ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 439-444

Nonlinear PCA: a new hierarchical approach

Matthias Scholz[1] and Ricardo Vigário[1,2]

[1] Fraunhofer Institut FIRST, Kekuléstr. 7, D - 12489 Berlin, Germany
[2] Neuronal Networks Research Centre, FIN - 02015 HUT, Finland
{scholz,rvigario}@first.fraunhofer.de

Abstract. Traditionally, nonlinear principal component analysis (NLPCA) is seen as nonlinear generalization of the standard (linear) principal component analysis (PCA). So far, most of these generalizations rely on a symmetric type of learning. Here we propose an algorithm that extends PCA into NLPCA through a hierarchical type of learning. The hierarchical algorithm (h-NLPCA), like many versions of the symmetric one (s-NLPCA), is based on a multi-layer perceptron with an auto-associative topology, the learning rule of which has been upgraded to accommodate the desired discrimination between components.
With h-NLPCA we seek not only the nonlinear subspace spanned by the optimal set of components, ideal for data compression, but we give particular interest to the order in which these components appear.
Due to its hierarchical nature, our algorithm is shown to be very efficient in detecting meaningful nonlinear features from real world data, as well as in providing a nonlinear whitening. Furthermore, in a quantitative type of analysis, the h-NLPCA achieves better classification accuracies, with a smaller number of components than most traditional approaches.

1 Introduction

When using any type of principal component analysis (PCA), linear or nonlinear, it is important to distinguish between applications where a mere reduction of the dimension is required and applications where the identification of a particular set of features, based on specific criteria, is important.
In the first set of applications, with clear emphasis to denoising and data compression, only a subspace with high descriptive power is sought. The individual features need not be unique. The only requirement is that the subspace explains, in a mean square error (MSE) sense, as much information contained in the data as possible.
Several neural network strategies exist that produce linear or nonlinear subspace PCA decompositions. However, such a decomposition is illposed in the sense that there are many possible solutions. Here we propose to enforce a hierarchical order of the principal components which yields essentially uncor-

ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 439-444

related features. Scaling these features to unit variance we obtain a whitening (sphering) transformation, which is a useful preprocessing step, for applications such as regression, classification or blind separation of sources.
A good algorithmic implementation of the hierarchical PCA should fulfill two important properties: scalability and stability. The former means that the first n components explain as much as possible of the variance in a n dimensional subspace of the data. The latter means that the i-th component of a n feature solution is identical to the i-th component of a m feature solution ($m \neq n$). Unlike the standard linear autoencoder, true PCA is an example of such hierarchical methods. Kernel PCA [6] is another algorithm that presents most of the hierarchical requirements, as it performs true PCA on a (nonlinear) feature space. The principal curves algorithm [2], however, behaves closer to the standard autoencoder, presenting no hierarchy.
Linear autoencoders will give hierarchically ordered features by training sequentially (deflationary), extracting the next feature on the remaining variance/error. However this does not work sufficiently in the nonlinear case. The remaining variance can not be considered regardless of the nonlinear mapping. In this paper we introduce a new hierarchical nonlinear PCA algorithm, denoted as h-NLPCA [7]. As a true nonlinear extension of PCA, this h-NLPCA is stable and fully scalable.
Oja's nonlinear PCA learning rule [5] should not be compared with the h-NLPCA, as it is, de facto, a linear algorithm with nonlinear training.

2 The algorithms

2.1 Autoencoder and linear PCA

The autoencoder, also known as *auto-associative neural network* or *bottleneck network*, is a multi-layer perceptron, with as many inputs as outputs and a smaller number of hidden feature units. During training, the targets for the output units are set to be equal to the inputs. The weights in the network are then taught to minimize the square error of the reconstruction. Because of this learning strategy, it can be shown that the linear autoencoder, with n features, converges to the n-th dimensional PCA subspace [1]. Note that in this learning configuration, the coded features have no particular order.

2.2 From linear to nonlinear PCA

An obvious extension of the linear autoencoder consists in the introduction of a nonlinear mapping by adding nonlinear hidden layers. This strategy is at the heart of the standard (often called symmetric) nonlinear principal component analysis network (s-NLPCA) [3], see Figure 1.

