Non-linear Two-Dimensional PCA

Joni Virta, University of Turku
Andreas Artemiou, Cardiff University

December 17th
CMStatistics 2022, London

Overview

Introduction

Dimension reduction for vectors

- **Linear dimension reduction** for a p-variate random vector x is achieved through linear projections,

$$a_1'x, \ldots, a_d'x,$$

or, more succinctly, as,

$$A'x.$$

- **Non-linear dimension reduction** simply replaces the linear maps with non-linear ones,

$$f_1(x), \ldots, f_d(x).$$

Dimension reduction for matrices

- **Linear dimension reduction** for a $(p_1 \times p_2)$ random matrix X is typically achieved through doubly linear projections,

$$a_j' X b_k, \qquad j \in \{1, \ldots, d_1\}, k \in \{1, \ldots, d_2\},$$

  or, more succinctly as,

$$A' X B.$$

Why $A'XB$?

Reduction via the form $A'XB$ has the following properties.

- The rows and columns of X are reduced separately.
- It does not matter whether we reduce the rows or the columns first.
- The reduced variable $A'XB$ is a matrix.

Can we achieve the same in the non-linear dimension reduction of random matrices?

Earlier approaches

Non-linear dimension reduction of matrices has been carried out by

Earlier approaches

Non-linear dimension reduction of matrices has been carried out by

- Vectorizing,

$$f_1(\mathrm{vec}\{X\}), \ldots, f_d(\mathrm{vec}\{X\}).$$

Earlier approaches

Non-linear dimension reduction of matrices has been carried out by

- Vectorizing,

$$f_1(\mathrm{vec}\{X\}), \ldots, f_d(\mathrm{vec}\{X\}).$$

- "Natural" non-linear maps from $\mathbb{R}^{p_1 \times p_2}$ to $\mathbb{R}$ (Signoretto et al., 2011),

$$f_1(X), \ldots, f_d(X).$$

Earlier approaches

Non-linear dimension reduction of matrices has been carried out by

- Vectorizing,

$$f_1(\mathrm{vec}\{X\}), \ldots, f_d(\mathrm{vec}\{X\}).$$

- "Natural" non-linear maps from $\mathbb{R}^{p_1 \times p_2}$ to $\mathbb{R}$ (Signoretto et al., 2011),

$$f_1(X), \ldots, f_d(X).$$

- Non-linear dimension reduction separately for each column (row) of X and aggregation (Kong et al., 2005; Nhat and Lee, 2007).

Our proposal for a general principle

Non-linear projection of a matrix

Let $X = \sum_{\ell=1}^{\mathrm{rank}(X)} \sigma_\ell u_\ell v_\ell'$ be an SVD of X.

The projection of X on (f, g) is

$$\sum_{\ell=1}^{\mathrm{rank}(X)} \sigma_\ell f(u_\ell) g(v_\ell).$$

Linear projections "$a'Xb$" are obtained as a special case.

Uniqueness

SVD is never unique!

This is not a problem for linear (f, g).

Uniqueness holds for non-linear (f, g) under the following conditions.

1. The non-zero singular values of X are almost surely distinct.
2. (f, g) are both either odd or even.

Application to PCA

Linear $(2D)^2$PCA

$(2D)^2$PCA is a matrix-version of PCA, resulting into reduction of the form $A'XB$ (Zhang and Zhou, 2005).

- The columns of A are d_1 leading eigenvectors of

$$\mathrm{E}[\{X - \mathrm{E}(X)\}\{X - \mathrm{E}(X)\}']$$

- The columns of B are d_2 leading eigenvectors of

$$\mathrm{E}[\{X - \mathrm{E}(X)\}'\{X - \mathrm{E}(X)\}]$$

Non-linear $(2D)^2$PCA via RKHS

- Let κ_1, κ_2 be pd-kernels that are both either odd or even.
- The RKHS $\mathcal{H}_1, \mathcal{H}_2$ induced by κ_1, κ_2 implicitly determine the class of non-linear functions we consider.
- For $X = \sum_{\ell=1}^{\mathrm{rank(X)}} \sigma_\ell u_\ell v_\ell'$, let

$$U = \sum_{\ell=1}^{\mathrm{rank(X)}} \sigma_\ell \{\kappa_1(\cdot, u_\ell) \otimes \kappa_2(\cdot, v_\ell)\}$$

  be the operator "representer" of X in the corresponding RKHS.
- Let,

$$H_1 = \mathrm{E}[\{U - \mathrm{E}(U)\}\{U - \mathrm{E}(U)\}^*],$$
$$H_2 = \mathrm{E}[\{U - \mathrm{E}(U)\}^*\{U - \mathrm{E}(U)\}].$$

Continued

- Let $a_1, \dots, a_{d_1}$ be leading d_1 eigenvectors of H_1.
- Let $b_1, \dots, b_{d_2}$ be leading d_2 eigenvectors of H_2.

