Data compression and regression based on local principal curves

Jochen Einbeck[1], Ludger Evers[2] and Kirsty Hinchliff[1]

[1] Department of Mathematical Sciences, Durham University, Durham, UK.
jochen.einbeck@durham.ac.uk

[2] Department of Statistics, University of Glasgow, Glasgow, UK.
ludger@stats.gla.ac.uk

Summary. Frequently the predictor space of a multivariate regression problem of the type $y = m(x_1, \dots, x_p) + \epsilon$ is intrinsically one-dimensional, or at least of far lower dimension than p. Usual modeling attempts such as the additive model $y = m_1(x_1) + \ldots + m_p(x_p) + \epsilon$, which try to reduce the complexity of the regression problem by making additional structural assumptions, are then inefficient as they ignore the inherent structure of the predictor space and involve complicated model and variable selection stages. In a fundamentally different approach, one may consider first approximating the predictor space by a (usually nonlinear) curve passing through it, and then regressing the response only against the one-dimensional projections onto this curve. This entails the reduction from a $p-$ to a one-dimensional regression problem.

As a tool for the compression of the predictor space we apply *local principal curves*. Taking things on from the results presented in [6], we show how local principal curves can be parametrized and how the projections are obtained. The regression step can then be carried out using any nonparametric smoother. We illustrate the technique using data from the physical sciences.

Key words: Dimension reduction, smoothing, principal curves, principal component regression.

1 Introduction

Principal curves are "smooth one-dimensional curves passing through the *middle* of a $p-$dimensional data set, providing a nonlinear summary of the data" [8]. Since Hastie & Stuetzle's pioneering work, principal curves have been further investigated, applied, and developed by quite a few researchers, and today exist at least half a dozen of algorithms for estimating them. These differ essentially in (i) what is understood of the "middle" of the data cloud; (ii) the algorithmic family ("top-down" or "bottom-up"); (iii) the criterion used for minimizing the error (if used at all).

Among the various principal curve concepts proposed are bias-corrected versions of the HS algorithm [1, 3], the polygonal line algorithm [10], the "principal curves of orientated points" (PCOPs, [7]), and the "local principal curves" (LPCs, [5]). PCOPs and LPCs are bottom-up algorithms, i.e., they proceed through the data cloud step by step and do not minimize a global error criterion. All other existing methods correspond to top-down algorithms, meaning that they start with some initial line which is then iteratively dwelled out until it fits satisfactorily through the data cloud and some global error criterion is minimized. Apart from the LPCs, which aim to approximate the density ridge, all concepts assume the existence of some theoretical "true" principal curve. Implementations of all algorithms mentioned above are publicly available and have been applied to a wide range of problems, including the recognition of hand-written characters [11], the reconstruction of river outlines or coastlines [5, 6], and path estimation from GPS tracks [2].

Surprisingly, the existing literature seems to be happy with knowing that principal curves can be estimated and that the resulting curve can be visualized, but has not proceeded with exploiting its benefits once it is there (with the notable exception of [3], who make use of HS principal curves for further pairwise compression of principal component scores). The value of their parametric counterpart, principal components, also brings to bear only when they are used for data compression or regression (e.g. [9], p. 66).

In Section 2, we consider a simple example taken from traffic engineering, illustrating how principal curves may be used for data compression and decompression. To motivate the necessity and value of nonparametric dimension reduction techniques, we proceed in Section 3 to a more complex application involving high-dimensional data from the future Galactic survey mission GAIA, and show how principal curves can be used for dimension reduction in multiple regression problems. In both cases, the technique used is that of local principal curves. We finish with a brief outlook on the extension to principal manifolds in Section 4.

2 Data compression with local principal curves

2.1 Local principal curves

Assume we are given a data set $X_1, \ldots, X_n$, with $X_i \in \mathbb{R}^p$, the intrinsic structure of which is to be described. Local principal curves [5, 6] are based on the idea that, at each point $x \in \mathbb{R}^p$ along a principal curve, the localized first principal component line forms approximately a tangent to the curve. They can be seen as a simple and fast approximation to the mathematically and computationally more demanding PCOPs [7]. Beginning at some starting point $x = x_0$, LPCs work successively through the data cloud, alternating between the following two steps:

(i) Calculate a localized center of mass $\mu^x = \sum_{i=1}^n w_i X_i$, where $w_i = K_H(X_i - x) X_i / \sum_{i=1}^n K_H(X_i - x)$.
(ii) Compute the 1^{st} local eigenvector γ^x of $\Sigma^x = (\sigma^x_{jk})_{(1 \le j,k \le p)}$, where $\sigma^x_{jk} = \sum_{i=1}^n w_i (X_{ij} - \mu^x_j)(X_{ik} - \mu^x_k)$ and μ^x_j denotes the j−th component of μ^x. Using a predetermined step size t_0, step from μ^x to $x := \mu^x + t_0 \gamma^x$.

