Backscoring in Principal Coordinates Analysis*

Julian J. Faraway†

September 9, 2011

Abstract

Principal coordinates analysis refers to the low-dimensional projection of data obtained from distance matrix based methods such as multidimensional scaling. Principal components analysis also produces a low-dimensional projection of data and has the convenience of explicit mappings to and from the data space and the projected score space being readily available. The map from data to score is called called out-of-sample embedding. We call the map from score to data, backscoring. We discuss how these mappings may be obtained for a principal coordinates analysis and demonstrate applications for orientation, shape, functional and mixed data. The application to functional data shows how both phase and amplitude variation can be described together. Backscoring is helpful for interpreting the meaning of scores and in simulating new data. Data and R code necessary to reproduce the results are provided as supplemental materials.

Keywords: functional data analysis, mixed data, multidimensional scaling, orientation, shape, principal components

1 Introduction

The term principal coordinates analysis (PCO) was introduced by Gower (1966) and refers to a method for extracting a representation using low-dimensional coordinates from a distance matrix derived from the data. The method described in the paper is essentially classical multidimensional scaling (cMDS), although the same considerations apply more widely to other methods of dimension reduction using distance matrices. The name PCO suggests a similarity to the well-known principal components analysis (PCA). Let us contrast the two.

Suppose that we have an $n \times p$ data matrix X. In PCA, we use a $p \times p$ covariance or correlation matrix R to find an orthogonal representation of the data using scores $s_1, s_2, \ldots, s_n$. Hopefully, the first few components of the scores represent a low dimensional projection that captures most of the variation in the data. In PCO, we use an $n \times n$ distance matrix Q to also find a low dimensional projection of the data except the scores are called coordinates.

For quantitative real-valued data and using Euclidean distances, the two methods produce identical results, with the PCA computation being simpler. For other types of data and/or different distances, there is either no necessary equivalence or PCA may not be possible. PCA has been extended to other types of data such as curves in Besse and Ramsay (1986), Riemmanian manifolds in Pennec (2006) and more abstract data spaces in Huckemann et al. (2010). The fundamental requirement is that the observations

*to appear in Journal of Computational and Graphical Statistics

†Department of Mathematical Sciences, University of Bath, BA2 7AY, United Kingdom, jjf23@bath.ac.uk

lie in an inner product space so that PCA can proceed. Traditional PCA is characterized by the use of the standard Euclidean inner product. However, not all data of interest lie in an inner product space. Shapes, orientations, tensors, images and other more complex data may lie on a manifold that does not naturally constitute an inner product space, at least not without additional assumptions. Some data consist of variables of a mixture of types which complicate the application of PCA. On the other hand, PCO only requires a distance measure and thus can be applied in a much wider range of situations. One might view PCA as a special case of PCO that only works in limited circumstances.

In both a PCA and a PCO, we select a low dimensional representation using just the first few dimensions. We may apply statistical analysis to these dimension reduced scores. The low dimensional form is easier to analyse in comparison to the high dimensions of the original data. PCA possesses some convenient features which are apparently lacking in PCO.

In a PCA, there is an explicit mapping, $D \to S$, from the data space D to the score space S expressed in terms of linear combinations of the data. Thus when new data arrive, we can map them onto the score space. This is important when, for example, we wish to classify or predict from new cases. In a PCO, this is not so straightforward. *Out-of-sample embedding* or scoring can be used which we describe later.

In PCA, there is also a linear mapping from $S \to D$. If $dim(S) < dim(D)$ due to earlier dimension reduction, the mapping is equivalent to setting the coordinates for the discarded dimensions of S to zero. This mapping is helpful in interpreting the meaning of principal components. We can map the score, $(0,\ldots 0,c,0,\ldots)$, where c is in the position of the j^{th} coordinate for a range of values of c to the data space. Plotting or otherwise examining this sequence of points as c varies suggests the interpretation of the j^{th} principal component. This is particularly useful for less common data types such as shapes or functions. Thus we can map conclusions for models on the score space back to the reality of the data space. We can also use the mapping to simulate new cases from approximately the same distribution as the data. We simulate new scores in the score space and then map back to the original data space. This is usually much easier and more effective than trying to simulate data directly in the data space. In a PCO, there is no obvious $S \to D$ mapping. The construction of such mappings, which we call *backscoring*, is the topic of this article.

The key to adding the desirable features of a PCA to a PCO is to construct the $D \to S$ and $S \to D$ mappings. We describe these in Section 2. In Section 3, we provide four example analyses of different data types and discuss the application to yet other types of data. In Section 4, we conclude with a discussion of the methods.

2 Scoring and backscoring

In this section we describe scoring, by which we mean the $D \to S$ mapping, and backscoring, meaning the $S \to D$ mapping. Scoring is also called out-of sample embedding. We will use classical MDS for the PCO, but the methods can be adapted to handle other distance matrix-based methods.

Classical multidimensional scaling starts by forming a matrix B from distance matrix Q such that:

$$B_{ij} = -(Q_{ij}^2 - Q_{i\cdot}^2 - Q_{\cdot j}^2 + Q_{\cdot\cdot}^2)/2$$

where the dots in the subscripts indicate that means are taken over the index. We then perform the eigendecomposition so that $B = SS^T$ with eigenvalues $\lambda_1 \geq \lambda_2 \geq \ldots \lambda_n$. The columns of matrix S contain the principal coordinates or scores.

Now suppose we consider a new data point x_{new} and we would like to place it within this coordinate system. It is possible simply to add this point to the data and recompute the eigendecomposition. However, this would change the original coordinates. Instead we might wish to view the original data as

establishing a coordinate system and want to place the new point within these coordinates. Gower (1968) describes one way this may be achieved:

$$s_{new} = \Lambda^{-1} S^T (d^2(\mathbf{x}, \bar{x}) - d^2(\mathbf{x}, x_{new}))/2 = \phi(x_{new}) \tag{1}$$

where $\Lambda = diag(\lambda_1, \ldots, \lambda_n)$, d is the distance function and $\bar{x}$ is the centroid. However, $d^2(\mathbf{x}, \bar{x}) = -diag(B)$ so explicit computation of the centroid is not yet needed. This computation requires only the distances $d(\mathbf{x}, x_{new})$ of the new point to the original points. In practice, we retain only the first few coordinate directions so appropriately reduced versions of Λ and S would be used, and s_{new} represents a projection onto the reduced score space.

One might object to this method on the grounds that it introduces an additional dimension. There are several more recent approaches to this problem. Trosset and Priebe (2008) describe a method with a more satisfying motivation that can be used for adding multiple new points. For the addition of single point, a simple approximation to this method turns out to be equivalent to the proposal of Anderson and Robinson (2003). Further, Trosset and Tang (2010) show that the methods proposed by Bengio et al. (2004) and de Silva and Tenenbaum (2003) join this equivalence class. Remarkably, we observe that the method of Gower described above is also identical in that the scores match on the dimensions computed in common. Thus several different motivations lead to the same solution. We need this simply computed solution because, as will subsequently become clear, we need to repeat this operation many times.

