Nonlinear Principal Component Analysis

8

William W. Hsieh

8.1 Introduction

In environmental sciences, one often encounters large datasets with many variables. For instance, one may have a dataset of the monthly sea surface temperature (SST) anomalies ("anomalies" are the departures from the mean) collected at $l = 1{,}000$ grid locations over several decades, i.e. the data are of the form $\mathbf{x} = [x_1, \ldots, x_l]$, where each variable x_i $(i = 1, \ldots, l)$ has n samples. The samples may be collected at times t_k $(k = 1, \ldots, n)$, so each x_i is a time series containing n observations. Since the SST of neighboring grids are correlated, and a dataset with 1,000 variables is quite unwieldy, one looks for ways to condense the large dataset to only a few principal variables. The most common approach is via principal component analysis (PCA), also known as empirical orthogonal function (EOF) analysis (Jolliffe 2002).

In the example with 1,000 variables, imagine we have plotted out all the n samples in the 1,000-dimensional data space, with each sample being a data point in this space. We then try to fit the best straight line through the data points. Mathematically, PCA looks for u, a linear combination of the x_i, and an associated vector $\mathbf{e}$ (which gives the direction of the desired straight line), with

$$u(t) = \mathbf{e} \cdot \mathbf{x}(t)\,, \tag{8.1}$$

so that

$$\langle \|\mathbf{x}(t) - \mathbf{e}\, u(t)\|^2 \rangle \text{ is minimized,} \tag{8.2}$$

where $\langle \cdots \rangle$ denotes a sample mean or time mean. Here u, called the first principal component (PC) (or score), is often a time series, while $\mathbf{e}$, called the first eigenvector (also called an empirical orthogonal function, EOF, or loading), is the first eigenvector of the data covariance matrix $\mathbf{C}$, with elements C_{ij} given by

$$C_{ij} = \frac{1}{n-1} \sum_{k=1}^{n} [x_i(t_k) - \langle x_i \rangle]\,[x_j(t_k) - \langle x_j \rangle]\,. \tag{8.3}$$

Together u and $\mathbf{e}$ make up the first PCA mode. In the above example, $\mathbf{e}$ simply describes a fixed spatial SST anomaly pattern. How strongly this pattern is manifested at a given time is controlled by the time series u.

From the residual, $\mathbf{x} - \mathbf{e}u$, the second PCA mode can similarly be extracted, and so on for the higher modes. In practice, the common algorithms for PCA extract all modes simultaneously (Jolliffe 2002; Preisendorfer 1988). By retaining only the leading modes, PCA has been commonly used to reduce the dimensionality of the dataset, and to extract the main patterns from the dataset.

Principal component analysis (PCA) can be performed using neural network (NN) methods (Oja 1982; Sanger 1989). However, far more interesting is the nonlinear generalization of PCA, where several distinct approaches have been developed (Cherkassky and Mulier 1998). As PCA finds a straight line which passes through the 'middle' of the data cluster, the obvious next step is to generalize the straight line to a curve. The multi-layer perceptron (MLP) model (see Section 1.8) has been adapted to perform nonlinear PCA (Kramer 1991; Hsieh 2004). Alternative approaches are the principal curves method (Hastie and Stuetzle 1989; Hastie et al. 2001), the kernel PCA method (Schölkopf et al. 1998) and the self-organizing

William W. Hsieh (✉)
Department of Earth and Ocean Sciences, 6339 Stores Rd., University of British Columbia, Vancouver, B.C. V6T 1Z4, Canada
Phone: 604-822-2821; fax: 604-822-6088,
email: whsieh@eos.ubc.ca

S. E. Haupt et al. (eds.), *Artificial Intelligence Methods in the Environmental Sciences*

map (SOM) technique (Kohonen 1982; Cherkassky and Mulier 1998).

In this chapter, we examine the use of MLP NN models for nonlinear PCA (NLPCA) in Section 8.2, the overfitting problem associated with NLPCA in Section 8.3, and the extension of NLPCA to closed curve solutions in Section 8.4. MATLAB codes for NLPCA are downloadable from http://www.ocgy.ubc.ca/projects/clim.pred/download.html.The discrete approach by self-organizing maps is presented in Sections 8.5, and the generalization of NLPCA to complex variables in Section 8.6.

8.2 Auto-Associative Neural Networks for NLPCA

The fundamental difference between NLPCA and PCA is that PCA only allows a linear mapping ($u = \mathbf{e} \cdot \mathbf{x}$) between $\mathbf{x}$ and the PC u, while NLPCA allows a nonlinear mapping. To perform NLPCA, Kramer (1991) proposed using the MLP NN in Fig. 8.1a where there are three hidden layers of neurons (i.e. variables) between the input and output layers. The NLPCA is basically a standard MLP NN (see Section 1.8) with four-layers of activation functions (i.e. transfer functions) mapping from the inputs to the outputs. One can view the NLPCA network as composed of two-standard two-layer MLP NNs placed one after the other. The first two-layer network maps from the inputs $\mathbf{x}$ through a hidden layer to the bottleneck layer with only one neuron u, i.e. a nonlinear mapping $u = f(\mathbf{x})$. The next two-layer MLP NN inversely maps from the nonlinear PC (NLPC) u back to the original higher dimensional $\mathbf{x}$-space, with the objective that the outputs $\mathbf{x}' = \mathbf{g}(u)$ be as close as possible to the inputs $\mathbf{x}$, where $\mathbf{g}(u)$ nonlinearly generates a curve in the $\mathbf{x}$-space, hence a 1-dimensional approximation of the original data. Because the target data for the output neurons $\mathbf{x}'$ are simply the input data $\mathbf{x}$, such networks are called auto-associative NNs. To minimize the MSE (mean square error) of this approximation, the objective function (also called cost function or loss function) $J = \langle \|\mathbf{x} - \mathbf{x}'\|^2 \rangle$ is minimized to solve for the parameters of the NN. Squeezing the input information through a bottleneck layer with only one neuron accomplishes the dimensional reduction.

In Fig. 8.1a, the activation function f_1 maps from $\mathbf{x}$, the input column vector of length l, to the first

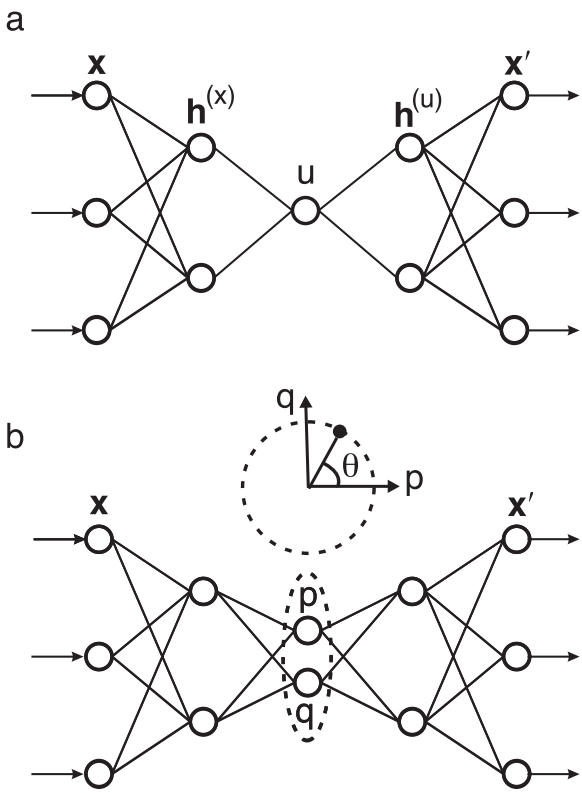

Fig. 8.1 (a) A schematic diagram of the autoassociative feed-forward multi-layer perceptron NN model for performing NLPCA. Between the input layer $\mathbf{x}$ on the left (the 0th layer) and the output layer $\mathbf{x}'$ on the far right (the 4th layer), there are three layers of 'hidden' neurons (the 1st, 2nd and 3rd layers). Layer 2 is the 'bottleneck' with a single neuron u giving the nonlinear principal component (NLPC). Layers 1 and 3, each with m hidden neurons, are called the encoding and decoding layers, respectively. (b) The NN model used for extracting a *closed* curve NLPCA solution. At the bottleneck, there are now two neurons p and q constrained to lie on a unit circle in the p-q plane, giving effectively one free angular variable θ, the NLPC. This network is suited for extracting a closed curve solution (Reprinted from Hsieh 2001. With permission from Blackwell)

hidden layer (the encoding layer), represented by $\mathbf{h}^{(x)}$, a column vector of length m, with elements

$$h_k^{(x)} = f_1((\mathbf{W}^{(x)}\mathbf{x} + \mathbf{b}^{(x)})_k), \tag{8.4}$$

where $\mathbf{W}^{(x)}$ is an $m \times l$ weight matrix, $\mathbf{b}^{(x)}$, a column vector of length m containing the offset (i.e. bias) parameters, and $k = 1, \ldots, m$. Similarly, a second activation function f_2 maps from the encoding layer to the bottleneck layer containing a single neuron, which represents the nonlinear principal component u,

$$u = f_2(\mathbf{w}^{(x)} \cdot \mathbf{h}^{(x)} + \overline{b}^{(x)}). \tag{8.5}$$

The activation function f_1 is generally nonlinear (usually the hyperbolic tangent or the sigmoidal function, though the exact form is not critical), while f_2 is usually taken to be the identity function.

Next, an activation function f_3 maps from u to the third hidden layer (the decoding layer) $\mathbf{h}^{(u)}$,

$$h_k^{(u)} = f_3((\mathbf{w}^{(u)} u + \mathbf{b}^{(u)})_k)\,, \tag{8.6}$$

$(k = 1, \ldots, m)$; followed by f_4 mapping from $\mathbf{h}^{(u)}$ to $\mathbf{x}'$, the output column vector of length l, with

$$x_i' = f_4((\mathbf{W}^{(u)}\mathbf{h}^{(u)} + \overline{\mathbf{b}}^{(u)})_i). \tag{8.7}$$

The objective function $J = \langle \|\mathbf{x} - \mathbf{x}'\|^2 \rangle$ is minimized by finding the optimal values of $\mathbf{W}^{(x)}$, $\mathbf{b}^{(x)}$, $\mathbf{w}^{(x)}$, $\overline{b}^{(x)}$, $\mathbf{w}^{(u)}$, $\mathbf{b}^{(u)}$, $\mathbf{W}^{(u)}$ and $\overline{\mathbf{b}}^{(u)}$. The MSE between the NN output $\mathbf{x}'$ and the original data $\mathbf{x}$ is thus minimized. The NLPCA was implemented using the hyperbolic tangent function for f_1 and f_3, and the identity function for f_2 and f_4, so that

$$u = \mathbf{w}^{(x)} \cdot \mathbf{h}^{(x)} + \overline{b}^{(x)}, \tag{8.8}$$

$$x_i' = (\mathbf{W}^{(u)}\mathbf{h}^{(u)} + \overline{\mathbf{b}}^{(u)})_i. \tag{8.9}$$

Furthermore, we adopt the normalization conditions that $\langle u \rangle = 0$ and $\langle u^2 \rangle = 1$. These conditions are approximately satisfied by modifying the objective function to

$$J = \langle \|\mathbf{x} - \mathbf{x}'\|^2 \rangle + \langle u \rangle^2 + (\langle u^2 \rangle - 1)^2\,. \tag{8.10}$$

The total number of (weight and offset) parameters used by the NLPCA is $2lm + 4m + l + 1$, though the number of effectively free parameters is two less due to the constraints on $\langle u \rangle$ and $\langle u^2 \rangle$.