The sequence of the local centers of mass μ^x makes up the local principal curve. Here, $K_H(\cdot) = |H|^{-1/2} K(H^{-1/2} \cdot)$, with a multivariate kernel K and a positive definite bandwidth matrix $H = \text{diag}(h_1^2, \ldots, h_p^2)$. Extensions to disconnected and branched curves were considered in [5] and [6], respectively, and are easily implemented by using suitable multiple starting points. Crossings can be handled conveniently using an angle penalization [5]. As in each iteration only points in the local neighborhood are considered, the algorithm is quite flexible, and, at the same time, robust to outlying data patterns.

2.2 Simple example: Speed-flow data

Fig. 1 displays data recorded on the Californian freeway FR57-N on 9th of July 2007. Each dot corresponds to the average of speed and flow values aggregated over 5-minute intervals. A LPC is fitted, using parameters $h_1 = h_2 = t_0 = 4$, and a starting point selected at random from the original data. The resulting points μ^x are symbolized by black squares in Fig. 1.

How does one go about connecting the points? For descriptive purposes a linear interpolation is sufficient, as it was handled in the original references [5, 6]. However, if the curve is to be used for further processing, it would need to be fully parametrized. One way of achieving this is to use a cubic spline (a piecewise polynomial function constructed from third order polynomials), yielding a continuous and differentiable smooth curve, as outlined below.

2.3 Parametrizations and Projections

For a fitted LPC consisting of L local centers of mass $\mu^{x_\ell} \equiv \mu^\ell = (\mu^\ell_1, \ldots, \mu^\ell_p)^T$, $\ell = 1, \ldots, L$, we seek a parametrization t such that the curve can be written as a function

$$f : \mathbb{R} \longrightarrow \mathbb{R}^p,\ t \mapsto (f_1(t), \ldots, f_p(t))^T,$$

attaining the L points μ^ℓ as outputs for certain parameter values t. Firstly, one end point is chosen to be the origin corresponding to $t = 0$. This is an arbitrary choice and we use the convention that t increases in the direction of γ^{x_0}. Technically, the curve is parametrized in three steps:

(i) Compute a discrete, preliminary parametrization $(s_\ell)_{(1 \le \ell \le L)}$, with the same origin as t, by adding up Euclidean distances between subsequent $\mu^\ell, \ell = 1, \ldots, L$.
(ii) For each $j = 1, \ldots, p$, lay a cubic spline through the set of points $(s_\ell, \mu^\ell_j)_{1 \le \ell \le L}$, yielding graphs $(s, \mu_j(s))$. Putting them together, one obtains a continuous and differentiable spline function $(\mu_1, \ldots, \mu_p)^T(s)$.

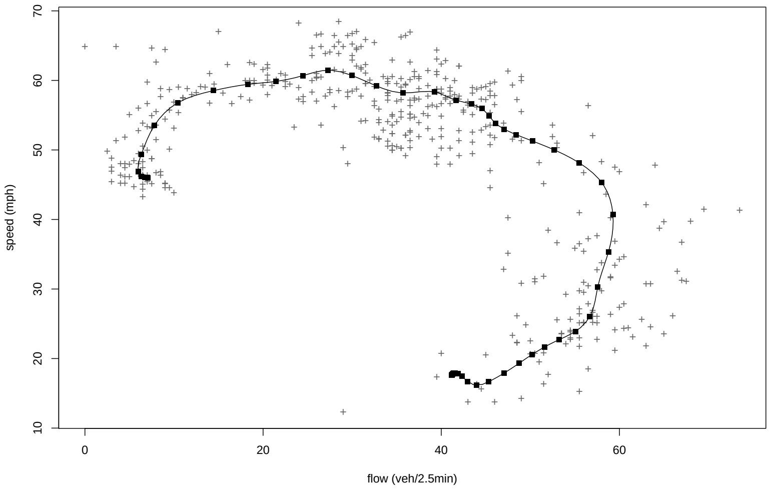

Fig. 1. Speed-flow data (+) and principal curve (solid curve) with local centers of mass (filled squares).

