{p\to\infty} c_0^2(p)/c_1^2(p) = 3$. This gives the requirement

$$\frac{\kappa_1}{12\kappa_0}\tau_\mathrm{w}^2 \ll 1. \qquad (118)$$

We define the *critical window width* $\tau_\mathrm{w}^*$ by

$$\tau_\mathrm{w}^* = 2\sqrt{\frac{3\kappa_0}{\kappa_1}}\,. \qquad (119)$$

Then eq. (118) is equivalent to requiring $\tau_\mathrm{w} \ll \tau_\mathrm{w}^*$. We set $\epsilon$ to

$$\epsilon = \left(\frac{\tau_\mathrm{w}}{\tau_\mathrm{w}^*}\right)^2 \qquad (120)$$

and $a_{ij}$ to

$$a_{ij} = (-1)^{i+j} \frac{c_{2i}c_{2j}}{(2i)!(2j)!} \left(\frac{3\kappa_0}{\kappa_1}\right)^{i+j} \kappa_{i+j}. \qquad (121)$$

We can apply the iterative diagonalization procedure of appendix C.1 to $\Xi_w^\mathrm{e}$ for $\epsilon \ll 1$ as long as variations in $a_{ij}$ are smaller than variations in $\epsilon^{i+j}$. It follows from the requirement $\epsilon \ll 1$ that the small-window solution is valid for $\tau_\mathrm{w} \ll \tau_\mathrm{w}^*$.

### *C.3. Recurrence relation for principal components*

Here we present a loose derivation for eq. (55), the recurrence relation for principal components. Consider the first two even Legendre coordinates,

[#14] It would be better to choose the value of $i$ which puts the most stringent requirement on $\tau_\mathrm{w}$. However, it is difficult to determine which $i$ this is. By the results of appendix B, the ratio $\kappa_{i+1}/\kappa_i$ is monotonically non-decreasing with $i$. On the other hand, $(i+1)^{-2}$ decreases. The ratio $c_{i+1}^2/c_i^2$ has negligible variation. Thus it is not clear whether $c_{i+1}^2\kappa_{i+1}/[(i+1)^2c_i^2\kappa_i]$ increases or decreases with $i$. However, in our experience, it is almost constant with $i$. We believe that this is related to the restrictions placed on $x(t)$ in section 3.

$w_0$ and $w_2$. By eq. (37), these coordinates are correlated, so a rotation is needed to decorrelate them. By eq. (28), $w_0 = \mathcal{O}(1)$ and $w_2 = \mathcal{O}(\tau_w^2)$, and by eq. (37), the covariance between them is order-$\tau_w^2$. Therefore, the rotation which decorrelates them has a negligible effect on $w_0$, but to $w_2$, it adds an order-$\tau_w^2$ component of $w_0$. Rotations between $w_0$ and $w_4$, $w_6$, $w_8$, etc., are similar. Thus when all these rotations have been carried out, $w_0$ is unchanged to leading order

$$w_0 \rightarrow w_0, \tag{122}$$

but each higher-order even Legendre coordinate $w_j$ has had an order $\tau_w^j$ component of $w_0$ added to it,

$$w_2 \rightarrow w_2 + \alpha_{02} w_0 = \mathcal{O}(\tau_w^2), \tag{123}$$

$$w_4 \rightarrow w_4 + \alpha_{04} w_0 = \mathcal{O}(\tau_w^4), \text{ etc.}, \tag{124}$$

where the $\alpha$'s are constants determined by the correlations between coordinates.

Next consider the rotations between $w_2 + \alpha_{02} w_0$ and higher-order coordinates such as $w_4 + \alpha_{04} w_4$. The coordinate $w_2 + \alpha_{02} w_0$ is now in the logical position formerly occupied by $w_0$: it is correlated only to higher-order coordinates, which are at least two orders of $\tau_w$ smaller. Therefore, this set of rotations brings about the following transformation:

$$w_2 + \alpha_{02} w_0 \rightarrow w_2 + \alpha_{02} w_0, \tag{125}$$

$$\begin{aligned} w_4 + \alpha_{04} w_0 \rightarrow {} & w_4 + \alpha_{04} w_0 \\ & + \alpha_{24}(w_2 + \alpha_{02} w_0) = \mathcal{O}(\tau_w^4), \end{aligned} \tag{126}$$

$$\begin{aligned} w_6 + \alpha_{06} w_0 \rightarrow {} & w_6 + \alpha_{06} w_0 \\ & + \alpha_{26}(w_2 + \alpha_{02} w_0) = \mathcal{O}(\tau_w^6), \\ & \text{etc.}, \end{aligned} \tag{127}$$

which preserves the linear independence of $w_0$ and $w_2 + \alpha_{02} w_0$.