Now let us consider the backscoring problem of mapping $S \to D$. If we use m-dimensional coordinates in S and $m = p$, then a unique solution may exist. However, due to dimension reduction, $m < p$, each s_{new} will correspond to a set of solutions in D. Within this set, a natural solution is the one closest to the mean, that is:

$$x_{new} = \arg \min_{s_{new} = \phi(x)} d(x, \bar{x}) \tag{2}$$

This can be justified by making an analogy to PCA. For $m < p$, the mapping of s_{new} to x_{new} is equivalent to the mapping of the augmented scores (s_{new}, v), a vector of length p, to x_{new} for any v, subject to the condition that $d(x_{new}, \bar{x})$ is minimized. The solution in the PCA case is simply $v = \mathbf{0}$. In PCO, it would be difficult to augment the score with zeroes up to dimension p for a number of reasons. Firstly, later eigenvalues λ_i in many PCO applications are frequently negative. Normally, this can be ignored as only the first few coordinates are used, but here this would be problematic. Secondly, finding the x_{new} that would satisfy such an equation is more difficult than the constrained minimization problem we do solve.

Actual computation of the $S \to D$ mapping may be difficult. We envisage use of this method where the application of PCA is difficult or impossible which implies a lack of structure on the data space. With no inner product and no vector space, an exact solution will be difficult to find. Let us consider a range of situations:

1. In the most extreme case, we have only the data observed and a way to compute distances between cases. Suppose for example we wish to visualize the first principal component. We want to find the set of x corresponding to scores of the form $(c, 0, 0, \ldots 0)$, but the best we can do is select observed cases where the second and remaining components are all small. Thus we will have only a rough solution and only for those c available in the data. If, as is usually the case, we only have scores in a few dimensions, we will not even know the values in the higher dimensions which may be quite different from zero. Thus our attempts to interpret the first PC will be handicapped. In practice, this is often how PCO results are interpreted, with variable success.

2. In other cases, we may have access to other examples of data not used in the construction of Q, but which are assuredly members of D. Alternatively, we may be able to modify existing cases to

produce new data values. For example, three of the datasets we will analyse are drawn from the study of human motion. We may construct a Q based on a particular set of observed motions, but there often exist other databases of similar motions. There are also several accepted methods of modifying existing motion data to produce authentic new motion data. In such a situation, we can use the $D \to S$ mapping to determine the scores for the new cases, giving us a richer set of potential approximate solutions to $S \to D$ problem. This offers some improvement over the data only case.

3. Given only the ability to construct distances between points, it is still possible to construct convex combinations of cases. For example, consider the formation of a mean from objects $x_1, \ldots x_n \in D$. The usual sample mean is $n^{-1}\sum_i x_i$. But this may not be meaningful for objects in some spaces. Instead we can use the Fréchet mean:

$$\text{argmin}_{x \in D} \sum_{i=1}^{n} d(x, x_i)$$

This demonstrates how the centroid can be found for (1) using only distances. For $\mathbb{R}^p$, if we define $d(x,y) = ||x-y||^2$, the two definitions are identical. We can generalize to convex combinations of l cases by finding

$$\text{argmin}_{x \in D} \sum_{i=1}^{l} w_i d(x, x_i) \quad \text{where} \quad \sum_i w_i = 1$$

This has the advantage of being (potentially) computable on non-vector spaces and may provide enough flexibility to solve the backscoring problem (2).

4. Another possibility that is used in some PCO applications is to regress x on the scores. This results in a model that can perform an $S \to D$ mapping. It can also aid in the interpretation of the coordinate directions. However, this is at best an approximate solution to the backscoring problem as posed here. Another difficulty is that x is at least multivariate and, as in the examples we will consider, may consist of objects such as shapes or functions. Performing a regression on these objects is not easy (indeed the PCO is sometimes a vehicle to avoid such regressions).

We will pursue solutions based on the third approach above. Unlike, the $D \to S$ case, there are no generic solutions. We will present four examples, each with a different type of data, where we present customized solutions, but with a preference for algorithms that require fewer properties (such as the existence of inner products) so that the ideas might be easier to extend to yet other data types. We are not aware of previous research on this problem. de Leeuw and Groenen (1997) discuss "inverse multidimensional scaling" meaning the construction of the set of distance matrices Q that would give rise to the chosen set of scores which is a different problem.

3 Examples

3.1 Orientations

The orientation of an object in a three dimensional space lies in a nonlinear space, $SO(3)$. Although there are several ways of representing an orientation, it is convenient to use a quaternion $\mathbf{q}$ written as $q = w + \mathbf{i}x + \mathbf{j}y + \mathbf{k}z$ where $x, y, z, w \in \mathbb{R}$ and $\mathbf{i}, \mathbf{j}, \mathbf{k}$ are imaginary numbers such that $\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1$. Orientations can be identified with unit quaternions, where $||\mathbf{q}|| = 1$, and thus represent a 3D manifold S^3 embedded in a 4D space. See Prentice (1986) for an example of using quaternions to model orientations.

There is a PCA-type method available here. We construct a tangent space with an origin at the centre of the data, project the data onto this 3D tangent space, perform the PCA and then project back on to the orientation space. The mappings to and from the tangent space are conveniently computed with quaternions by using the log and exponential maps described in Grassia (1998). This will be satisfactory provided the data are not too dispersed as this will reduce the distortion due to mapping onto the tangent space. This will provide a useful comparison to the PCO approach described below.

Data for the example are drawn from those described in Faraway and Choe (2009) which also contains more details on the use of quaternions to model orientations. The data consist of 279 orientations of the right hand at rest near the knee prior to making a reach from a seated position. The maximum angular distance between any two points is 71.6° indicating that the data are not that tightly clustered.

To perform the PCA, we compute the mean orientation and centre the data at this point. We then compute the log map, projecting the data onto the tangent space at the origin which in this case lies in three dimensions. The principal components explain 54.4%, 34.0% and 11.6% of the standard deviation respectively. A 3D plot of the data in the tangent space is shown in the first panel of Figure 1. Each point in the plot corresponds to a hand orientation.