The $d_1 \times d_2$ matrix Z of non-linear, two-dimensional principal components of X is now found element-wise as

$$z_{jk} := \sum_{\ell=1}^{\mathrm{rank}(X)} \sigma_\ell a_j(u_\ell) b_k(v_\ell).$$

Practical implementation

- Let κ_1, κ_2 be pd-kernels that are both either odd or even.
- Let $X_1, \ldots, X_n$ be a sample of matrices
- The spaces $\mathcal{H}_1, \mathcal{H}_2$ are typically infinite-dimensional, but we estimate finite-dimensional approximations of them using the sample.
- This leads to a linear *coordinate representation* for the method, depending on the sample only through two kernel matrices of size $nd \times nd$, where d is a tuning parameter corresponding to the rank of the used SVD.

Simulation

Non-linear matrix data?

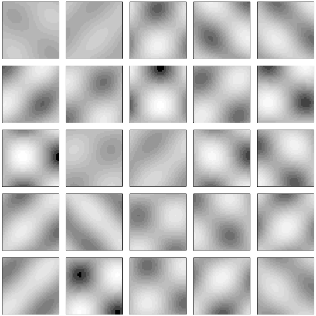
Can we say what kind of data the method excels in?

For example, the MNIST dataset is very linearly separable and can be solved with standard linear methods.

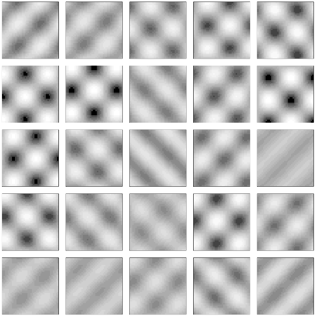
What do non-linearly separable images look like?

Simulated non-linear image data

Two-group data with a shifting checkerboard pattern.



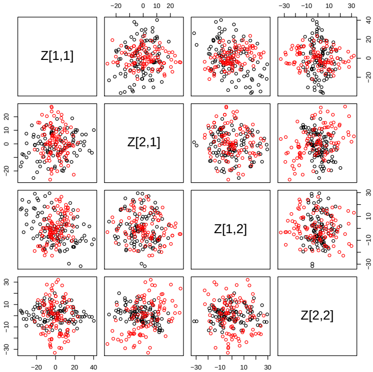
(a) 25 images from Group 1



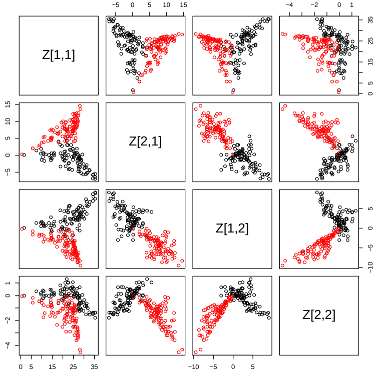
(b) 25 images from Group 2

Results

Paired scatterplots of the first 2×2 latent variables.



(a) Linear $(2D)^2$PCA



(b) Non-linear $(2D)^2$PCA (even RBF kernel).

Final remarks and to-do list

- Consistency and convergence rate.
- Every pd-kernel induces an odd and an even kernel (Krejnik and Tyutin, 2012). But which one should we use?

References

Kong, H., Wang, L., Teoh, E. K., Li, X., Wang, J.-G., and Venkateswarlu, R. (2005). Generalized 2D principal component analysis for face image representation and recognition. *Neural Networks*, 18(5-6):585–594.

Krejnik, M. and Tyutin, A. (2012). Reproducing kernel Hilbert spaces with odd kernels in price prediction. *IEEE transactions on neural networks and learning systems*, 23(10):1564–1573.

Nhat, V. D. M. and Lee, S. (2007). Kernel-based 2DPCA for face recognition. In *2007 IEEE International Symposium on Signal Processing and Information Technology*, pages 35–39. IEEE.

Signoretto, M., De Lathauwer, L., and Suykens, J. A. (2011). A kernel-based framework to tensorial data analysis. *Neural Networks*, 24(8):861–874.

Zhang, D. and Zhou, Z.-H. (2005). $(2D)^2$PCA: Two-directional two-dimensional PCA for efficient face representation and recognition. *Neurocomputing*, 69(1-3):224–231.