(iii)Recalculate the parameter through the arc length of this spline function: $t = \int_0^s \sqrt{(\mu_1'(u))^2 + \ldots + (\mu_p'(u))^2}\,du$.

It should be noted that no smoothing is involved in (ii) — this is a purely mechanical step interpolating the μ^ℓ through a string of cubic polynomials.

Once that this parametrization is established, each data point X_i, $i = 1, \ldots, n$, can be projected on the point of the curve nearest to it (in terms of Euclidean distances), yielding the projection index t_i. Data can be decompressed by evaluating the principal curve f, represented through the $p-$dimensional spline function, at t_i.

An illustration is given in Fig. 2. Note that, though the parametrization is *unit-speed* (i.e., distances in parameter space correspond to distances in data space along the principal curve), the projections are not *topology-preserving*: data points which are neighboring in data space are not necessarily neighboring in parameter space. This is a general property of data compression through principal curves, which distinguishes such methods from topology-preserving, but less interpretable mappings [12].

3 Regression with principal curves

3.1 GAIA data

GAIA is an astrophysics mission of the European Space Agency (ESA). A satellite is to be launched in 2011 which will undertake a detailed survey of

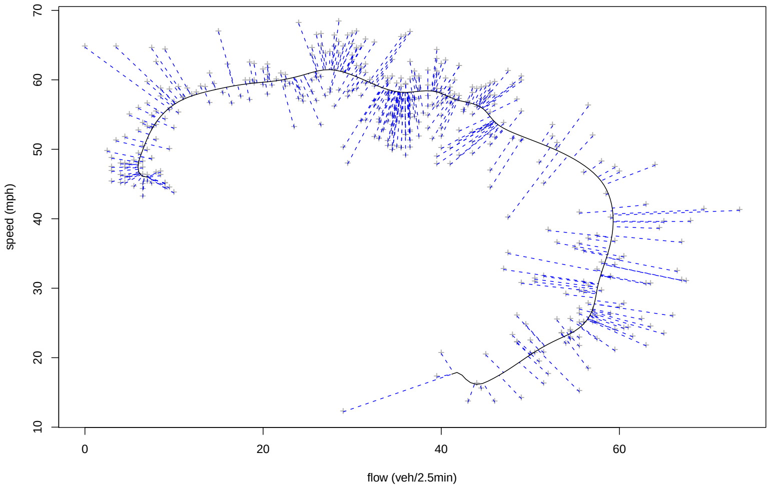

Fig. 2. Speed-flow data with principal curve (solid) and projections (dashed lines).

over 10^9 stars in our Galaxy and extragalactic objects. The aims of the mission are, among others, to classify objects into stars, galaxies, quasars, etc., and to determine astrophysical parameters ("APs": temperature, metallicity, gravity) from spectroscopic data (photon counts at certain wavelengths) [4]. Yet, one has to work with simulated data generated through complex computer models. Fig. 3 gives an example for a set of $n = 8286$ sixteen-dimensional photon counts simulated from APs through computer models.

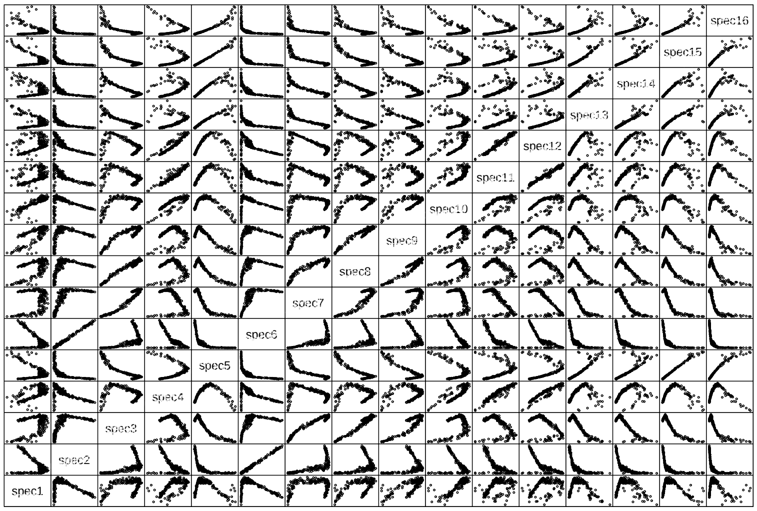

Fig. 3. GAIA data. Pairwise plots of 16-dim. photon counts.