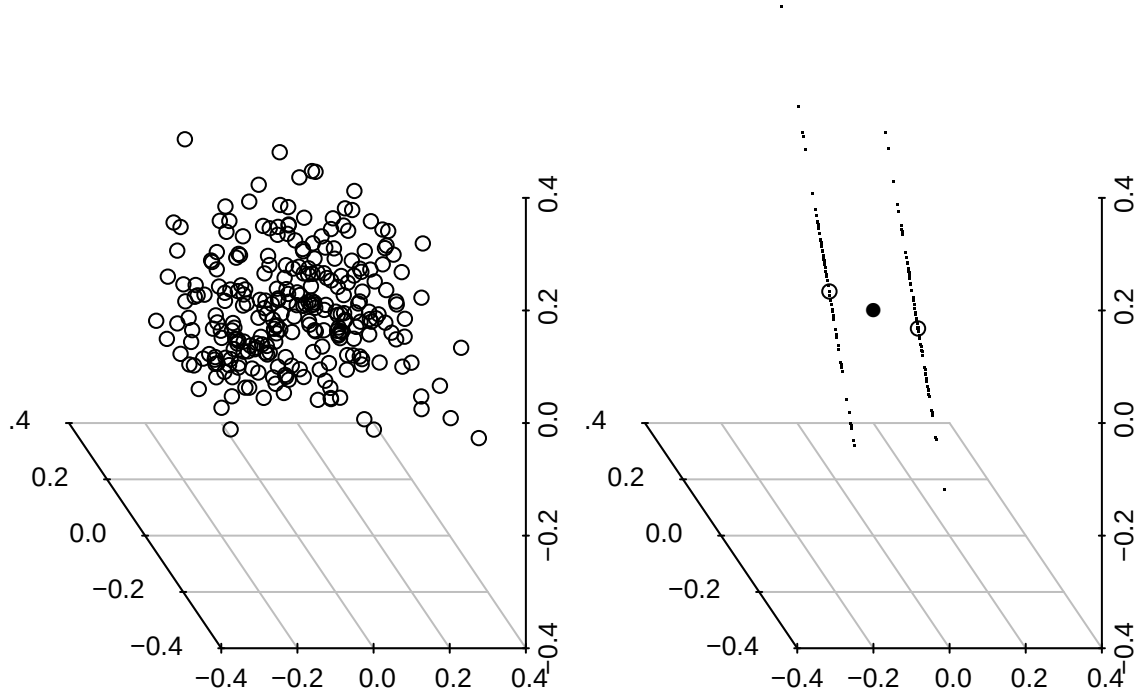

Figure 1: Tangent space projections of the hand orientation data. The left panel shows the data while the right panel shows a representation of the first principal coordinate. The origin(centroid) is marked by a solid circle while the solutions for ± 1 in the direction of the first principal coordinate are shown by open circles. The candidate solutions are shown as dots.

The PCO analysis requires a distance measure. We use the shortest angular distance between two points in S^3. We compute the distance matrix Q and then perform classical MDS. Eigenvalues for the decomposition indicate an almost identical explanation of the variation as that seen in the PCA. The scores are also very similar. The $D \rightarrow S$ mapping using the general method described in (1) requires only the computation of the distance of the new orientation to the existing ones. This is straightforward.

Now consider the $S \to D$ mapping. If we solve the MDS in three dimensions, there is a 1-1 correspondence that can be solved by inverting the $D \to S$ mapping. This somewhat difficult to compute, but not very interesting since we will want to consider situations where the dimension of the MDS solution is rather less than the full dimensionality. For this reason, let us consider the 2D MDS solution where the $S \to D$ mapping will require finding the best solution in D which is three dimensional. We also wish to develop an algorithm which uses only the distance measure and weighted averages of the orientations. In this case, we have rather more structure than that and so a more direct solution is possible. However, we want an algorithm that will work for the more difficult data types and thus forgo the use of this available structure.

To generate potential solutions for given s_{new}:

1. Select three orientations, x_1, x_2 and x_3 at random from the data and compute $s_i = \phi(x_i)$ for $i = 1, 2, 3$.
2. Find w_i such that $\sum_i w_i s_i = s_{new}$ and $\sum_i w_i = 1$.
3. Compute the weighted average x_w of the x_i's given weights w_i. Find $s_w = \phi(x_w)$.
4. Adjust the weights until $s_w = s_{new}$. (We find that only a one-step adjustment is sufficient in this example).

We repeat these steps for many randomly selected triplets. We select the solution that is closest to the centroid. The method is illustrated in Figure 2.

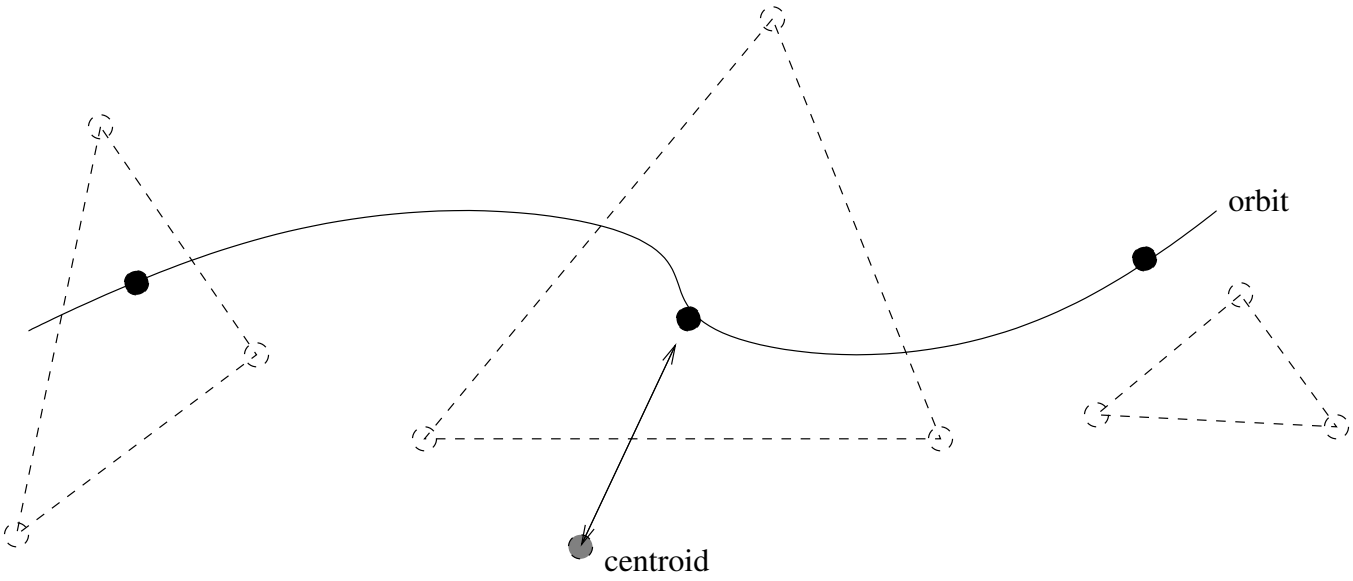

Figure 2: A backscoring method: randomly selected triplets of cases are used to generate potential solutions along an orbit corresponding to the desired score. The chosen solution is that closest to the centroid.

We note that:

- It is possible that for a given triplet of x, no solution will be found. We then simply sample another triplet.
- One might prefer to require $w_i \in [0, 1]$ which would mean no solution for some triplets. We did not restrict the weights here.

- We could use more than three x, but for more complex data types it tends to be easier to form convex averages of smaller numbers of objects, especially when there is some concern that the proposed average might not lie in the data space. For PCO solutions in d dimensions, we would require at least $d+1$ sampled cases.

- This algorithm is inefficient, but robust. Certainly one may devise faster methods for given types of data, but we need a simple approach that will work with a wide range of data types, particularly the less structured functional data considered in an example to follow.

We applied the algorithm, using 100 randomly selected triplets, to find the orientation corresponding to the scores $(\pm 1, 0)$. This would be useful in interpreting the first principal coordinate. The candidate and selected solutions are shown in the second panel of Figure 1. The solutions correspond very closely to the principal components values for $(\pm 1, 0, 0)$. Note that we would not normally use the tangent space at all for the PCO, but we have projected our solution onto this space for comparison to the original data in the other panel. For this particular application, it would be best to visualize this as a dynamically changing hand orientation.

For this data type, PCA is easier to use, but we have shown that the PCO can produce comparable results. Furthermore, the PCO is more flexible in that it would still work for widely dispersed data and allows for the choice of different distance measures.

3.2 Shapes

Shapes described by landmarks in 3D provide another example of nonlinear data where PCO can be useful. We take our example from a shape consisting of 38 markers placed on the faces of 35 normal children in part of a study on the effects of cleft lip on facial motion. The motion of the faces is subsequently recorded, but in this example we consider only the initial position of the face. The technology behind the data collection and the motivation in terms of dental surgery is described in Faraway (2004).