The choice of m, the number of hidden neurons in both the encoding and decoding layers, follows a general principle of parsimony. A larger m increases the nonlinear modeling capability of the network, but could also lead to overfitted solutions (i.e. wiggly solutions which fit to the noise in the data). If f_4 is the identity function, and $m = 1$, then (8.9) implies that all x_i' are linearly related to a single hidden neuron, hence there can only be a linear relation between the x_i' variables. Thus, for nonlinear solutions, we need to look at $m \geq 2$. Actually, one can use different numbers of neurons in the encoding layer and in the decoding layer; however, keeping them both at m neurons gives roughly the same number of parameters in the forward mapping from $\mathbf{x}$ to u and in the inverse mapping from u to $\mathbf{x}'$. It is also possible to have more than one neuron at the bottleneck layer. For instance, with two bottleneck neurons, the mode extracted will span a 2-D surface instead of a 1-D curve.

Because of local minima in the objective function, there is no guarantee that the optimization algorithm reaches the global minimum. Hence a number of runs with random initial weights and offset parameters was made. Also, a portion (e.g. 15%) of the data was randomly selected as validation data and withheld from the training of the NNs. Runs where the MSE was larger for the validation dataset than for the training dataset were rejected to avoid overfitted solutions. Then the run with the smallest MSE was selected as the solution.

In general, the presence of local minima in the objective function is a major problem for NLPCA. Optimizations started from different initial parameters often converge to different minima, rendering the solution unstable or nonunique. Adding weight penalty terms to the objective function (also called "regularization") is an answer.

The purpose of the weight penalty terms is to limit the nonlinear power of the NLPCA, which came from the nonlinear activation functions in the network. The activation function tanh has the property that given x in the interval $[-L, L]$, one can find a small enough weight w, so that $\tanh(wx) \approx wx$, i.e. the activation function is almost linear. Similarly, one can choose a large enough w, so that tanh approaches a step function, thus yielding Z-shaped solutions. If we can penalize the use of excessive weights, we can limit the degree of nonlinearity in the NLPCA solution. This is achieved with a modified objective function

$$J = \langle \|\mathbf{x} - \mathbf{x}'\|^2 \rangle + \langle u \rangle^2 + (\langle u^2 \rangle - 1)^2 + P \sum_{ki} (W_{ki}^{(x)})^2, \tag{8.11}$$

where P is the weight penalty parameter. A large P increases the concavity of the objective function, and forces the weights $\mathbf{W}^{(x)}$ to be small in magnitude, thereby yielding smoother and less nonlinear solutions than when P is small or zero. Hence, increasing P also reduces the number of effectively free parameters of the model. We have not penalized other weights in the network. In principle, $\mathbf{w}^{(u)}$ also controls the nonlinearity in the inverse mapping from u to $\mathbf{x}'$. However if the nonlinearity in the forward mapping from $\mathbf{x}$ to u is already being limited by penalizing $\mathbf{W}^{(x)}$, then there

is no need to further limit the weights in the inverse mapping.

In summary, one needs to choose m large enough so that the NN model has enough flexibility to approximate the true solution well. The weight penalty P can be regarded as a smoothing parameter, i.e. if P is large enough, zigzags and wiggles in the curve solution can be eliminated. How to choose P and m objectively has only recently been addressed, and is discussed in Section 8.3.

In effect, the linear relation ($u = \mathbf{e} \cdot \mathbf{x}$) in PCA is now generalized to $u = f(\mathbf{x})$, where f can be any nonlinear continuous function representable by an MLP NN mapping from the input layer to the bottleneck layer; and $\langle \|\mathbf{x} - \mathbf{g}(u)\|^2 \rangle$ is minimized. Limitations in the mapping properties of the NLPCA are discussed by Newbigging et al. (2003). The residual, $\mathbf{x} - \mathbf{g}(u)$, can be input into the same network to extract the second NLPCA mode, and so on for the higher modes.

That the classical PCA is indeed a linear version of this NLPCA can be readily seen by replacing all the activation functions with the identity function, thereby removing the nonlinear modeling capability of the NLPCA. Then the forward map to u involves only a linear combination of the original variables as in the PCA.

In the classical linear approach, there is a well-known dichotomy between PCA and rotated PCA (RPCA) (Richman 1986). In PCA, the linear mode which accounts for the most variance of the dataset is sought. However, as illustrated in Preisendorfer (1988, Fig. 7.3), the resulting eigenvectors may not align close to local data clusters, so the eigenvectors may not represent actual physical states well. One application of RPCA methods is to rotate the PCA eigenvectors, so they point closer to the local clusters of data points (Preisendorfer 1988). Thus the rotated eigenvectors may bear greater resemblance to actual physical states (though they account for less variance) than the unrotated eigenvectors, hence RPCA is also widely used (Richman 1986; von Storch and Zwiers 1999). As there are many possible criteria for rotation, there are many RPCA schemes, among which the varimax (Kaiser 1958) scheme is perhaps the most popular. We will compare NLPCA with PCA and RPCA in the following subsection.

8.2.1 Applications of NLPCA

The NLPCA has been applied to the Lorenz (1963) three-component chaotic system (Monahan 2000; Hsieh 2001). For the tropical Pacific climate variability, the NLPCA has been used to study the SST field (Monahan 2001; Hsieh 2001) and the sea level pressure (SLP) field (Monahan 2001). The Northern Hemisphere atmospheric variability (Monahan et al. 2000, 2001) and the subsurface thermal structure of the Pacific Ocean (Tang and Hsieh 2003) have also been investigated by the NLPCA. In remote sensing, Del Frate and Schiavon (1999) applied NLPCA to the inversion of radiometric data to retrieve atmospheric profiles of temperature and water vapour.

The tropical Pacific climate system contains the famous interannual variability known as the El Niño-Southern Oscillation (ENSO), a coupled atmosphere-ocean interaction involving the oceanic phenomenon El Niño and the associated atmospheric phenomenon, the Southern Oscillation. The coupled interaction results in anomalously warm SST in the eastern equatorial Pacific during El Niño episodes, and cool SST in the central equatorial Pacific during La Niña episodes (Philander 1990; Diaz and Markgraf 2000). ENSO is an irregular oscillation, but spectral analysis does reveal a broad spectral peak at the 4–5 year period. Hsieh (2001) used the tropical Pacific SST data (1950–1999) to make a three-way comparison between NLPCA, RPCA and PCA. The tropical Pacific SST anomaly (SSTA) data (i.e. the SST data with the climatological seasonal cycle removed) were pre-filtered by PCA, with only the three leading modes retained. PCA modes 1, 2 and 3 accounted for 51.4%, 10.1% and 7.2%, respectively, of the variance in the SSTA data. Due to the large number of spatially gridded variables, NLPCA could not be applied directly to the SSTA time series, as this would lead to a huge NN with the number of model parameters vastly exceeding the number of samples. Instead, the first three PCs (PC1, PC2 and PC3) were used as the input $\mathbf{x}$ for the NLPCA network.

The data are shown as dots in a scatter plot in the PC1-PC2 plane (Fig. 8.2), where the cool La Niña states lie in the upper left corner, and the warm El Niño states in the upper right corner. The NLPCA solution is a U-shaped curve linking the La Niña states at one end (low u) to the El Niño states at the other end (high u),

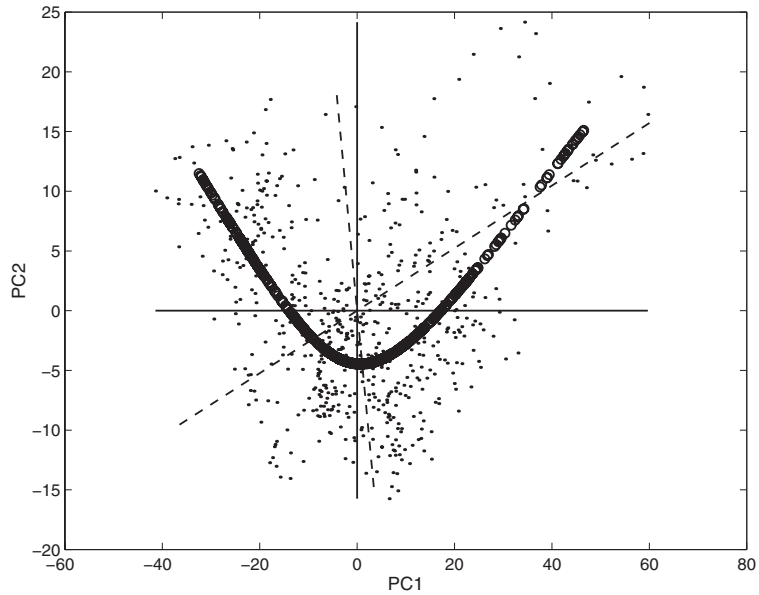

Fig. 8.2 Scatter plot of the SST anomaly (SSTA) data (shown as dots) in the PC1-PC2 plane, with the El Niño states lying in the upper right corner, and the La Niña states in the upper left corner. The PC2 axis is stretched relative to the PC1 axis for better visualization. The first mode NLPCA approximation to the data is shown by the (overlapping) small circles, which traced out a U-shaped curve. The first PCA eigenvector is oriented along the horizontal line, and the second PCA, by the vertical line. The varimax method rotates the two PCA eigenvectors in a counterclockwise direction, as the rotated PCA (RPCA) eigenvectors are oriented along the dashed lines. (As the varimax method generates an orthogonal rotation, the angle between the two RPCA eigenvectors is 90° in the 3-dimensional PC1-PC2-PC3 space) (Reprinted from Hsieh 2001. With permission from Blackwell)

similar to that found originally by Monahan (2001). In contrast, the first PCA eigenvector lies along the horizontal line, and the second PCA, along the vertical line (Fig. 8.2). It is easy to see that the first PCA eigenvector describes a somewhat unphysical oscillation, as there are no dots (data) close to either ends of the horizontal line. For the second PCA eigenvector, there are dots close to the bottom of the vertical line, but no dots near the top end of the line, i.e. one phase of the mode 2 oscillation is realistic, but the opposite phase is not. Thus if the underlying data has a nonlinear structure but we are restricted to finding linear solutions using PCA, the energy of the nonlinear oscillation is scattered into multiple PCA modes, many of which represent unphysical linear oscillations.