Note that, for the actual estimation problem, the photon counts form the *predictor space* and the APs form the *response space*, this is opposite to the direction of simulation. As a consequence, the regression problem may be degenerate (i.e. one set of photon counts may be associated to two different APs). In the following, we will focus on the temperature, which features the least amount of degeneracy. We use a sample of size $n' = 1000$ from the original data for all following calculations. Fitting a multiple linear regression model for temperature against the sixteen individual photon counts leads to a residual standard error of 1978 on 983 degrees of freedom, with $t-$values for all variables around 0.65 and corresponding $p-$values around 0.51. We conclude that this does not constitute a useful model for the data.

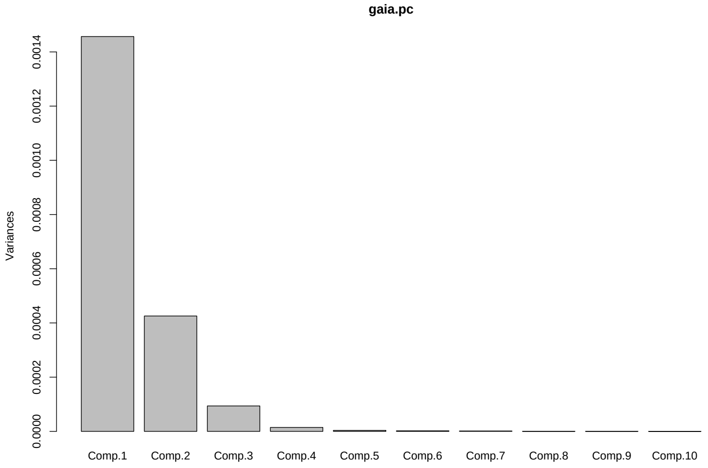

Fig. 4. Scree plot for GAIA data.

3.2 Principal component regression

The usual remedies in this case are model/variable selection procedures or dimension reduction techniques. The second one is obviously the most promising here. A common starting point for the application of the latter is the scree plot (Fig. 4), indicating that at most three components (these explain 98.9% of the total variance) appear to be sufficient to capture the information provided by these data. The usual way to continue is then to regress y = temperature against the scores associated with the largest three principal components, i.e.

$$y = \beta_0 + \beta_1 \text{score}_1 + \beta_2 \text{score}_2 + \beta_3 \text{score}_3 + \epsilon \tag{1}$$

Fitting this trivariate linear regression problem leads to a residual standard error of 2060 on 996 degrees of freedom, with $p-$values $< 2e - 16$ for all four regression parameters. The residual standard error of this model is naturally larger than the previous one, being just an approximation of the full linear model based on 98.9% of the available information. Nevertheless, this model is the more appropriate one. It remains the question whether the first three PC scores still feature some inner structure which we could exploit.

3.3 Dimension reduction with local principal curves

To investigate this, we produce a three-dimensional scatterplot of the PC scores, and shade lower temperatures with darker grey tones (Fig. 5 left). Clearly there is some curvilinear inner structure, which is informative for the target variable, temperature. Hence, the following is to do:

(1) Fit a principal curve through the 3-dim. data cloud of PC scores.
(2) Parametrize the principal curve and project all data points onto it.
(3) Fit temperature (or other APs) against the (1-dimensional) projections.

For task (1), a LPC is straightforwardly fitted[3] (Fig. 5 left). Alternatively, any other principal curve algorithm which provides access to the parametrization and allows for continuous projections could be used. This would include the HS algorithm, as far as it copes with the complexity of the data in itself. Algorithms based on piecewise line segments as in [10] are rather problematic for this purpose as projections tend to be clustered around the knots, unless the procedure outlined in Subsection 2.3 is additionally applied to them.

We perform task (2) as described in Subsection 2.3 and plot temperature against the projection indices. In (3), we are left with a simple one-dimensional nonparametric regression problem $y_i = m(t_i) + \varepsilon_i$. We used penalized smoothing splines to fit this model but any nonparametric smoother could be used. The smooth fit is shown in Fig. 5 (right).