Our first PCO analysis is based on the Riemann distance. We form the solution in two dimensions. The $D \rightarrow S$ mapping may be obtained using the scoring method described in (1). Only the distances of the new data point to 35 shapes are needed. The construction of the $S \rightarrow D$ mapping is based on an extension to the algorithm described for orientations. To implement this algorithm, we need a way to compute a weighted average of shapes. We use the generalised Procrustes algorithm (Gower (1975)) on, in this case, the three shapes and then form the weighted average. Other weighted averaging methods could be used — we merely require that they produce a valid shape.

In the orientation example, the space of orientations having a specified score was only one dimensional, but in this shape example, this potential solution space has about 100 dimensions. Simply generating random members of this space from triplets of shapes is insufficient to find a solution close to the optimum. We modify the algorithm to provide a more directed search. One first needs to compute the centroid before computing shapes corresponding to other scores.

1. Generate an initial candidate solution from a triplet of shapes having the required score. For finding the centroid, we specify a score of $\mathbf{0}$. We measure the fitness of the solution by computing the mean squared distance from the data in the case of the centroid or the distance from the centroid otherwise.

2. Generate another potential solution from another randomly selected triplet. Find the best weighted average of this triplet and the current best solution. Weights are not restricted to $[0,1]$ to allow moving away from the direction of the triplet.

3. Repeat the second step until a satisfactory result is obtained.

Some experimentation is necessary to tune the best combination of step lengths and convergence criteria. Although it is possible to devise much more efficient optimization methods in this particular example, we wish to restrict ourselves to simpler methods that can be used for data objects have less structure than shapes. Feasibility is our concern now rather than efficiency of computation. It is important that we use only weighted averages of small numbers of objects as this increases confidence that these averages remain within the valid data space. This is not a concern for shapes, but will be a concern in the example to follow.

We applied these methods to the example data and found a centroid as depicted in the first panel of Figure 3. The faces corresponding to scores of $(\pm 1, 0)$ are shown in the second panel of Figure 3. We see that this direction represents the contrast between longer, thinner faces and shorter, wider faces.

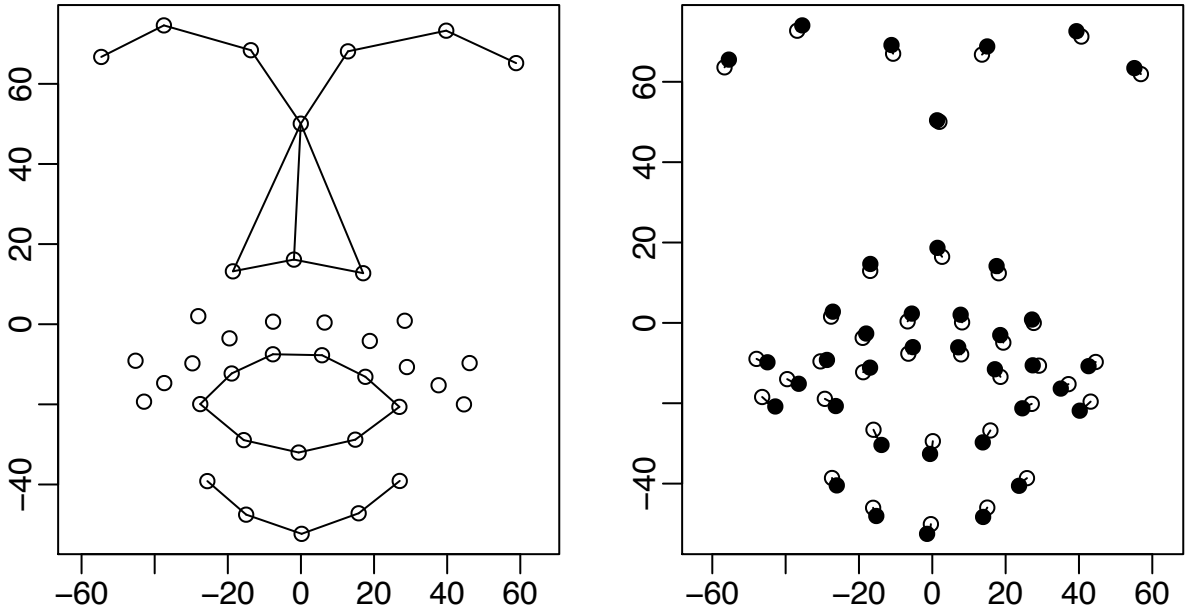

Figure 3: The centroid of the faces as viewed from the front is shown on the left. The lines joining points describe the eyebrows, nose, lip and chin line. The scale is in mm. On the right, faces with a score of $(\pm 1, 0)$ are shown.

The tangent space approach to PCA can also be applied here and will work well provided the shapes are not too widely dispersed. These methods are described in texts such as Dryden and Mardia (1998). The Riemann shape distance is the measure consistent with this approach. The PCA is computed using the `shapes` package of Dryden (2009). The results of this analysis are very similar to that obtained in the PCO. The PCA is much easier to compute in this instance, but this does confirm that the PCO implementation provides sensible solutions.

PCO is more flexible. We can change the distance measure used and the PCO is still available whereas an equivalent PCA would be hard to devise. For example, in Lele and Richtsmeier (2000), an alternative methodology for shape analysis is presented based on the interlandmark distances. Suppose we compute a distance matrix E for each observed shape such that $E_{ij} = d(x_i, x_j)$ for $i, j = 1, \ldots n$ where x_i are the coordinates of landmark i. We might scale the shape for the size of the face by computing $F = E / \sum_{i,j} E_{ij}$. Now given F^1 and F^2 , we might define the distance between the corresponding shapes as:

$$\sum_{i \neq j} (\log F^1_{ij} - \log F^2_{ij})^2$$

The PCO using this distance resulted in substantially different proportions of the variance explained as derived from the eigenvalues. The plot showing the interpretation of the first principal component is shown in Figure 4. We see that the interpretation of this component is fairly similar to the Riemann distance again resulting in a short/wide vs. long/thin contrast.

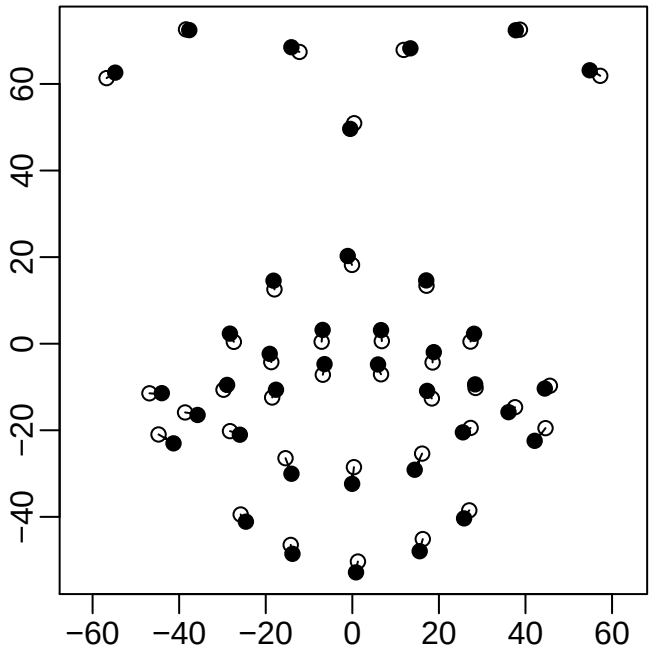

Figure 4: Faces with a score of $(\pm 1, 0)$ are shown based on the PCO using the distance proposed by Lele.