For comparison, a varimax rotation (Kaiser 1958; Preisendorfer 1988), was applied to the first three PCA eigenvectors. The varimax criterion can be applied to either the loadings or the PCs depending on one's objectives (Richman 1986; Preisendorfer 1988); here it is applied to the PCs. The resulting first RPCA eigenvector, shown as a dashed line in Fig. 8.2, spears through the cluster of El Niño states in the upper right corner, thereby yielding a more accurate description of the El Niño anomalies (Fig. 8.3c) than the first PCA mode (Fig. 8.3a), which did not fully represent the intense warming of Peruvian waters. The second RPCA eigenvector, also shown as a dashed line in Fig. 8.2, did not improve much on the second PCA mode, with the PCA spatial pattern shown in Fig. 8.3b, and the RPCA pattern in Fig. 8.3d). In terms of variance explained, the first NLPCA mode explained 56.6% of the variance, versus 51.4% by the first PCA mode, and 47.2% by the first RPCA mode.

With the NLPCA, for a given value of the NLPC u, one can map from u to the three PCs. This is done by assigning the value u to the bottleneck neuron and mapping forward using the second half of the network in Fig. 8.1a. Each of the three PCs can be multiplied by its associated PCA (spatial) eigenvector, and the three added together to yield the spatial pattern

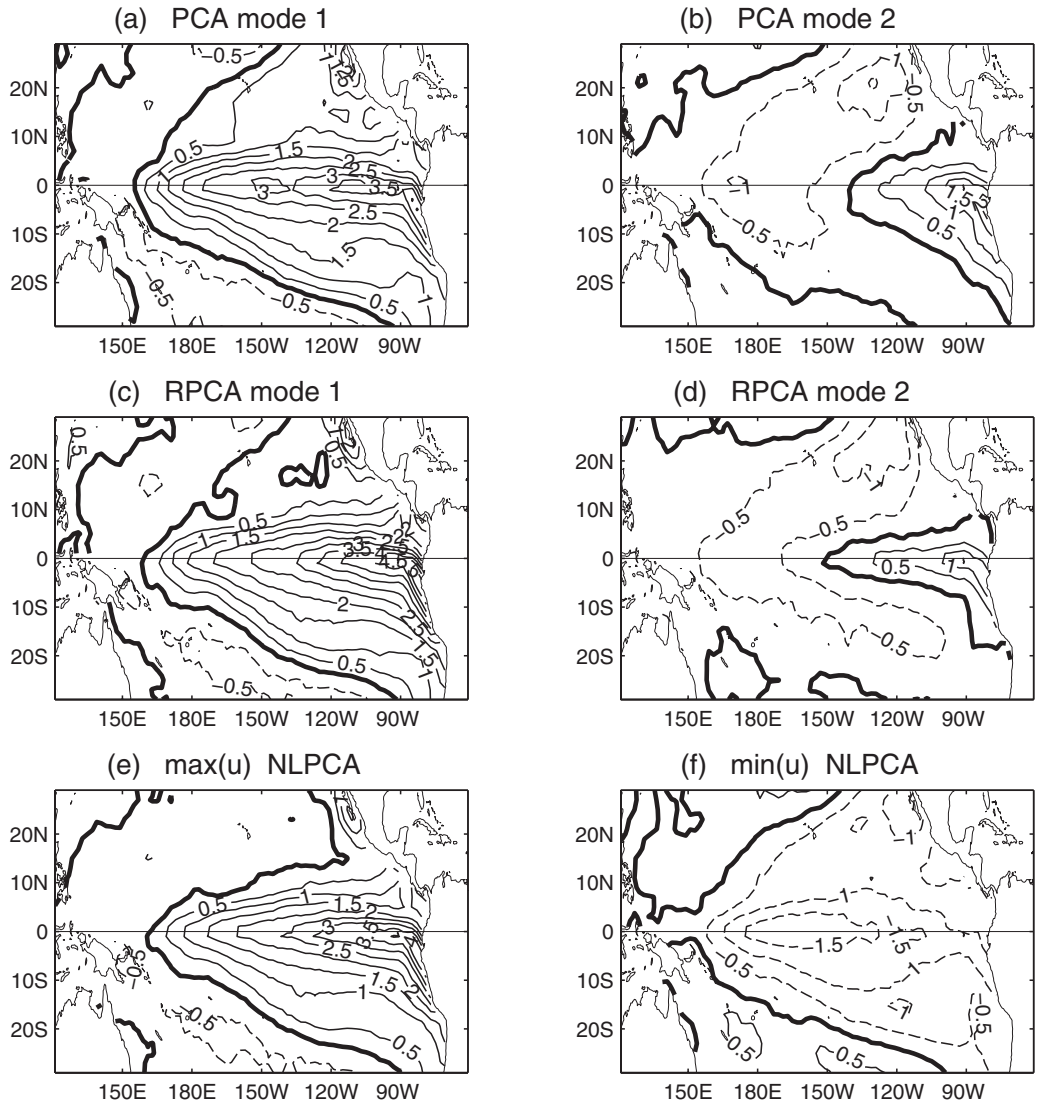

Fig. 8.3 The SSTA patterns (in °C) of the PCA, RPCA and the NLPCA. The first and second PCA spatial modes are shown in (a) and (b) respectively, (both with their corresponding PCs at maximum value). The first and second varimax RPCA spatial modes are shown in (c) and (d) respectively, (both with their corresponding RPCs at maximum value). The anomaly pattern as the NLPC u of the first NLPCA mode varies from (e) maximum (strong El Niño) to (f) its minimum (strong La Niña). With a contour interval of 0.5°C , the positive contours are shown as solid curves, negative contours, dashed curves, and the zero contour, a thick curve (Reprinted from Hsieh 2004. With permission from American Geophysical Union)

for that particular value of u. Unlike PCA which gives the same spatial anomaly pattern except for changes in the amplitude as the PC varies, the NLPCA spatial pattern generally varies continuously as the NLPC changes. Figure 8.3e, f show respectively the spatial anomaly patterns when u has its maximum value (corresponding to the strongest El Niño) and when u has its minimum value (strongest La Niña). Clearly the asymmetry between El Niño and La Niña, i.e. the cool anomalies during La Niña episodes (Fig. 8.3f) are observed to center much further west than the warm anomalies during El Niño (Fig. 8.3e) (Hoerling et al. 1997), is well captured by the first NLPCA mode – in contrast, the PCA mode 1 gives a La Niña which is simply the mirror image of the El Niño (Fig. 8.3a). The asymmetry explains why El

Niño has been known by Peruvian fishermen for many centuries due to its strong SSTA off the coast of Peru and its devastation of the Peruvian fishery, whereas the La Niña, with its weak manifestation in the Peruvian waters, was not appreciated until the last two decades of the 20th century.

In summary, PCA is used for two main purposes: (i) to reduce the dimensionality of the dataset, and (ii) to extract features or recognize patterns from the dataset. It is primarily purpose (ii) where PCA can be improved upon. Both RPCA and NLPCA take the PCs from PCA as input. However, instead of multiplying the PCs by a fixed orthonormal rotational matrix, as performed in the varimax RPCA approach, NLPCA performs a nonlinear mapping of the PCs. RPCA sacrifices on the amount of variance explained, but by rotating the PCA eigenvectors, RPCA eigenvectors tend to point more towards local data clusters and are therefore more representative of physical states than the PCA eigenvectors.

With a linear approach, it is generally impossible to have a solution simultaneously (a) explaining maximum global variance of the dataset and (b) approaching local data clusters, hence the dichotomy between PCA and RPCA, with PCA aiming for (a) and RPCA for (b). Hsieh (2001) pointed out that with the more flexible NLPCA method, both objectives (a) and (b) may be attained together, thus the nonlinearity in NLPCA unifies the PCA and RPCA approaches. It is easy to see why the dichotomy between PCA and RPCA in the linear approach automatically vanishes in the nonlinear approach. By increasing m, the number of hidden neurons in the encoding layer (and the decoding layer), the solution is capable of going through all local data clusters while maximizing the global variance explained. (In fact, for large enough m, NLPCA can pass through all data points, though this will in general give an undesirable, overfitted solution.)

The tropical Pacific SST example illustrates that with a complicated oscillation like the El Niño-La Niña phenomenon, using a linear method such as PCA results in the nonlinear mode being scattered into several linear modes (in fact, all three leading PCA modes are related to this phenomenon) – hence the importance of the NLPCA as a unifier of the separate linear modes. In the study of climate variability, the wide use of PCA methods has created the somewhat misleading view that our climate is dominated by a number of spatially fixed oscillatory patterns, which may in fact be due to the limitation of the linear method. Applying NLPCA to the tropical Pacific SSTA, we found no spatially fixed oscillatory patterns, but an oscillation evolving in space as well as in time.

8.3 Overfitting in NLPCA

When using nonlinear machine learning methods, the presence of noise in the data can lead to overfitting. When plentiful data are available (i.e. far more samples than model parameters), overfitting is not a problem when performing nonlinear regression on noisy data. Unfortunately, even with plentiful data, overfitting is a problem when applying NLPCA to noisy data (Hsieh 2001; Christiansen 2005; Hsieh 2007). As illustrated in Fig. 8.4, overfitting in NLPCA can arise from the geometry of the problem, rather than from the scarcity of data. Here for a Gaussian-distributed data cloud, a nonlinear model with enough flexibility will find the zigzag solution of Fig. 8.4b as having a smaller MSE than the linear solution in Fig. 8.4a. Since the distance between the point A and a, its projection on the NLPCA curve, is smaller in Fig. 8.4b than the corresponding distance in Fig. 8.4a, it is easy to see that the more zigzags there are in the curve, the smaller is the MSE. However, the two neighboring points A and B, on opposite sides of an ambiguity line, are projected far apart on the NLPCA curve in Fig. 8.4b. Thus simply searching for the solution which gives the smallest MSE does not guarantee that NLPCA will find a good solution in a highly noisy dataset.