3.4 Direct local principal curve regression

One may be wondering if there is a shortcut to this. Instead of the 2-stage strategy "PC+LPC" used so far, one could consider to fit the local principal curve *directly* through the $n' \times 16$ dimensional photon counts, as shown in Fig. 6. Comparing this result cursorily with Fig. 3, it appears that the data are reasonably represented (For a more quantitative evaluation of the accuracy of a principal curve, a coverage measure is available [5], and for the assessment of its precision bootstrap methods may be applied [2]). Indeed, the one-stage strategy is feasible in principle, and the results for both strategies are quite similar. However, as data gets sparse in high dimensions, the LPC may miss remote parts of the predictor space (the previously mentioned robustness may backfire here), which then get inadequately projected. The consequence of

[3] using the default settings of R function `lpc` for the parameters; these are: $h_j = 1/10 \times \{\text{range of variable } j\}$, and $t_0 = (1/d) \sum_j h_j$

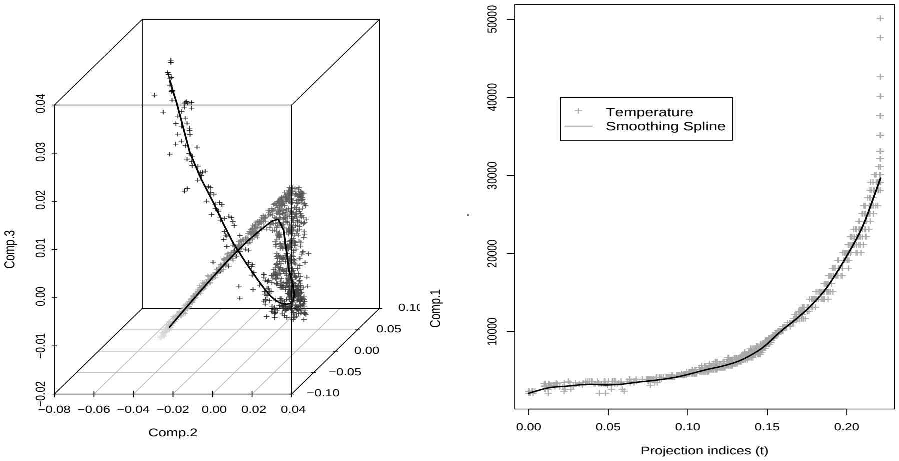

Fig. 5. Left: Scatterplot of first three principal component scores with local principal curve (—). The less intense the grey tone, the larger is the temperature; right: Temperatures fitted versus projection indices.

this is an increased sensitivity of the 16-dimensional LPC to the choice of the starting point compared to the 3-dimensional one. When approximating data through PCA in a first step, data are far less sparse in the second. Principal components cannot miss isolated data points as PC lines can be thought of as being infinitely long.

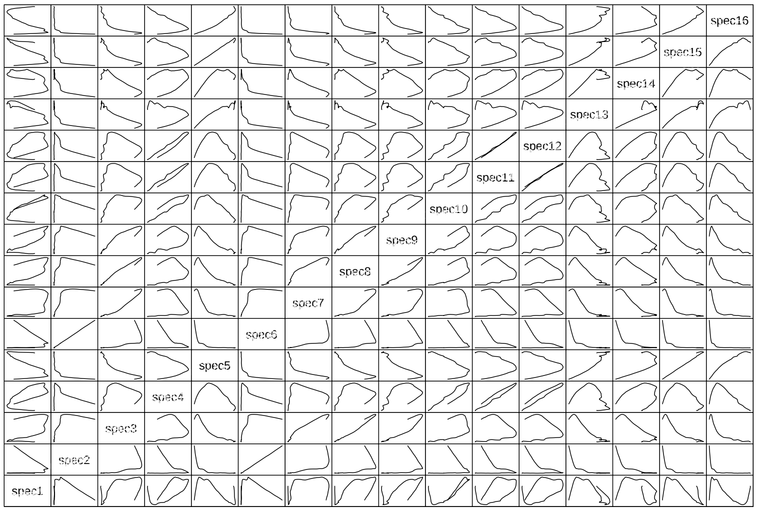

Fig. 6. Pairwise plot of LPC fitted through 16-dim. photon counts.

3.5 Prediction and Comparison

For a new observation x_{new} (i.e., here, a new set of spectra), prediction proceeds as follows: (i) Project x_{new} onto the LPC (either in one or two steps), giving t_{new}. (ii) Compute $\hat{y}_{new} = \hat{m}(t_{new})$ from the fitted nonparametric smoother (hereafter: NS).

Table 1 shows prediction errors for each 200 observations sampled from the training data set and the remaining $n - n' = 7286$ data points, respectively. Beside the methods discussed so far, we include an additive model using PC scores (a model just as in (1), but with all linear terms replaced by smooth functions; hereafter: AM).