Our point is not to advocate this particular distance, but to demonstrate that a PCO analysis is more flexible than a PCA yet can produce the same useful outputs for subsequent analysis.

3.3 Functions

Data in the form of curves or functions is becoming more commonly available. The data for this example derives from the same facial motion study described above. We consider 29 normal children used in the study. We consider the percentage relative change in distance between the commissures (corners of the lips) while performing a kissing motion. If the distance between commissures at time t is $d(t)$ then our outcome measure is $r(t) = 100(d(t)/d_0 - 1)$ where d_0 is the distance at rest computed using the average of the first and last observation. The motion is recorded at the rate of 60Hz for 4 seconds resulting in 240 equally spaced observations per curve.

The data is shown in Figure 5, where we also show the pointwise mean. We can see there is some variation in the start, duration and magnitude of the kiss for these subjects. The pointwise mean is clearly an unsatisfactory measure of the centre of this data as it is quite unlike any observed curve.

There is a growing body of research on PCA for functional data starting with Besse and Ramsay (1986). A common solution to the horizontal variation is to align or register the curves first and then apply PCA as can be seen in Ramsay and Li (1998) and Wang and Gasser (1997). This approach essentially removes the phase variation and focuses the attention on the amplitude variation. However, we might be simultaneously interested in both kinds of variation. In Izem and Marron (2007), a partially parametric

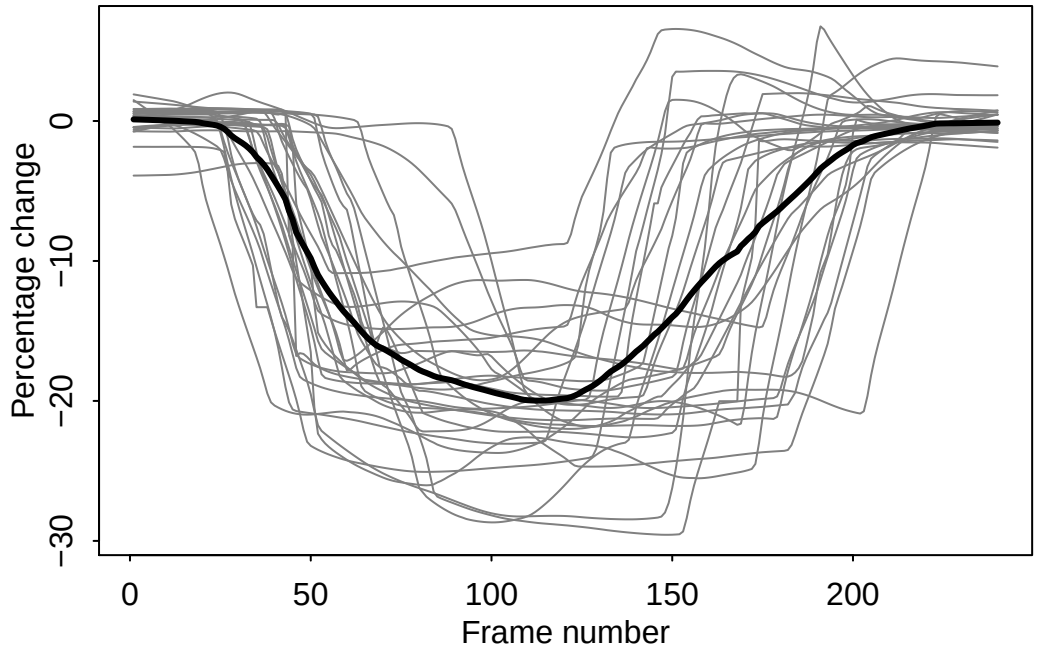

Figure 5: 29 curves showing percentage contraction during kissing. The thicker line is the pointwise mean.

approach is used tackle this problem, but we shall pursue a more fully nonparametric method here. We demonstrate here how a PCO allows us to address this problem while still providing PCA-like outputs.

Dynamic time warping was introduced in signal processing and speech recognition work as a way to align the same words spoken by different individuals — see Itakura (1975) and Sakoe and Chiba (1978). The method has also been used in functional data analysis for registration purposes as in Wang and Gasser (1997) for example. We shall use it for a distance measure. Consider two functions f and g observed at the same equally spaced m time points and a set of pairs of indices $\pi = ((a_1,b_1),\ldots,(a_K,b_K))$. A pair (a_k,b_k) represents a mapping from f_{a_k} to g_{b_k}. The set π has the constraints:

1. $(a_1,b_1) = (1,1)$ and $(a_K,b_K) = (m,m)$
2. $(a_k - a_{k-1}, b_k - b_{k-1}) \in ((1,0),(0,1),(1,1))$ for $k > 1$

The basic form of the algorithm chooses π to minimize a possibly scaled version of $\sum_{i=1}^{K}(f_{a_k} - g_{b_k})^2$. The solution is found using dynamic programming. There are many variations on the basic form of the algorithm most of which constrain the path π so that it cannot vary too far from a 1-1 mapping. In our case, we apply first differences so that we are matching on the first derivative. We also use the so-called Itakura parallelogram constraint. This is appropriate when the functions are known to match at the endpoints and we wish to limit the distortion close to these endpoints. These modifications worked well for this particular dataset in that they resulted in qualitatively satisfactory matching of curves. The DTW computations are provided by Giorgino (2009). The procedure illustrated for two selected curves in Figure 6 where the grey lines denote the pairs (a_k,b_k).

We define the distance between two functions observed at the same discrete times as

$$d(f,g) = \left\{ \frac{1}{K} \sum_{i=1}^{K} \frac{(f_{a_k} - g_{b_k})^2}{v_y^2} + \frac{(a_k - b_k)^2}{v_x^2} \right\}^{1/2}$$