Hsieh (2001) added weight penalty to the Kramer (1991) NLPCA model to smooth out excessively wiggly solutions, but did not provide an objective way to select the optimal weight penalty parameter P. With NLPCA, if the overfitting arise from the data geometry (as in Fig. 8.4b) and not from data scarcity, using independent data to validate the MSE from the various models is not a viable method for choosing the appropriate P. Instead, Hsieh (2007) proposed an "inconsistency" index for detecting the projection of neighboring points to distant parts of the NLPCA curve, and use the index to choose the appropriate P.

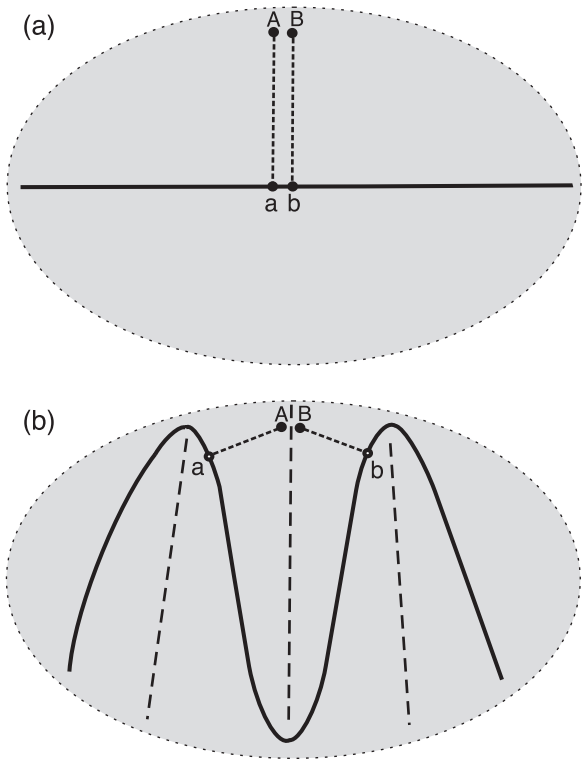

Fig. 8.4 Schematic diagram illustrating overfitting on noisy data. (a) PCA solution for a Gaussian data cloud, with two neighboring points A and B shown projecting to the points a and b on the PCA straight line solution. (b) A zigzag NLPCA solution found by a flexible enough nonlinear model. Dashed lines illustrate ambiguity lines where neighboring points (e.g. A and B) on opposite sides of these lines are projected to a and b, far apart on the NLPCA curve (Reprinted from Hsieh 2007. With permission from Elsevier)

For each data point $\mathbf{x}$, find its nearest neighbor $\tilde{\mathbf{x}}$. The NLPC for $\mathbf{x}$ and $\tilde{\mathbf{x}}$ are u and $\tilde{u}$, respectively. With $C(u, \tilde{u})$ denoting the (Pearson) correlation between all the pairs $(u, \tilde{u})$, the inconsistency index I was defined in Hsieh (2007) as

$$I = 1 - C(u, \tilde{u}). \tag{8.12}$$

If for some nearest neighbor pairs, u and $\tilde{u}$ are assigned very different values, $C(u, \tilde{u})$ would have a lower value, leading to a larger I, indicating greater inconsistency in the NLPC mapping. With u and $\tilde{u}$ standardized to having zero mean and unit standard deviation, (8.12) is equivalent to

$$I = \frac{1}{2} \langle (u - \tilde{u})^2 \rangle . \tag{8.13}$$

In statistics, various criteria, often in the context of linear models, have been developed to select the right amount of model complexity so neither overfitting nor underfitting occurs. These criteria are often called "information criteria" (IC) (von Storch and Zwiers 1999). An IC is typically of the form

$$\text{IC} = \text{MSE} + \text{complexity term}, \tag{8.14}$$

where MSE is evaluated over the training data and the complexity term is larger when a model has more free parameters. The IC is evaluated over a number of models with different free parameters, and the model with the minimum IC is selected as the best. As the presence of the complexity term in the IC penalizes models which use excessive number of free parameters to attain low MSE, choosing the model with the minimum IC would rule out complex models with overfitted solutions.

In Hsieh (2007), the data were randomly divided into a training data set and a validation set (containing 85% and 15% of the original data, respectively), and for every given value of P and m, the model was trained a number of times from random initial weights, and model runs where the MSE evaluated over the validation data was larger than the MSE over the training data were discarded. To choose among the model runs which had passed the validation test, a new holistic IC to deal with the type of overfitting arising from the broad data geometry (Fig. 8.4b) was introduced as

$$H = \text{MSE} + \text{inconsistency term} \tag{8.15}$$

$$= \text{MSE} - C(u, \tilde{u}) \times \text{MSE} \; = \; \text{MSE} \times I, \tag{8.16}$$

where MSE and C were evaluated over all (training and validation) data, inconsistency was penalized, and the model run with the smallest H value was selected as the best. As the inconsistency term only prevents overfitting arising from the broad data geometry, validation data are still needed to prevent "local" overfitting from excessive number of model parameters, since H, unlike (8.14), does not contain a complexity term.

Consider the test problem in Hsieh (2007): For a random number t uniformly distributed in the interval $(-1, 1)$, the signal $\mathbf{x}^{(s)}$ was generated by using a quadratic relation

$$x_1^{(s)} = t, \qquad x_2^{(s)} = \frac{1}{2} t^2. \tag{8.17}$$

Isotropic Gaussian noise was added to the signal $\mathbf{x}^{(s)}$ to give the noisy data $\mathbf{x}$ with 500 samples. NLPCA was performed on the data using the network in Fig. 8.1a with $m = 4$ and with the weight penalty P at various values (10, 1, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 0). For each value of P, the model training was done 30 times starting from random initial weights, and model runs where the MSE evaluated over the validation data was larger than the MSE over the training data were deemed ineligible. In the traditional approach, among the eligible runs over the range of P values, the one with the lowest MSE over all (training and validation) data was selected as the best. Figure 8.5a shows this solution where the zigzag curve retrieved by NLPCA is very different from the theoretical parabolic signal (8.17), demonstrating the pitfall of selecting the lowest MSE run.

In contrast, in Fig. 8.5b, among the eligible runs over the range of P values, the one with the lowest information criterion H was selected. This solution, which has a much larger weight penalty ($P = 0.1$) than that in Fig. 8.5a ($P = 10^{-4}$), shows less wiggly behaviour and better agreement with the theoretical parabolic signal.

Even less wiggly solutions can be obtained by changing the error norm used in the objective function from the mean square error to the mean absolute error (MAE), i.e. replacing $\langle \|\mathbf{x} - \mathbf{x}'\|^2 \rangle$ by $\langle \sum_j |x_j - x'_j| \rangle$ in equation (8.11). The MAE norm is known to be robust to outliers in the data (Bishop 1995, p. 210). Figure 8.5c is the solution selected based on minimum H with the MAE norm used. While wiggles are eliminated, the solution underestimates the curvature in the parabolic signal. The rest of this paper uses the MSE norm.

In summary, with noisy data, not having plentiful samples could cause a flexible nonlinear model to overfit. In the limit of infinite samples, overfitting cannot occur in nonlinear regression, but can still occur in NLPCA due to the geometric shape of the data distribution. A new inconsistency index I for detecting the projection of neighboring points to distant parts of the NLPCA curve has been introduced, and incorporated into a holistic IC H to select the model with the appropriate weight penalty parameter and the appropriate number of hidden neurons. An alternative approach for model selection was proposed by Webb (1999), who applied a constraint on the Jacobian in the objective function.

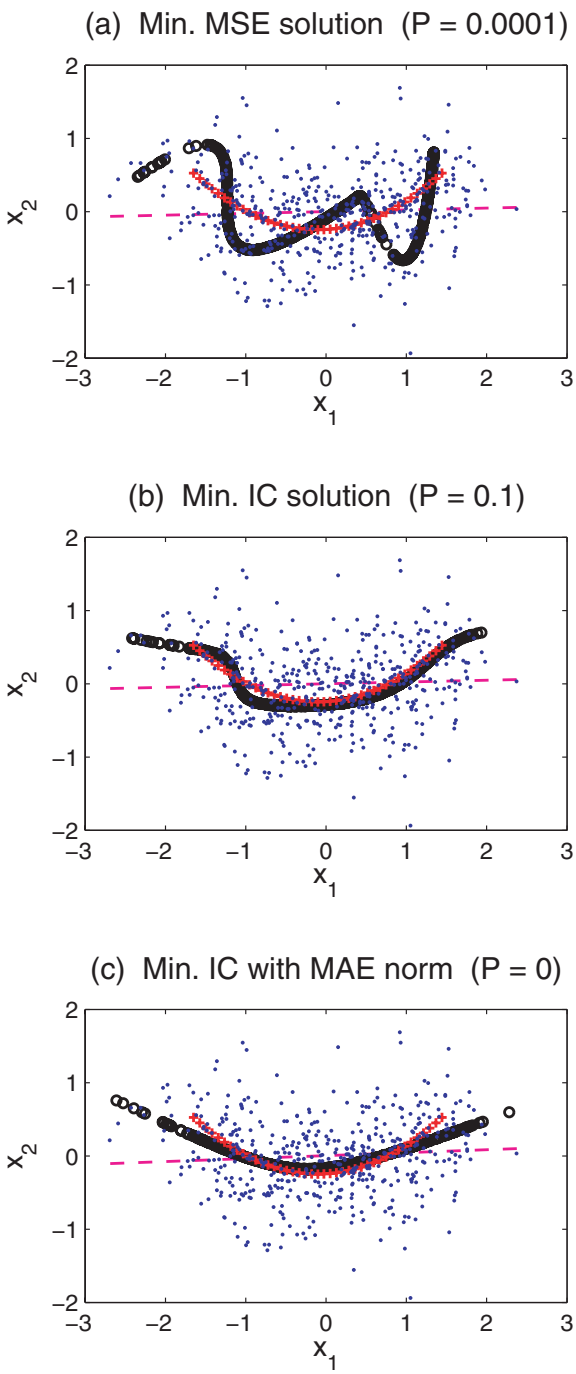

Fig. 8.5 The NLPCA solution (shown as densely overlapping black circles) for the synthetic dataset (dots), with the parabolic signal curve indicated by "+" and the linear PCA solution by the dashed line. The solution was selected from the multiple runs over a range of P values based on (a) minimum MSE, (b) minimum IC H, and (c) minimum IC together with the MAE norm (Reprinted from Hsieh 2007. With permission from Elsevier)

8.4 NLPCA for Closed Curves

While the NLPCA is capable of finding a continuous open curve solution, there are many phenomena involving waves or quasi-periodic fluctuations, which call for a continuous closed curve solution. Kirby and Miranda (1996) introduced an NLPCA with a circular node at the network bottleneck [henceforth referred to as the NLPCA(cir)], so that the nonlinear principal component (NLPC) as represented by the circular node is an angular variable θ, and the NLPCA(cir) is capable of approximating the data by a closed continuous curve. Figure 8.1b shows the NLPCA(cir) network, which is almost identical to the NLPCA of Fig. 8.1a, except at the bottleneck, where there are now two neurons p and q constrained to lie on a unit circle in the p-q plane, so there is only one free angular variable θ, the NLPC.