Table 1. Prediction errors ($/10^3$) in comparison. $\hat{\varepsilon}_i$ is the difference between true and predicted temperature (LM= Linear Model, PC=Principal components, AM=Additive model, NS=Nonparametric Smoother)

		LM	PC+LM	PC+AM	PC+LPC+NS	LPC+ NS
Training	average ($\hat{\varepsilon}_i^2$)	4'119	4'395	1'318	2'633	2'215
data	median ($\hat{\varepsilon}_i^2$)	1'035	1'300	123	51	66
Test	average ($\hat{\varepsilon}_i^2$)	6'393	6'743	2'054	5'695	4'667
data	median ($\hat{\varepsilon}_i^2$)	723	808	147	45	46

As expected, and mentioned earlier, PC+LM is slightly worse than LM, and obviously PC+AM is better than PC+LM. The three nonparametric approaches clearly beat the parametric ones. The best median of squared residuals is taken by PC+LPC+NS, which is of a similar magnitude as that for LPC+NS and PC+AM. The mean of the squared residuals falls behind for the LPC-based methods compared to PC+AM. This can be explained as points close to the "end" of the data cloud are all projected onto the endpoint of the LPC, which leads to a degeneracy at either $t = 0$ or $t = t_{max}$ (or both). This is visible in Fig's 2 and 5 (right). So, though the LPC-based methods work very well for the large bulk of the data, they do not handle the few points close to the endpoints of the principal curve very well. Artificially extrapolating the fitted LPC beyond its natural endpoints may help to solve this problem.

4 Outlook

Local principal curves are well suited to compress complex high-dimensional data structures, as long as the intrinsic dimensionality of the data cloud is close to one. When the intrinsic dimensionality is two or larger, the extension to *local principal manifolds* should be considered. In particular, the GAIA data may be better approximated by a two-dimensional principal surface. This would be particular helpful for the prediction of other APs as gravity or metallicity, information on which tends to be orthogonal to the principal curve approximating the predictor space. The work on extending LPC methodology to higher-dimensional structures is currently ongoing, based on the idea of replacing the building block "localized principal component" by suitably orientated triangles or tetrahedrons.

Acknowledgements

We would like to thank Coryn-Bailer-Jones, leader of the group "Astrophysical parameters" based at MPIA Heidelberg, for providing the simulated GAIA data and explaining the background of the GAIA mission. The collaboration between the first two authors was supported by LMS Grant Ref 4709. The third author was funded by an EPSRC Vacation Bursary.

References

1. J. D. Banfield and A. E. Raftery. Ice flow identification in satellite images using mathematical morphology and clustering about principal curves. *Journal of the American Statistical Association* 87:7–16, 1992.
2. C. Brunsdon. Path estimation from GPS tracks. *Geocomputation 2007, NUI Maynooth, Ireland*.
3. K. Chang and J. Ghosh. Principal curves for nonlinear feature extraction and classification. *SPIE Applications of Artificial Neural Networks in Image Processing III*, 3307:120–129, 1998.
4. C.A.L. Bailer-Jones. Determination of stellar parameters with GAIA. *Astrophysics and Space Science*, 280:21–29, 2002.
5. J. Einbeck, G. Tutz, and L. Evers. Local principal curves. *Statistics and Computing*, 15:301–313, 2005.
6. J. Einbeck, G. Tutz, and L. Evers. Exploring multivariate data structures with local principal curves. In Weihs, C. and Gaul, W., editors, *Classification - The Ubiquitous Challenge*, pages 256–263. Springer, Heidelberg, 2005.
7. P. Delicado. Another look at principal curves and surfaces. *Journal of Multivariate Analysis*, 77:84–116, 2001.
8. T. Hastie and W. Stuetzle. Principal curves. *Journal of the American Statistical Association*, 84:502–516, 1989.
9. T. Hastie, R. Tibshirani and J. Friedman. *The Elements of Statistical Learning.* Springer, New York, 2001.
10. B. Kégl, A. Krzyżak, T. Linder and K. Zeger. Learning and design of principal curves. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22:281–297, 2000.
11. B. Kégl and A. Krzyżak. Piecewise linear skeletonization using principal curves. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24:59–74, 2002.
12. M. Peña, W. Barbakh and Colin Fyfe. Elastic maps and nets for approximating principal manifolds and their application to microarray data visualization. In A.N. Gorban et al., editors, *Principal Manifolds for Data Visualization and Dimension Reduction*, pages 131–150. Springer, Berlin, 2008.