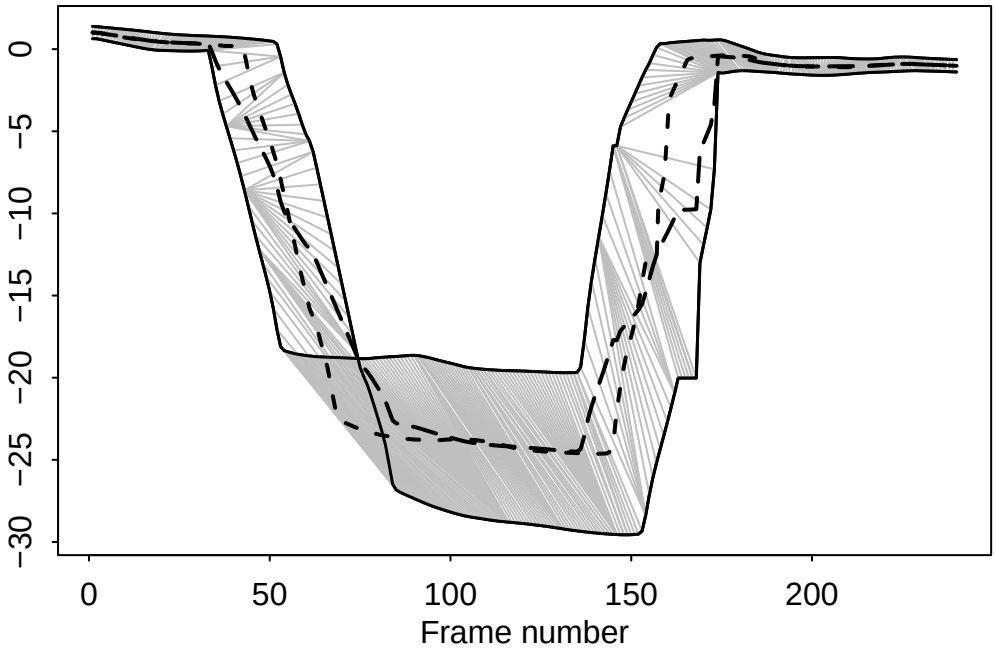

Figure 6: Mean of two curves using DTW mapping. The curves are shown with thin black lines. The DTW mapping is given by the grey lines connecting the two curves. The short dashed line is the DTW mean while the long dashed line is the pointwise mean.

where v_y and v_x are scale factors chosen to place the two components of the sum on a similar scale. They represent the relative weighting of amplitude and phase variation. In this application, we set v_x and v_y equal to the observed ranges. This distance measure is complicated and not convenient for analysis. Small changes in f or g could result in discrete changes in the DTW mapping. Hence there is a lack of continuity that makes difficult the optimisation needed below.

Using this distance a PCO analysis can be performed. The first two dimensions explain about 52% of the variation. We cannot perform a PCA based on this distance because we lack the necessary structure such as an inner product. The $D \to S$ mapping can be achieved using the general method described earlier. All we need are the distances of the new point to the existing points. The $S \to D$ mapping requires more effort. First we need to compute a weighted average of two curves with respect to the distance using the following procedure:

1. Compute the DTW mapping between f and g as $\pi = ((a_1, b_1), \ldots, (a_K, b_K))$
2. Compute, using weight w, $(fg)^w = wf_{a_i} + (1-w)g_{b_i}$ for $i = 1, \ldots, K$
3. Linearly interpolate $(fg)^w$ to produce the weighted mean

The procedure illustrated for two selected curves in Figure 6. The pointwise mean is seen to be inadequate as it has slopes less than either curve and a region of "maximum kiss" which is also less than either curve. The DTW mapping successfully connects the corresponding points in each curve and results in a more credible mean.

The backscoring algorithm requires us to form convex averages of small numbers of cases. The DTW-based weighted mean does not lend itself to more than two curves, but we may sequentially form weighted means of pairs of curves in the group to be averaged in order to construct an overall weighted average. In our example, we shall use scores in two dimensions only so we shall need convex averages

of three curves. We shall take the weighted mean of the first two and then combine the result with the third curve. Due to nonlinearity, the outcome depends on the ordering of the computation. However, in practice, we find little difference. In Liu and Muller (2004), a different method of convex averaging for time warped curves is presented, but our method is simpler for the purposes required in this application.

The backscoring algorithm to produce the curve corresponding to the target score works as for the shapes example except we now make the restriction that $w_i \geq 0$. Although the averaging method described above could accept $w_i < 0$, it might produce atypical curves and so we avoid that here. If for any given triplet, a solution that obeys this constraint is not available, we simply randomly generate another triplet. For a target score far from the origin, it is possible that no solution exists, but this may not be a practical concern.

We calculate the centroid and the curves corresponding to $(1,0),(-1,0),(0,1)$ and $(0,-1)$. The result is shown in Figure 7.

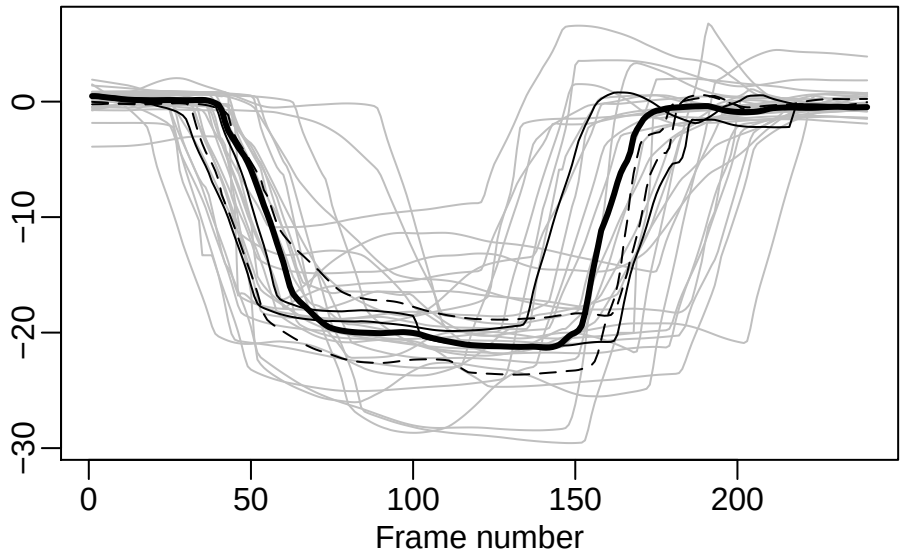

Figure 7: PCO analysis of functional data. The grey lines indicate the data, the thick line is the centroid, the thin solid lines correspond to scores of $(\pm 1,0)$ and the dashed lines correspond to $(0,\pm 1)$.

We see that the direction of the first component corresponds to the length of the kiss while the second component corresponds to the magnitude of the kiss.

3.4 Mixed Data

Data of mixed type, that is some variables quantitative and some categorical often arise in practice, but standard PCA cannot be directly applied. In Hill and Smith (1976) a PCO-like approach to the analysis of this type of data was proposed. The idea is based on yet another equivalent interpretation of PCA. Suppose we find the vector y_1 of length of n that is best correlated with the data in the sense of maximising:

$$\sum_{j=1}^{p} cor^2(x_j, y_1)$$

Once suitably normalised, y_1 is identical to the first coordinate of the scores from a PCA when all x_j are quantitative. The second coordinate y_2 may be found by performing the same maximisation subject to the

constraint that it is uncorrelated with y_1. The other coordinates may be found iteratively in this manner. Now when an x_j is categorical, we replace the squared correlation in the sum with the R^2 from the one-way ANOVA of y on x_j. Notice that this method provides only the scores, but no linear combinations of the variables so it is analogous to PCO and not PCA. We discuss how the method may be supplemented so that it does have functionality equivalent to PCA.

An implementation of this can be found in Dray and Dufour (2007) where a more convenient, but less intuitive method is used resulting in the following mappings. The data to score space mapping is given by

$$S = XWA$$

where X is a data matrix derived from the original data using binary indicators for the categorical variables. An l-level factor is represented within X as an $p \times l$ binary incidence matrix where each row contains a single one in the column corresponding to the level that was observed. The continuous variables appear standardized as columns in X. W is a diagonal matrix of variable weights where continuous variables have weight one while the l entries for a categorical variable sum to one with the entries proportional to the frequency of each level within the factor. A is the weighted principal components of X. It is formed from the eigenvectors of X^TXW normalised so that $A^TWA = I$. S is a matrix of scores. Thus any new data point can be mapped to the score space after a suitable transformation to dummy variables.