At the bottleneck in Fig. 8.1b, analogous to u in (8.8), we calculate the pre-states p_o and q_o by

$$p_o = \mathbf{w}^{(x)} \cdot \mathbf{h}^{(x)} + \overline{b}^{(x)}, \quad \text{and}$$
$$q_o = \tilde{\mathbf{w}}^{(x)} \cdot \mathbf{h}^{(x)} + \tilde{b}^{(x)}, \tag{8.18}$$

where $\mathbf{w}^{(x)}$, $\tilde{\mathbf{w}}^{(x)}$ are weight parameter vectors, and $\overline{b}^{(x)}$ and $\tilde{b}^{(x)}$ are offset parameters. Let

$$r = (p_o^2 + q_o^2)^{1/2}, \tag{8.19}$$

then the circular node is defined with

$$p = p_o/r, \quad \text{and} \quad q = q_o/r, \tag{8.20}$$

satisfying the unit circle equation $p^2 + q^2 = 1$. Thus, even though there are two variables p and q at the bottleneck, there is only one angular degree of freedom from θ (Fig. 8.1b), due to the circle constraint. The mapping from the bottleneck to the output proceeds as before, with (8.6) replaced by

$$h_k^{(u)} = f_3((\mathbf{w}^{(u)} p + \tilde{\mathbf{w}}^{(u)} q + \mathbf{b}^{(u)})_k). \tag{8.21}$$

When implementing NLPCA(cir), Hsieh (2001) found that there are actually two possible configurations: (i) A restricted configuration where the constraints $\langle p \rangle = 0 = \langle q \rangle$ are applied, and (ii) a general configuration without the constraints. With (i), the constraints can be satisfied approximately by adding the extra terms $\langle p \rangle^2$ and $\langle q \rangle^2$ to the objective function. If a closed curve solution is sought, then (i) is better than (ii) as it has effectively two fewer parameters. However, (ii), being more general than (i), can more readily model open curve solutions like a regular NLPCA. The reason is that if the input data mapped onto the p-q plane covers only a segment of the unit circle instead of the whole circle, then the inverse mapping from the p-q space to the output space will yield a solution resembling an open curve. Hence, given a dataset, (ii) may yield either a closed curve or an open curve solution. It uses $2lm + 6m + l + 2$ parameters.

Hsieh (2007) found that the IC H not only alleviates overfitting in open curve solution, but also chooses between open and closed curve solutions when using NLPCA(cir) in configuration (ii). The inconsistency index and the IC are now obtained from

$$I = 1 - \frac{1}{2}\left[C(p, \tilde{p}) + C(q, \tilde{q})\right], \quad \text{and}$$
$$H = \text{MSE} \times I, \tag{8.22}$$

where p and q are from the bottleneck (Fig. 8.1b), and $\tilde{p}$ and $\tilde{q}$ are the corresponding nearest neighbor values.

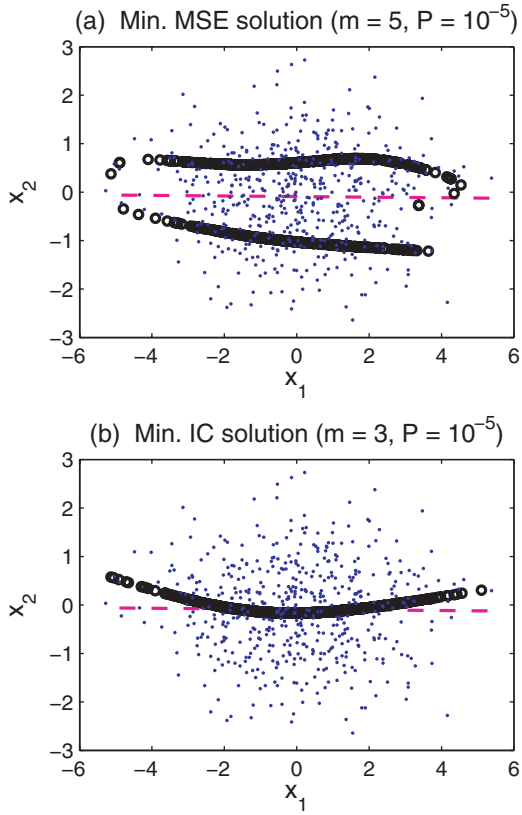

Fig. 8.6 The NLPCA(cir) mode 1 for a Gaussian dataset, with the solution selected based on (a) minimum MSE and (b) minimum IC. The PCA mode 1 solution is shown as a dashed line (Reprinted from Hsieh 2007. With permission from Elsevier)

For a test problem, consider a Gaussian data cloud (with 500 samples) in 2-dimensional space, where the standard deviation along the x_1 axis was double that along the x_2 axis. The dataset was analyzed by the NLPCA(cir) model with $m = 2, \ldots, 5$ and $P = 10$, 1, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 0. From all the runs, the solution selected based on the minimum MSE has $m = 5$ (and $P = 10^{-5}$) (Fig. 8.6a), while that selected based on minimum H has $m = 3$ (and $P = 10^{-5}$) (Fig. 8.6b). The minimum MSE solution has (normalized) MSE = 0.370, $I = 9.50$ and $H = 3.52$, whereas the minimum H solution has the corresponding values of 0.994, 0.839 and 0.833, respectively, where for easy comparison with the linear mode, these values for the nonlinear solutions have been normalized upon division by the corresponding values from the linear PCA mode 1. Thus the IC correctly selected a nonlinear solution (Fig. 8.6b) which is similar to the linear solution. It also rejected the closed curve solution of Fig. 8.6a, in favour of the open curve solution of Fig. 8.6b, despite its much larger MSE.

For an application of NLPCA(cir) on real data, consider the quasi-biennial oscillation (QBO), which dominates over the annual cycle or other variations in the equatorial stratosphere, with the period of oscillation varying roughly between 22 and 32 months. Average zonal (i.e. westerly) winds at 70, 50, 40, 30, 20, 15 and 10 hPa (i.e. from about 20 to 30 km altitude) during 1956–2006 were studied. After the 51-year means were removed, the zonal wind anomalies U at seven vertical levels in the stratosphere became the seven inputs to the NLPCA(cir) network (Hamilton and Hsieh 2002; Hsieh 2007). Since the data were not very noisy (Fig. 8.7), a rather complex model was used, with m ranging from 5 to 9, and $P = 10^{-1}$, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 0. The smallest H occurred when

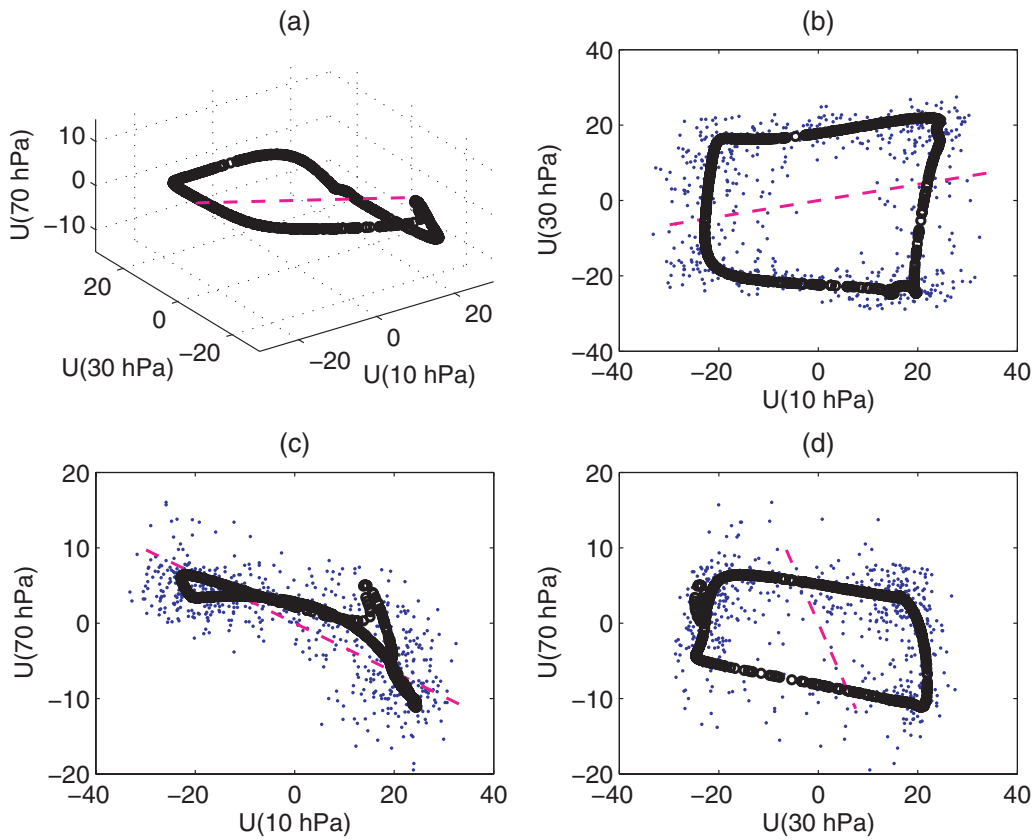

Fig. 8.7 The NLPCA(cir) mode 1 solution for the equatorial stratospheric zonal wind anomalies. For comparison, the PCA mode 1 solution is shown by the dashed line. Only three out of seven dimensions are shown, namely the zonal velocity anomaly U at the top, middle and bottom levels (10, 30 and 70 hPa). Panel (a) gives a 3-D view, while (b–d) give 2-D views (Reprinted from Hsieh 2007. With permission from Elsevier)

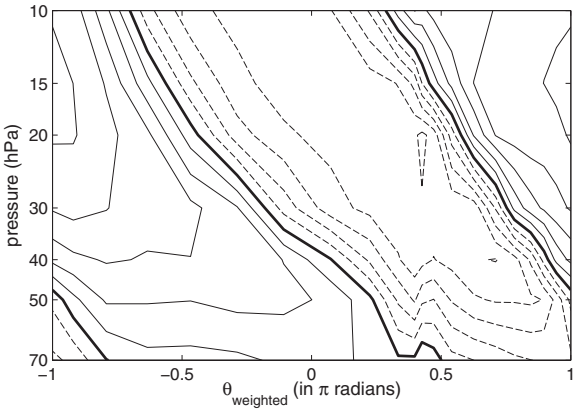

Fig. 8.8 Contour plot of the NLPCA(cir) mode 1 zonal wind anomalies as a function of pressure and phase θ_{weighted}, where θ_{weighted} is θ weighted by the histogram distribution of θ (see Hamilton and Hsieh 2002). Thus θ_{weighted} is more representative of actual time during a cycle than θ. Contour interval is 5 m s^{-1}, with westerly winds indicated by solid lines, easterlies by dashed lines, and zero contours by thick lines (Reprinted from Hsieh 2007. With permission from Elsevier)

$m = 8$ and $P = 10^{-5}$, with the closed curve solution shown in Fig. 8.7. Thus in this example, by choosing a rather large m and a small P, the H IC justified having considerable model complexity, including the wiggly behaviour seen in the 70 hPa wind (Fig. 8.7c). The wiggly behaviour can be understood by viewing the phase-pressure contour plot of the zonal wind anomalies (Fig. 8.8): As the easterly wind anomaly descends with time (i.e. as phase increases), wavy behaviour is seen in the 40, 50 and 70 hPa levels at θ_{weighted} around 0.4–0.5. This example demonstrates the benefit of having an IC to objectively decide on how smooth or wiggly the fitted curve should be.