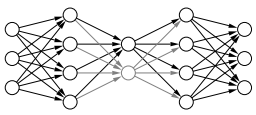


Figure 1: Five layer nonlinear autoencoder network ([3-4-2-4-3]). The mappings at input and output are nonlinear, whereas the three middle layers constitute a standard linear encoder.

ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 439-444

The s-NLPCA, as stated in the introduction, has no feature discrimination power. It is, therefore, mainly used to determine a nonlinear PCA subspace.
There are two strongly related ways to introduce hierarchy constraints to the feature space. In much the same way as in linear PCA, one is to force the i-th feature to account for the i-th highest variance projection. Another strategy would be to search in the original data space for the smallest mean squared reconstruction error while using the first i features. The former may be harder to solve than the latter, due to bounding conditions. Hence, we will present a learning strategy that focuses on the reconstruction MSE, $E = \frac{1}{dN}\sum_n^N \sum_k^d (x_k^n - \hat{x}_k^n)^2$, where x and $\hat{x}$ are, respectively, the original and the reconstructed data. N is the number of samples, d is the dimension. For simplicity, we will restrict our discussion to the case of two dimensional feature space. All the conclusions can be generalized to any other dimension.
E_1 and $E_{1,2}$ are the mean reconstruction errors when using, respectively, only the 1st or both the 1st and the 2nd features. In order to perform the h-NLPCA, we have to impose not only $\min E_{1,2}$ (as in the s-NLPCA), but also $\min E_1$. This can be done by minimizing the hierarchical error: $E_H = E_1 + E_{1,2}$.
Yet, we may want to give a stronger emphasis to optimal single or subspace feature explanations. This trade-off could be balanced by weighting the error terms E_1 and $E_{1,2}$ by use of a hyper-parameter α:

$$E_H = \alpha E_1 + E_{1,2} \quad ; \qquad \alpha \in (0, \infty).$$

However, selecting the optimal α increases the computational costs while the performance gain is moderate. Therefore we set α to 1 in all our experiments, which robustly balances the trade-off as shown in Figure 2 for star data.

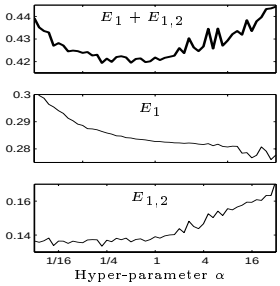


Figure 2: Dependency between the errors and the hyper-parameter α for a nonlinear five-layer autoencoder ([19-10-2-10-19]). The medians of 100 sweeps are plotted. The left side (corresponding to $\alpha \to 0$) is equivalent to a 2 dimensional encoding of a s-NLPCA network. The right side ($\alpha \to \infty$) is equivalent to the standard NLPCA with a single unit in the feature layer.

For training of the h-NLPCA, as with its symmetric counterpart, we used conjugate gradient descent. Yet, at each training iteration, the single error terms E_1, $E_{1,2}$ have to be calculated separately. This is performed in a s-NLPCA fashion by networks with one or two units in the feature layer, respectively. The gradient ∇E_H will be the sum of the single gradients $\nabla E_H = \alpha \nabla E_1 + \nabla E_{1,2}$. If a weight w_i does not exist, $\frac{\partial E_1}{\partial w_i}$ is set to zero.
A weight decay regularizer needs to be added: $E = E_H + \nu \sum_i w_i^2$. In most experiments, $\nu = 0.001$ was a good choice. Furthermore, to achieve more robust results, the weights of the nonlinear layer were initialized such that the sigmoidal nonlinearities worked in a linear regime ,which corresponds to starting the h-NLPCA network with the *simple* PCA solution.

ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 439-444

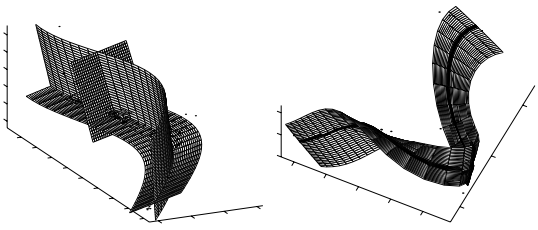
3 Experiments

To give experimental illustrations of the performance of h-NLPCA, we mostly used two different data sets with clear nonlinear behaviors. The first consisted of 19 dimension spectral information, gathered from 487 stars (details in [8]). The second data set is based on electromyographic (EMG) recordings for different muscle activities (labeled as 0, 10, 30, 50 and 70% of maximal personal strength). The one dimensional EMG is then embedded into a d dimensional space and analyzed as a recurrence plot [10]. Our data consisted then of 10 recurrence qualification analysis (RQA) for 35 samples (the 5 force levels for each of the 7 subjects). For more details on this data set, see [4].

3.1 Detecting nonlinearities

Figure 3 shows how h-NLPCA manages to correctly extract the nonlinear characteristics in both the *Star* and the *EMG* data. Note that, if in the first example the nonlinearities seem to be mild, it is clearly not the case for the second. Furthermore, in the EMG plotting, it seems that most of the variance is explained by the first two features.

Figure 3: The first three nonlinear components plotted into the space of the first three linear PCA components. Each grid represents two of the three nonlinear features, while the third is set to zero. These nonlinear components are extracted by using the hierarchical NLPCA. $E_H = E_1 + E_{1,2} + E_{1,2,3}$ (α is set to 1).



Star data EMG data

3.2 Feature extraction

Figure 4 plots the two first features (left and right column) against the 5 force levels for both linear PCA and h-NLPCA. As expected, the linear projections relate to the force level only in a nonlinear manner, whereas the first feature in the h-NLPCA shows clear linear relation to it. Note that the second nonlinear feature bears no relation to the force, as probably all the force information is explained by the first nonlinear component.

linear PCA hierarchical NLPCA

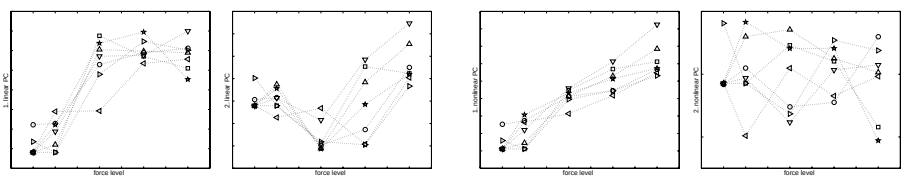


Figure 4: Linear and nonlinear PCA features, plotted against the 5 force levels.

ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 439-444

3.3 Nonlinear whitening (sphering)

If the nonlinearly uncorrelated features are scaled to unit variance, one could say that we have some sort of *nonlinear whitening*, or whitening in a nonlinear feature space. Figure 5 shows how h-NLPCA performs in this whitening task. Three dimensional data is generated in a 3/4 circle with added noise. For comparison purposes, we also show the linear and the s-NLPCA whitening. Note that, as expected, both the linear PCA and s-NLPCA have not been able to deal with the nonlinearity, whereas h-NLPCA normalizes the data to an almost spherical distribution.

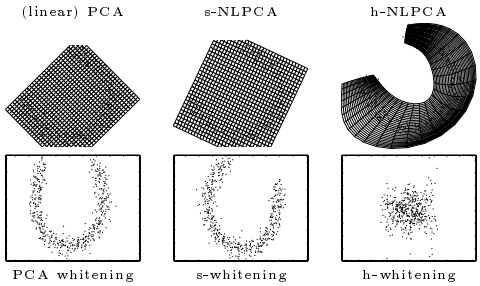


Figure 5: PCA, s-NLPCA, and h-NLPCA whitening of nonlinear correlated noisy data. On the top row are shown the joint distributions of the data along the first two features. The grids represent the coordinates of the feature space. On the bottom row are the whitening results, plotted on that same feature space.