For the mapping from score space to the data space, for a given score s we seek an x that maps to the required score subject to the constraint that it is closest to the mean value of the data. The mean is in the transformed dummy space of X and thus needs to be weighted. This amounts to solving:

$$\min x^TWx \qquad \text{s.t.} \qquad s = x^TWA \tag{3}$$

The solution simplifies to $x = As$. Thus the solution is effectively PCA on X with appropriate weighting.

Now the x in the model matrix space needs to be mapped back to the real data space. For quantitative variables, the transformation is direct. For categorical variables the corresponding part of x is a vector of length equal to the number levels in the factor. The vector sums to one and could be regarded as a probability of each level outcome. For interpretation purposes, we can take the most probable level. For simulation, the probability of each level can be used, although some normalisation is required when a computed probability is less than zero.

To illustrate these ideas, we take some environmental data on dune meadows as also used in Dray and Dufour (2007). The data has 20 cases with three quantitative and two categorical variables, one with three levels and the other with four. We apply the Hill–Smith method to find a two-dimensional decomposition. The information is usually displayed in a duality diagram, which does provide some suggestion as to how the coordinate directions should be interpreted. We take an alternate approach here displaying the mapping of the first component having score $(c,0)$ for $c \in [-2,2]$ to the data space as described above. We also show the second component by varying $(0,c)$ in the same manner. Figure 8 shows these components.

We have kept the three quantitative variables on the standardized scale to ease comparison. We see that the first component varies along an axis varying from high values of manure, the "Both" level of "Use" and the "SF" level of "Management" at one end of the scale to low levels of manure, the "Hayfield" level of "Use" and the "NM" level of management. The second component may be interpreted in similar manner (note that the plot for "Use" for each component is almost but not exactly the same). Of course, roughly the same information might be extracted from the duality diagram although perhaps our diagram does allow a more direct interpretation.

A more interesting application of the backscoring algorithm in this instance is in simulating data from the same underlying distribution that gave rise to the observed data. In general one can do this

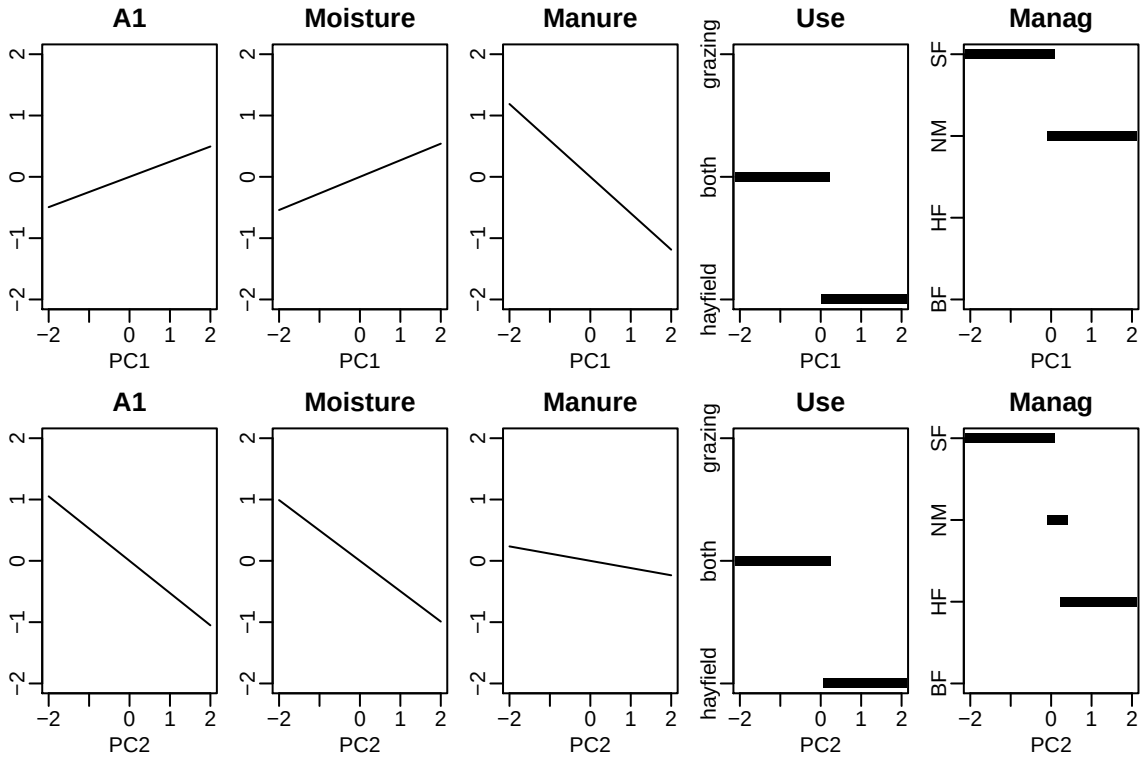

Figure 8: Interpretation of the components for Hill–Smith analysis of some mixed variable type data. The first row of plots shows the values corresponding to a score of $(c,0)$ for $c \in [-2,2]$ and thus provides an interpretation of the first principal component. The second row of plots provides the corresponding interpretation of the second component.

by modelling the data and then simulating from that model, but perhaps we do not want to impose the parametric assumptions necessary for such modelling. In bootstrapping, one simply resamples from the observed cases to make this simulation, but this is somewhat crude for purposes other than bootstrapping. We propose an intermediate alternative here:

1. Simulate score s from $N(0,\Lambda)$ where $\Lambda = diag(\lambda_1, \lambda_2, \ldots)$
2. Find x from (3)
3. For categorical components of x, simulate realised level of the variable using the probabilities derived from x. For quantitative components, rescale and round as necessary.

For the example data, we obtain approximately the same marginal distribution for the data as empirically observed. An exact result is difficult in this case because the three quantitative variables have been severely rounded. In addition, we have approximately the same correlation structure (in the sense of Hill–Smith method as implemented here).

Dray and Dufour (2007) describe several other models involving categorical data including correspondence analysis where the same formulation applies and the above methods could be used.

3.5 Other examples

There are other data types which might benefit from these methods. In Wang and Marron (2007), tree-structured data was described with binary splits combined with lengths for the tree "branches". The data

described the structure of blood vessels. It is difficult to do PCA on such data because of the lack of necessary structure. In the paper, the authors developed a method with PCA-like characteristics, but also rather unlike standard PCA. Defining a distance measure on such tree objects is somewhat easier. PCO could then be applied, which combined with the backscoring algorithm, can lead to a result more closely analogous to PCA.

Diffusion tensors as used in MRI (Pajevic and Basser (2003), Fletcher and Joshi (2004)) do not have a natural vector space, but defining distance measures is a more straightforward task leading again to PCO supplemented by backscoring.

Three of the examples above extract components of human motion data, which is a composite of the motion of head, hands, legs etc. Although PCA is available for some parts of this motion, it is difficult to apply it to the aggregate. In contrast, it is somewhat easier to define distances between motions that can be weighted according to the specific application. We have the choice to experiment with different choices of distance measure. With these distances, PCO is possible along with backscoring to allow a more direct interpretation as well as synthesise new motions. This an example of data of more complex structure for which these methods may be useful.