The observed strong asymmetries between the easterly and westerly phases of the QBO (Hamilton 1998; Baldwin et al. 2001) are captured by this NLPCA(cir) mode – e.g. the much more rapid transition from easterlies to westerlies than the reverse transition, and the much deeper descend of the easterlies than the westerlies (Fig. 8.8). For comparison, Hamilton and Hsieh (2002) constructed a *linear* model of θ, which was unable to capture the observed strong asymmetry between easterlies and westerlies.

The actual time series of the wind measured at a particular height level is somewhat noisy and it is often desirable to have a smoother representation of the QBO time series which captures the essential features at all vertical levels. Also, the reversal of the wind from westerly to easterly and vice versa occurs at different times for different height levels, rendering it difficult to define the phase of the QBO. Hamilton and Hsieh (2002) found that the phase of the QBO as defined by the NLPC θ is more accurate than previous attempts to characterize the phase, leading to a stronger link between the QBO and northern hemisphere polar stratospheric temperatures in winter (the Holton-Tan effect) (Holton and Tan 1980) than previously found.

The NLPCA(cir) approach has also been used successfully in capturing the non-sinusoidal propagation of underwater sandbars off beaches in the Netherlands and Japan (Ruessink et al. 2004). Hsieh and Wu (2002) developed a nonlinear singular spectrum analysis method based on the NLPCA(cir) model.

8.5 Self-Organizing Maps

In this section, we examine a discrete version of NLPCA. The goal of *clustering* or cluster analysis is to group the data into a number of subsets or "clusters", such that the data within a cluster are more closely related to each other than data from other clusters. By projecting all data belonging to a cluster to the cluster center, data compression can be achieved.

The *self-organizing map* (SOM) method, introduced by Kohonen (1982, 2001), approximates a dataset in multidimensional space by a flexible grid (typically of one or two dimensions) of cluster centers. Widely used for clustering, SOM can also be regarded as a discrete version of NLPCA (Cherkassky and Mulier 1998).

As with many neural network models, self-organizing maps have a biological background (Rojas 1996). In neurobiology, it is known that many structures in the cortex of the brain are 2-D or 1-D. In contrast, even the perception of color involves three types of light receptors. Besides color, human vision also processes information about the shape, size, texture, position and movement of an object. So the question naturally arises on how 2-D networks of neurons in the brain can process higher dimensional signals.

Among various possible grids, the rectangular grid is most commonly used by SOM. For a 2-dimensional rectangular grid, the grid points $\mathbf{i}_j = (l, m)$, where l and m take on integer values, i.e. $l = 1, \ldots,$

L, $m = 1, \ldots, M$, and $j = 1, \ldots, LM$. (If a 1-dimensional grid is desired, simply set $M = 1$).

To initialize the training process, PCA is usually first performed on the dataset, and a grid $\mathbf{z}_j(0)$ is placed on the plane spanned by the two leading PCA eigenvectors, with each $\mathbf{z}_j(0)$ corresponding to $\mathbf{i}_j$ on the integer grid. As training proceeded, the initial flat 2D surface of $\mathbf{z}_j(0)$ is bended to fit the data. The original SOM was trained in a flow-through manner (i.e. samples are presented one at a time during training), though algorithms for batch training are now also available. In flow-through training, there are two steps to be iterated, starting with $n = 1$:

Step (i): At the nth iteration, select a sample $\mathbf{x}(n)$ from the data space, and find among the points $\mathbf{z}_j(n-1)$, the one with the closest (Euclidean) distance to $\mathbf{x}(n)$. Call this closest neighbor $\mathbf{z}_k(n)$, corresponding to the integer grid point $\mathbf{i}_k(n)$.

Step (ii): Let

$$\mathbf{z}_j(n) = \mathbf{z}_j(n-1) + \eta\, h(\|\mathbf{i}_j - \mathbf{i}_k(n)\|^2) \times [\mathbf{x}(n) - \mathbf{z}_j(n-1)], \tag{8.23}$$

where η is the learning rate parameter and h is a neighborhood or kernel function. The neighborhood function gives more weight to the grid points $\mathbf{i}_j$ near $\mathbf{i}_k(n)$, than those far away, an example being a Gaussian drop-off with distance. Note the distance of the neighbors are computed for the fixed grid points ($\mathbf{i}_j = (l, m)$), not for their corresponding positions $\mathbf{z}_j(n)$ in the data space. Typically, as n increases, the learning rate η is decreased gradually from the initial value of 1 towards 0, while the width of the neighborhood function is also gradually narrowed (Cherkassky and Mulier 1998).

As an example, consider the famous Lorenz 'butterfly'-shaped attractor from chaos theory (Lorenz 1963). Describing idealized atmospheric convection, the Lorenz system is governed by three (nondimensionalized) differential equations:

$$\dot{x} = -ax + ay, \quad \dot{y} = -xz + bx - y, \quad \dot{z} = xy - cz, \tag{8.24}$$

where the overhead dot denotes a time derivative, and a, b and c are three parameters. A chaotic system is generated by choosing $a = 10, b = 28$, and $c = 8/3$. The Lorenz data is fitted by a 2-dimensional SOM (from the MATLAB neural network toolbox) in Fig. 8.9, and by a 1-dimensional SOM in Fig. 8.10. The

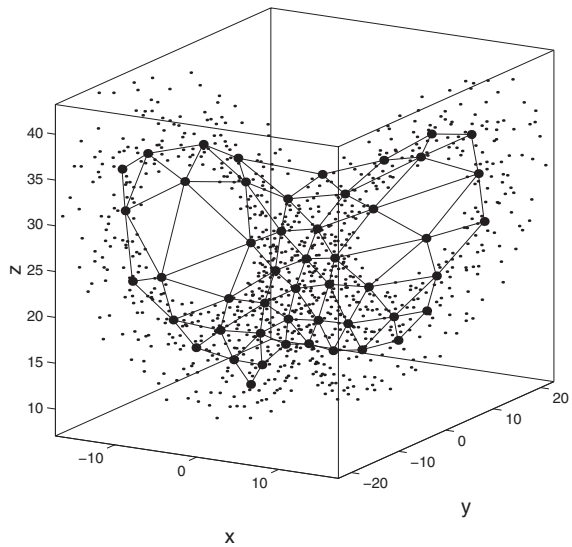

Fig. 8.9 A 2-dimensional self-organizing map (SOM) where a 7×7 mesh is fitted to the Lorenz (1963) attractor data

1-dimensional fit resembles a discrete version of the NLPCA solution found using auto-associative neural networks (Monahan 2000).

SOM has been applied to the classification of satellite-sensed ocean color (Yacoub et al. 2001), sea surface temperature (Richardson et al. 2003), sea level height (Hardman-Mountford et al. 2003), scatterometer winds (Richardson et al. 2003) and ocean currents

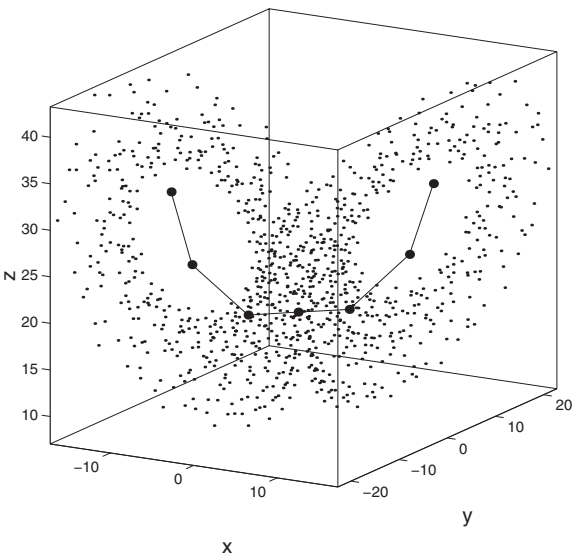

Fig. 8.10 A 1-dimensional self-organizing map (SOM) where a curve with seven nodes is fitted to the Lorenz (1963) attractor data

(Liu et al. 2006). Villman et al. (2003) applied SOM to not only clustering low-dimensional spectral data from the LANDSAT thematic mapper, but also to the high-dimensional hyperspectral AVIRIS (Airbourn Visible-Near Infrared Imaging Spectrometer) data where there are about 200 frequency bands. A 2-D SOM with a mesh of 40×40 was applied to AVIRIS data to classify the geology of the land surface.

Cavazos (1999) applied a 2×2 SOM to cluster the winter daily precipitation over 20 grid points in northeastern Mexico and southeastern Texas. From the wettest and driest clusters, composites of the 500 hPa geopotential heights and sea level pressure were generated, yielding the large scale meteorological conditions associated with the wettest and driest clusters. Cavazos (2000) and Cavazos et al. (2002) also applied SOMs with more clusters to other areas of the world.

8.6 NLPCA for Complex Variables

Complex principal component analysis (CPCA) is PCA applied to complex variables. In the first type of application, a 2-dimensional vector such as the wind (u, v) can be treated as a complex variable $w = u + \mathrm{i}v$ and analyzed by CPCA. In the second type of application, a real time-varying field can be complexified by the Hilbert transform and analyzed by CPCA, often called Hilbert PCA (von Storch and Zwiers 1999) to distinguish from the first type of application.

Earlier in this chapter, we have examined the auto-associative multi-layer perceptron NN approach of Kramer (1991) for performing nonlinear PCA. Here we will discuss how the same approach can be applied to complex variables, giving rise to nonlinear complex PCA (NLCPCA).