When all sets of decorrelating transformations have been carried out, each Legendre coordinate has had a same-order component of each lower-order Legendre coordinate added to it.

$$w_0 \rightarrow w_0, \tag{128}$$

$$w_2 \rightarrow w_2 + \alpha'_{02} w_0, \tag{129}$$

$$w_4 \rightarrow w_4 + \alpha'_{24} w_2 + \alpha'_{04} w_0, \text{ etc.} \tag{130}$$

Since each set of transformations preserves the linearly independence set by previous sets, the right-hand sides are linearly independent, and therefore equal to the principal components (to leading order).

$$y_j = w_j + \sum_{i=0}^{j-1} \alpha'_{ij} w_i + \mathcal{O}(\tau_w^{j+2}) \tag{131}$$

The linear independence of principal components can be used to determine the coefficients of the linear combination, giving the recurrence relation

$$y_j(t) = w_j(t) - \sum_{i=0}^{j-1} y_i(t) \frac{\langle y_i w_j \rangle}{\langle y_i^2 \rangle} + \mathcal{O}(\tau_w^{j+2}). \tag{132}$$

Thus to leading order the principal components are a Gram–Schmidt orthogonalization of Legendre coordinates.

## Acknowledgements

We thank James Theiler for valuable discussions and particularly for pointing out the integration trick behind eq. (36). J. Gibson wishes to thank David Campbell and Timothy Sullivan. We are grateful for support from the National Institute of Mental Health under grant 1-R01-MH47184-01.

## References

[1] G.U. Yule, On a method of investigating periodicities in disturbed series with special reference to wolfer's sunspot numbers, Phil. Trans. R. Soc. London A, 226 (1927) 267–298.

[2] N.H. Packard, J.P. Crutchfield, J.D. Farmer, and R.S. Shaw, Geometry from a time series, Phys. Rev. Lett. 45 (1980) 712–716.

[3] F. Takens, Detecting strange attractors in fluid turbulence, in: Dynamical Systems and Turbulence, eds. D. Rand and L.S. Young (Springer, Berlin, 1981).

[4] M. Casdagli, S. Eubank, J.D. Farmer and J. Gibson, State space reconstruction in the presence of noise, Physica D 51 (1991) 52–98.

[5] H.D.I. Abarbanel, R. Brown, and J.B. Kadtke, Prediction and system identification in chaotic nonlinear systems: Time series with broadband spectra, Phys. Lett. A 138 (1989) 401–408.

[6] A.M. Fraser, Information and entropy in strange attractors, IEEE Transactions on Information Theory, IT-35 (1989).

[7] A.M. Fraser and H.L. Swinney, Independent coordinates for strange attractors from mutual information, Phys. Rev. A 33 (1986) 1134–1140.

[8] W. Liebert and H.G. Schuster, Proper choice of the time delay for the analysis of chaotic time series, Phys. Lett. A, 142 (1988) 107–111.

[9] D.S. Broomhead and G.P. King, Extracting qualitative dynamics from experimental data, Physica D 20 (1987) 217–236.

[10] A.I. Mees, P.E. Rapp and L.S. Jennings, Singular-value decomposition and embedding dimension, Phys. Rev. A 36 (1987) 340.

[11] T. Sauer, J. Yorke and M. Casdagli, Embedology, Santa Fe Institute technical report (1991).

[12] R. Vautard and M. Ghil, Singular spectrum analysis in nonlinear dynamics, with applications to paleoclimatic time series, Physica D 35 (1989) 395–424.

[13] D. Elliot and K.R. Rao, Fast Transforms: Algorithms, Analyses, Applications (Harcourt Brace Jovanovich, 1982).

[14] M. Ghil and R. Vautard, Interdecadal oscillations and the warming trend in global temperature time series, Nature 350 (1991) 324–327.

[15] A.M. Fraser, Reconstructing attractors from scalar time series: A comparison of singular system and redundancy criteria. Physica D 34 (1989) 391–404.

[16] F.B. Hildebrand, Advanced Calculus for Applications (Prentice-Hall, New York, 1976).

[17] G.H. Golub and C.F. Van Loan, Matrix Computations (Johns Hopkins, Baltimore, 1983).