3.4 Classification accuracy

A final experiment was carried out in order to see if the information contained in the h-NLPCA features can be used for classification purposes. Here, we have generated a 3 dimensional set of 3000 samples, belonging to 4 classes, with two dicotomical classifications (resembling the problem of male (M) and female (F) specimens of species A and B). The data was then nonlinearly transformed (2 directions using quadratic nonlinearities, whereas the remaining used a tanh). The classification was performed by a linear Support Vector Machine [9].

	classification 'F' to 'M'					classification 'A' to 'B'				
# features	1	**2**	3	10	20	1	2	**3**	10	20
linear PCA	40.6	**30.3**	30.3	—	—	48.3	50.0	**32.0**	—	—
s-NLPCA	45.9	**31.3**	—	—	—	50.0	50.0	—	—	—
h-NLPCA	48.5	**17.7**	17.7	—	—	50.0	50.0	**13.9**	—	—
kPCA σ10	48.5	**38.3**	9.0	3.5	3.5	50.0	50.0	**45.8**	13.2	13.0
kPCA σ0.5	48.5	**39.8**	39.7	3.5	1.4	50.0	50.0	**50.0**	35.5	2.2

Table 1: Testing error rates for an artificial data set using linear Support Vector Machines trained on the principal components of several feature extraction algorithms.

From Table 1, it is clear that one needs at least 2 features to descriminate between sex, and a 3rd for the species. In both classifications, h-NLPCA gives the best accuracy for the smallest number of components. As expected, linear PCA and s-NLPCA produced fairly poor classifications. The advantage of kernel PCA lies in the possibility of using very high dimensional feature spaces in which its classification performace improves significantly.

ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 439-444

4 Conclusion

We have introduced a new algorithmic approach to nonlinear PCA. This is based on the traditional autoencoder MLP, but has a hierarchical type of learning strategy. With such an algorithm we showed that it is possible to correctly detect nonlinearities in the data. Furthermore, due to it ordered nature, we can obtain the standard nonlinear PCA subspace, as well as the extraction of meaningful nonlinear orthogonal features.
It could be argued that the h-NLPCA and the kernel PCA share the same philosophy, as both of them perform some sort of nonlinear mapping to a feature space, where both perform true linear PCA representation of the data. Yet kernel PCA seems unable to recover the feature associated with the force, as the h-NLPCA did. One possible explanation for such a result is that kernel PCA was not developed for such tasks. On the other hand, the nonlinear mapping from h-NLPCA is tuned by the data, whereas the one from the kernel PCA is decoupled from the data. It should be expected that a particular setting for the kernel PCA would reach comparable results, but such a search for optimal kernel transformation exceeds the scope of this paper.

Acknowledgements:
The authors thank Klaus-R. Müller and other members of the IDA group for valuable discussions, David T. Mewett, Flinders University, Australia for providing the EMG data, and Jürgen Stock, Centro de Investigaciones de Astronomia (CIDA), Venezuela for the star data. R.V. was funded by the EU (Marie Curie fellowship HPMF-CT-2000-00813).

References

[1] P. Baldi and K. Hornik. Neural networks and principal component analysis: Learning from examples without local minima. *Neural Networks*, 2:53 – 58, 1989.

[2] T. Hastie and W. Stuetzle. Principal curves. *JASA*, 84:502–516, 1989.

[3] M. Kramer. Nonlinear principal component analysis using auto-associative neural networks. *AIChE Journal*, 37(2):233–243, 1991.

[4] D. Mewett, K. Reynolds, and H. Nazeran. Principal components of recurrence quantification analysis of EMG. *Proceedings of the 23rd Annual IEEE/EMBS Conference*, Oct.25-28 2001.

[5] E. Oja. The nonlinear PCA learning rule in independent component analysis. *Neurocomputing*, 17(1):25 – 46, 1997.

[6] B. Schölkopf, A. Smola, and K.-R. Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10:1299–1319, 1998.

[7] M. Scholz. Nonlinear PCA based on neural networks. Diploma Thesis, Dep. of Computer Science, Humboldt-University Berlin, in preparation. In German.

[8] J. Stock and M. Stock. Quantitative stellar spectral classification. *Revista Mexicana de Astronomia y Astrofisica*, 34:143 – 156, 1999.

[9] V. Vapnik. *The nature of statistical learning theory*. Springer, New York, 1995.

[10] C. Webber Jr and J. Zbilut. Dynamical assessment of physiological systems and states using recurrence plot strategies. *J. of Appl. Physiology*, 76:965 – 973, 1994.