In the real domain, a common nonlinear activation function is the hyperbolic tangent function $\tanh(x)$, bounded between -1 and $+1$ and analytic everywhere. For a complex activation function to be bounded and analytic everywhere, it has to be a constant function (Clarke 1990), as Liouville's theorem states that entire functions (i.e. functions that are analytic on the whole complex plane) which are bounded are always constants. The function $\tanh(z)$ in the complex domain has an infinite number of singularities located at $(\frac{1}{2} + l)\,\pi i$, $l \in \mathbb{N}$. Using functions like $\tanh(z)$ (without any constraint) leads to non-convergent solutions (Nitta 1997).

Traditionally, the complex activation functions used focussed mainly on overcoming the unbounded nature of the analytic functions in the complex domain. Some complex activation functions basically scaled the magnitude (amplitude) of the complex signals but preserved their arguments (phases) (Georgiou and Koutsougeras 1992; Hirose 1992), hence they are less effective in learning non-linear variations in the argument. A more traditional approach has been to use a "split" complex nonlinear activation function (Nitta 1997), where the real and imaginary components are used as separate real inputs for the activation function. This approach avoids the unbounded nature of the nonlinear complex function but results in a nowhere analytic complex function, as the Cauchy-Riemann equations are not satisfied (Saff and Snider 2003).

Kim and Adali (2002) proposed a set of elementary activation functions with the property of being *almost everywhere* (*a.e.*) bounded and analytic in the complex domain. The complex hyperbolic tangent, $\tanh(z)$, is among them, provided the complex optimization is performed with certain constraints on z. If the magnitude of z is within a circle of radius $\frac{\pi}{2}$, then the singularities do not pose any problem, and the boundedness property is also satisfied. In reality, the dot product of the input and weight vectors may be $\geq \frac{\pi}{2}$. Thus a restriction on the magnitudes of the input and weights is needed.

The NLCPCA model proposed by Rattan and Hsieh (2004, 2005) uses basically the same architecture (Fig. 8.1a) as the NLPCA model of Kramer (1991), except all the input and output variables, and the weight and offset parameters are now complex-valued. The magnitude of input data are scaled by dividing each element in the rth row of the $l \times n$ data matrix $\mathbf{Z}$ (with l the number of variables and n the number of observations) by the maximum magnitude of an element in that row, so each element of $\mathbf{Z}$ has magnitude ≤ 1. The weights at the first hidden layer are randomly initalized with small magnitude, thus limiting the magnitude of the dot product between the input vector and weight vector to be about 0.1, and a weight penalty term is added to the objective function J to restrict the weights to small magnitude during optimization. The weights at subsequent layers are also randomly initialized with small magnitude and penalized during

optimization by the objective function

$$J = \langle \|\mathbf{z} - \mathbf{z}'\|^2 \rangle + P \sum_j |w_j|^2, \tag{8.25}$$

where $\mathbf{z}$ is the model output, $\mathbf{z}'$, the target data, w_j, the individual weights from hidden layers 1, 2 and 3, and P, the weight penalty parameter.

Since the objective function J is a real function with complex weights, the optimization of J is equivalent to finding the vanishing gradient of J with respect to the real and the imaginary parts of the weights (Rattan and Hsieh 2005). All the weights (and offsets) in the model are combined into a single weight vector $\mathbf{w}$. Hence the gradient of the objective function with respect to the complex weights can be split into (Georgiou and Koutsougeras 1992):

$$\frac{\partial J}{\partial \mathbf{w}} = \frac{\partial J}{\partial \mathbf{w}^{\mathrm{R}}} + i \frac{\partial J}{\partial \mathbf{w}^{\mathrm{I}}} \tag{8.26}$$

where $\mathbf{w}^{\mathrm{R}}$ and $\mathbf{w}^{\mathrm{I}}$ are the real and the imaginary components of the weight vector. The two components can be put into a single real parameter vector during nonlinear optimization using an algorithm for real variables.

The tropical Pacific wind anomalies (expressed as $w = u + iv$) have been analyzed by NLCPCA in Rattan and Hsieh (2004), where a comparison between the first mode of CPCA and that of NLCPCA revealed a large difference in the spatial anomaly patterns during strong El Niño episodes but a much smaller difference during strong La Niña episodes, indicating stronger nonlinearity was manifested in the El Niño side than the La Niña side of the oscillation.

The second type of NLCPCA application is for nonlinear Hilbert PCA. In Rattan et al. (2005), evolution of the offshore bottom topography at three sandy barred beaches were studied. All three sites were characterized by sandbars with interannual quasi-periodic offshore propagation. A bar cycle comprises bar birth in the inner nearshore, followed by up to several years of net offshore migration and final disappearance in the outer nearshore zone. CPCA was applied to the complexified topographic anomaly data, and the five leading CPCs were retained as inputs for the NLCPCA NN model. The first NLCPCA mode and the first CPCA mode of the topographic anomalies at Egmond aan Zee (The Netherlands) were compared. The topographic anomalies reconstructed from the nonlinear and linear mode were divided in 8 θ classes, each $\pi/4$ in width, where θ is the phase of the (nonlinear or linear) complex PC. Figure 8.11 shows how the shape of the topographic anomalies change with phase. The CPCA shows sinusoidal-shaped topographic anomalies propagating offshore, while the NLCPCA shows non-sinusoidal anomalies – relatively steep sandbars and shallow, broad troughs. The percentage variance explained by the first NLCPCA mode was 81.4% versus 66.4% by the first CPCA mode. Thus, using the NLCPCA as nonlinear Hilbert PCA successfully captures the non-sinusoidal wave properties which were missed by the linear method.

8.7 Summary and Discussion

The nonlinear generalization of the classical PCA method has been achieved by a number of different approaches (neural networks, principal curves, kernel methods, etc.). We have presented nonlinear PCA (NLPCA) using neural network methods. The tropical Pacific SST example illustrates that with a complicated oscillation like the El Niño-La Niña phenomenon, using a linear method such as PCA results in the nonlinear mode being scattered into several linear modes. In the study of climate variability, the wide use of PCA methods has created the somewhat misleading view that our climate is dominated by a number of spatially fixed oscillatory patterns, which may in fact be due to the limitation of the linear method.

By using a curve instead of a straight line to fit the data, NLPCA is susceptible to overfitting, yielding excessively wiggly curves. The introduction of a weight penalty parameter P in the objective function allowed the wiggles to be smoothed, but the lack of an objective selection criterion for P (hence the amount of smoothing) has been a weakness in NLPCA, until recent advances have allowed the objective selection of P and m (which controls the number of hidden neurons, hence model complexity). While the information criterion introduced by Hsieh (2007) for model selection worked well in climate datasets where there is one dominant signal (e.g. ENSO in the tropical Pacific SST; QBO in the stratospheric wind), it remains inadequate for dealing with datasets which contain two or more distinct signals of roughly comparable strength – e.g. in the extratropical N. Hemisphere climate, where there has been considerable controversy on the use of

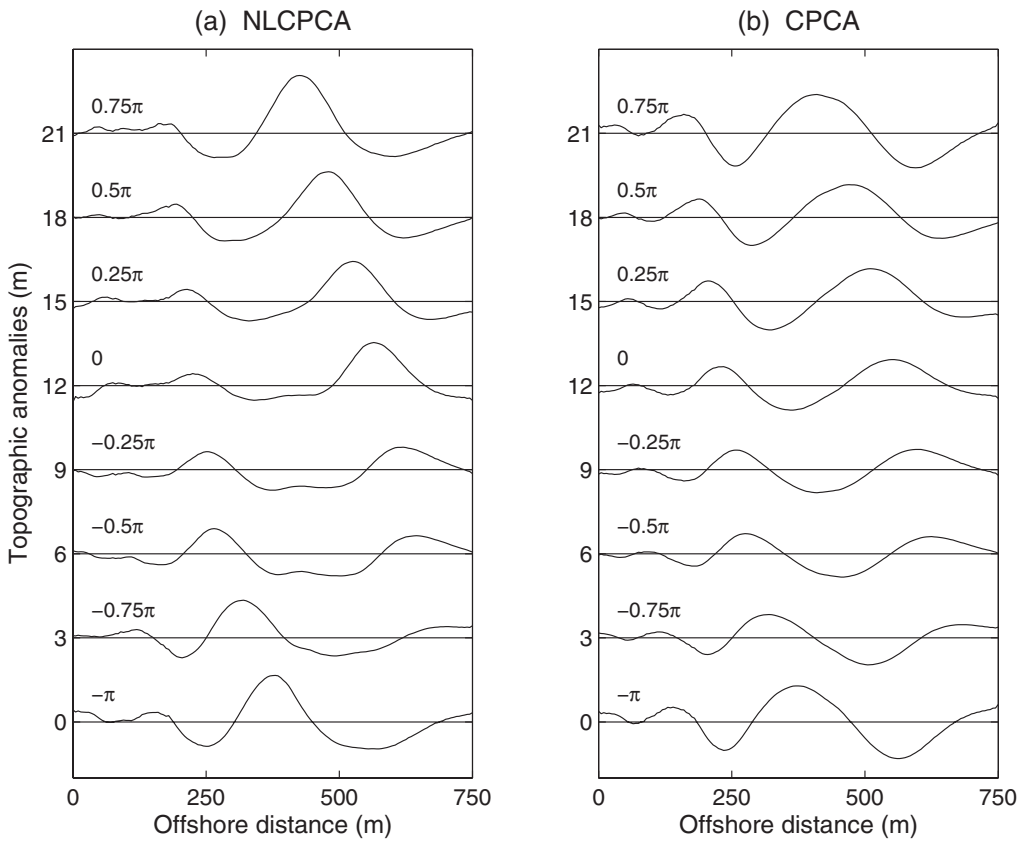

Fig. 8.11 Sequence of topographic anomalies as a function of the offshore distance at Egmond in $\pi/4$-wide θ classes centered around $\theta = -\pi$ to $\theta = 0.75\pi$ based on (a) NLCPCA mode 1 and (b) CPCA mode 1. The results for each phase class have been vertically shifted for better visualization. The phase generally decreases with time, so the anomalies gradually propagate offshore (Modified from Rattan et al. 2005)

NLPCA (Christiansen 2005, 2007; Monahan and Fyfe 2007), there are two signals of comparable magnitude, the Arctic Oscillation and the Pacific-North American teleconnection. The reason is that if there are two comparable signals, the total signal forms a 2-D surface whereas the NLPCA model will be trying to fit a 1-D curve to this surface, resulting in a hybrid mode with attributes from both signals. While it is possible to have two neurons in the bottleneck layer in the NLPCA network, so that a 2-D solution is extracted, there is no simple way to separate the two signals. Clearly more research is needed in developing model selection criteria in NLPCA for such complicated noisy datasets.

NLPCA has also been generalized to closed curve solutions, and to complex variables. Self-organizing maps (SOM) provide a discrete version of NLPCA. Due to space limitation, further generalization to nonlinear singular spectrum analysis and nonlinear canonical correlation analysis have not been presented here (see the review by Hsieh 2004).

References

Baldwin, M., Gray, L., Dunkerton, T., Hamilton, K., Haynes, P., Randel, W., Holton, J., Alexander, M., Hirota, I., Horinouchi, T., Jones, D., Kinnersley, J., Marquardt, C., Sato, K., & Takahashi, M. (2001). The quasi-biennial oscillation. *Reviews of Geophysics*, *39*, 179–229.

Bishop, C. M. (1995). *Neural networks for pattern recognition*. (482 pp.) Oxford: Oxford University Press.

Cavazos, T. (1999). Large-scale circulation anomalies conducive to extreme precipitation events and derivation of daily rainfall in northeastern Mexico and southeastern Texas. *Journal of Climate*, *12*, 1506–1523.

Cavazos, T. (2000). Using self-organizing maps to investigate extreme climate events: An application to wintertime precipitation in the Balkans. *Journal of Climate*, *13*, 1718–1732.

Cavazos, T., Comrie, A. C., & Liverman, D. M. (2002). Intraseasonal variability associated with wet monsoons in southeast Arizona. *Journal of Climate*, *15*, 2477–2490.

Cherkassky, V., & Mulier, F. (1998). *Learning from data* (441 pp.). New York: Wiley.

Christiansen, B. (2005). The shortcomings of nonlinear principal component analysis in identifying circulation regimes. *Journal of Climate*, *18*, 4814–4823.

Christiansen, B. (2007). Reply to Monahan and Fyfe's comment on "The shortcomings of nonlinear principal component analysis in identifying circulation regimes". *Journal of Climate*, *20*, 378–379. DOI: 10.1175/JCLI4006.1.

Clarke, T. (1990). Generalization of neural network to the complex plane. *Proceedings of International Joint Conference on Neural Networks*, *2*, 435–440.

Del Frate, F., & Schiavon, G. (1999). Nonlinear principal component analysis for the radiometric inversion of atmospheric profiles by using neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, *37*, 2335–2342.

Diaz, H. F., & Markgraf, V. (Eds.) (2000) *El Nino and the southern oscillation: Multiscale variability and global and regional impacts* (496 pp.). Cambridge: Cambridge University Press.

Georgiou, G., & Koutsougeras, C. (1992). Complex domain backpropagation. *IEEE Transactions on Circults and Systems II*, *39*, 330–334.

Hamilton, K. (1998). Dynamics of the tropical middle atmosphere: A tutorial review. *Atmosphere-Ocean*, *36*, 319–354.

Hamilton, K., & Hsieh, W. W. (2002). Representation of the QBO in the tropical stratospheric wind by nonlinear principal component analysis. *Journal of Geophysical Research*, *107*. DOI: 10.1029/2001JD001250.

Hardman-Mountford, N. J., Richardson, A. J., Boyer, D. C., Kreiner, A., & Boyer, H. J. (2003). Relating sardine recruitment in the Northern Benguela to satellite-derived sea surface height using a neural network pattern recognition approach. *Progress in Oceanograply*, *59*, 241–255.

Hastie, T., & Stuetzle, W. (1989). Principal curves. *Journal of the American Statistical Association*, *84*, 502–516.

Hastie, T., Tibshirani, R., & Friedman, J. (2001). *Elements of statistical learning: Data mining, inference and prediction* (552 pp.). New York: Springer.

Hirose, A. (1992). Continuous complex-valued backpropagation learning. *Electronic Letters*, *28*, 1854–1855.

Hoerling, M. P., Kumar, A., & Zhong, M. (1997). El Nino, La Nina and the nonlinearity of their teleconnections. *Journal of Climate*, *10*, 1769–1786.

Holton, J. R., & Tan, H.-C. (1980). The influence of the equatorial quasi-biennial oscillation on the global circulation at 50 mb. *Journal of the Atmospheric Sciences*, *37*, 2200–2208.

Hsieh, W. W. (2001). Nonlinear principal component analysis by neural networks. *Tellus*, *53*A, 599–615.

Hsieh, W. W. (2004). Nonlinear multivariate and time series analysis by neural network methods. *Reviews of Geophysics*, *42*, RG1003. DOI: 10.1029/2002RG000112.

Hsieh, W. W. (2007). Nonlinear principal component analysis of noisy data. *Neural Networks*, *20*, 434–443. DOI 10.1016/j.neunet.2007.04.018.

Hsieh, W. W., & Wu, A. (2002). Nonlinear multichannel singular spectrum analysis of the tropical Pacific climate variability using a neural network approach. *Journal of Geophysical Research*, *107*. DOI: 10.1029/2001JC000957.

Jolliffe, I. T. (2002). *Principal component analysis* (502 pp.) Berlin: Springer.

Kaiser, H. F. (1958). The varimax criterion for analytic rotation in factor analysis. *Psychometrika*, *23*, 187–200.

Kim, T., & Adali, T. (2002). Fully complex multi-layer perceptron network for nonlinear signal processing. *Journal of VLSI Signal Processing*, *32*, 29–43.

Kirby, M. J., & Miranda, R. (1996). Circular nodes in neural networks. *Neural Computation*, *8*, 390–402.

Kohonen, T. (1982). Self-organzing formation of topologically correct feature maps. *Biological Cybernetics*, *43*, 59–69.

Kohonen, T. (2001). *Self-Organizing maps* (3rd ed., 501 pp.) Berlin: Springer.

Kramer, M. A. (1991). Nonlinear principal component analysis using autoassociative neural networks. *AIChE Journal*, *37*, 233–243.

Liu, Y., Wieisberg, R. H., & Mooers, C. N. K. (2006). Performance evaluation of the self-organizing map for feature extraction. *Journal of Geophysical Research*, *111*. DOI: 10.1029/2005JC003117.

Lorenz, E. N. (1963). Deterministic nonperiodic flow. *Journal of the Atmospheric Sciences*, *20*, 130–141.

Monahan, A. H. (2000). Nonlinear principal component analysis by neural networks: Theory and application to the Lorenz system. *Journal of Climate*, *13*, 821–835.

Monahan, A. H. (2001). Nonlinear principal component analysis: Tropical Indo-Pacific sea surface temperature and sea level pressure. *Journal of Climate*, *14*, 219–233.

Monahan, A. H., & Fyfe, J. C. (2007). Comment on "The shortcomings of nonlinear principal component analysis in identifying circulation regimes". *Journal of Climate*, *20*, 375–377. DOI: 10.1175/JCLI4002.1.

Monahan, A. H., Fyfe, J. C., & Flato, G. M. (2000). A regime view of northern hemisphere atmospheric variability and change under global warming. *Geophysics Research Letters*, *27*, 1139–1142.

Monahan, A. H., Pandolfo, L., & Fyfe, J. C. (2001). The preferred structure of variability of the northern hemisphere atmospheric circulation. *Geophysical Research Letters*, *28*, 1019–1022.

Newbigging, S. C., Mysak, L. A., & Hsieh, W. W. (2003). Improvements to the non-linear principal component analysis method, with applications to ENSO and QBO. *Atmosphere-Ocean*, *41*, 290–298.

Nitta, T. (1997). An extension of the back-propagation algortihm to complex numbers. *Neural Networks*, *10*, 1391–1415.

Oja, E. (1982). A simplified neuron model as a principal component analyzer. *Journal of Mathematical Biology*, *15*, 267–273.

Philander, S. G. (1990). *El Niño, La Niña, and the southern oscillation* (293 pp.). San Diego, CA: Academic.

Preisendorfer, R. W. (1988). *Principal component analysis in meteorology and oceanography* (425 pp.). Amsterdam: Elsevier.

Rattan, S. S. P., & Hsieh, W. W. (2004), Nonlinear complex principal component analyis of the tropical Pacific interannual wind variability. *Geophysical Research Letters*, *31* (21), L21201. DOI: 10.1029/2004GL020446.

Rattan, S. S. P., & Hsieh, W. W. (2005). Complex-valued neural networks for nonlinear complex principal component analysis. *Neural Networks*, *18*, 61–69. DOI: 10.1016/j.neunet.2004.08.002.

Rattan, S. S. P., Ruessink, B. G., & Hsieh, W. W. (2005). Nonlinear complex principal component analysis of nearshore bathymetry. *Nonlinear Processes in Geophysics*, *12*, 661–670.

Richardson, A. J., Risien, C., & Shillington, F. A. (2003). Using self-organizing maps to identify patterns in satellite imagery. *Progress in Oceanography*, *59*, 223–239.

Richman, M. B. (1986). Rotation of principal components. *Journal of Climatology*, *6*, 293–335.

Rojas, R. (1996). *Neural networks – A systematic introduction* (502 pp.). Berlin: Springer.

Ruessink, B. G., van Enckevort, I. M. J., & Kuriyama, Y. (2004). Non-linear principal component analysis of nearshore bathymetry. *Marine Geology*, *203*, 185–197.

Saff, E. B., & Snider, A. D. (2003). *Fundamentals of complex analysis with applications to engineering and science* (528 pp.). Englewood Cliffs, NJ: Prentice-Hall.

Sanger, T. D. (1989). Optimal unsupervised learning in a single-layer linear feedforward neural network. *Neural Networks*, *2*, 459–473.

Schölkopf, B., Smola, A., & Muller, K.-R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, *10*, 1299–1319.

Tang, Y., & Hsieh, W. W. (2003). Nonlinear modes of decadal and interannual variability of the subsurface thermal structure in the Pacific Ocean. *Journal of the Geophysical Research*, *108*. DOI: 10.1029/2001JC001236.

Villmann, T., Merenyi, E., & Hammer, B. (2003). Neural maps in remote sensing image analysis. *Neural Networks*, *16*, 389–403.

von Storch, H., & Zwiers, F. W. (1999). *Statistical analysis in climate research* (484 pp.). Cambridge: Cambridge University Press.

Webb, A. R. (1999). A loss function approach to model selection in nonlinear principal components. *Neural Networks*, *12*, 339–345.

Yacoub, M., Badran, F., & Thiria, S. (2001). A topological hierarchical clustering: Application to ocean color classification. *Artificial Neural Networks-ICANN 2001, Proceedings. Lecture Notes in Computer Science*, 492–499.