In Schölkopf et al. (1998), a method known as kernel PCA was introduced. The lower dimensional data is mapped to a much higher dimensional feature space. The PCA is performed in this higher dimensional which would be impractical were it not for the use of the so-called kernel trick which allows the computation of an $n \times n$ matrix on which the PCA is carried out. There are several kernels in common use, one of these being the radial basis function kernel which is a function of the distance matrix Q that we have used above. Although described as a PCA-like method, kernel PCA is more akin to PCO in that the scores are computed directly from the inner products. For PCO, this possibility was noted by Gower (1966). In a very precise sense, PCO is the original kernel method. The link between non-metric MDS and kernel PCA has also been pointed out by Williams (2002). A backscoring approach is possible here also. The map from data to scores follows directly from the method while the algorithms described earlier can provide the backscoring.

Finally we remark that the ISOMAP method of Tenenbaum et al. (2000) involves a projection of high dimensional data onto a nonlinear low dimensional space. The same problems with placing a new point in this smaller space and interpreting the meaning of the projected data in the lower dimensional space arise.

4 Discussion

There are a substantial number of PCO-like methods in so far as they are based on a distance matrix Q. There is some potential to upgrade these to PCA-like functionality by using methods like those described in this paper. In some cases, a data space possesses sufficient structure to directly develop PCA or related methods. Our interest is more in those data spaces which possess only a distance and a smaller amount of structure, such as the ability to perform convex averaging, to enable the backscoring to take place. Thus the value of the methods described here is for data types lacking strong structure or for less malleable distance measures, such as DTW, which make it difficult to develop the structure for a PCA-like method.

The methodology here is presented more as proof of general concept than recommended practical implementation. For any particular type of data and distance, more efficient algorithms for backscoring could be developed and more concrete results obtained. For example, the backscoring for the curve data example took several hours to compute although no great effort was made at code optimization. However, the lack of structure means it is inherently difficult to perform such calculations quickly. For problems with larger n some adjustments will be necessary. For example, in de Silva and Tenenbaum

(2003) Landmark MDS is introduced as a method of dealing with very large distance matrices. At the same time, larger n makes the backscoring problem easier in that there is a higher chance of finding a data points close to the required solution.

The notion of object oriented data analysis is described Wang and Marron (2007). Just as in object oriented programming, where different types of classes may have methods in common, we might consider different data types which require certain common statistical methods. For example, we often require a summary that represents a central observation. The details on how this measure of centre is computed differ from data type to data type — a simple average may suffice for continuous univariate data while something more sophisticated would be needed for, say, shape data. However, we would like this "centre" method to exist for all data types. PCA could be viewed as method for describing variation in data, but this only works for a limited, if commonly used, range of data types. The purpose of this paper, viewed in an OOP light, is to develop the corresponding methods for a wider range of data types.

Acknowledgments

The hand orientation data was obtained from HuMoSim at the University of Michigan. The face data was collected by Dr. Carroll-Ann Trotman of the University of Maryland. The research was supported in part by grant DE13814 from the US National Institute for Dental Research. The author thanks a referee for some constructive advice.

Supplemental Materials

Title: Archive of data and R code reproducing the results shown.

Data Hand orientation, Facial shape and Lip Purse data in R .rda format

Code R code to reproduce the results

References

Anderson, M. and J. Robinson (2003). Generalized discriminant analysis based on distances. *Australian & New Zealand Journal of Statistics 45*(3), 301–318.

Bengio, Y., J. Paiement, P. Vincent, O. Delalleau, N. Le Roux, and M. Ouimet (2004). Out-of-sample extensions for lle, isomap, mds, eigenmaps, and spectral clustering. In *Advances in neural information processing systems 16: proceedings of the 2003 conference*, pp. 177. The MIT Press.

Besse, P. and J. Ramsay (1986). Principal components analysis of sampled functions. *Psychometrika 51*, 285–311.

de Leeuw, J. and P. J. Groenen (1997). Inverse multidimensional scaling. *Journal of Classification 14*, 3–21.

de Silva, V. and J. Tenenbaum (2003). Global versus local methods in nonlinear dimensionality reduction. In S. T. S. Becker and K. Obermayer (Eds.), *Advances in Neural Information Processing Systems 15*, pp. 705–712. Cambridge, MA: MIT Press.

Dray, S. and A. Dufour (2007). The ade4 Package: Implementing the Duality Diagram for Ecologists. *Journal of Statistical Software 22*(4), 6.

Dryden, I. (2009). *shapes: Statistical shape analysis*. R package version 1.1-3.

Dryden, I. and K. Mardia (1998). *Statistical Shape Analysis*. Chichester: Wiley.

Faraway, J. (2004). Modeling continuous shape change for facial animation. *Statistics and Computing 14*, 357–363.

Faraway, J. and S. B. Choe (2009). Modeling orientation trajectories. *Statistical Modelling 9*, 51–68.

Fletcher, P. and S. Joshi (2004). Principal geodesic analysis on symmetric spaces: Statistics of diffusion tensors. *Computer Vision and Mathematical Methods in Medical and Biomedical Image Analysis 3117*, 87–98.

Giorgino, T. (2009). Computing and Visualizing Dynamic Time Warping Alignments in R: The dtw Package. *Journal of Statistical Software 31*(7), 1–24.

Gower, J. (1966). Some distance properties of latent root and vector methods used in multivariate analysis. *Biometrika 53*, 325–338.

Gower, J. (1968). Adding a point to vector diagrams in multivariate analysis. *Biometrika 55*(3), 582–585.

Gower, J. (1975). Generalized procrustes analysis. *Psychometrika 40*(1), 33–51.

Grassia, F. (1998). Practical parameterization of rotations using the exponential map. *The Journal of Graphics Tools 3*, 29–48.

Hill, M. and A. Smith (1976). Principal component analysis of taxonomic data with multi-state discrete characters. *Taxon 25*(2/3), 249–255.

Huckemann, S., T. Hotz, and A. Munk (2010). Intrinsic shape analysis: Geodesic PCA for Riemannian manifolds modulo isometric lie group actions. *Statistica Sinica 20*, 1–30.

Itakura, F. (1975). Minimum prediction residual principle applied to speech recognition. *Acoustics, Speech, and Signal Processing [see also IEEE Transactions on Signal Processing], IEEE Transactions on 23*(1), 67–72.

Izem, R. and J. Marron (2007). Analysis of nonlinear modes of variation for functional data. *Electronic Journal of Statistics 1*, 641–676.

Lele, S. and J. Richtsmeier (2000). *An Invariant Approach to Statistical Analysis of Shapes*. Chapman & Hall.

Liu, X. and H. Muller (2004). Functional convex averaging and synchronization for time-warped random curves. *Journal Of The American Statistical Association 99*, 687–699.

Pajevic, S. and P. Basser (2003). Parametric and non-parametric statistical analysis of DT-MRI data. *Journal Of Magnetic Resonance 161*, 1–14.

Pennec, X. (2006). Intrinsic statistics on Riemannian manifolds: Basic tools for geometric measurements. *Journal of Mathematical Imaging and Visualization 25*, 127–154.

Prentice, M. (1986). Orientation statistics without parametric assumptions. *JRSS-B 48*, 214–222.

Ramsay, J. and X. Li (1998). Curve registration. *Journal of the Royal Statistical Society, Series B 60*, 351–363.

Sakoe, H. and S. Chiba (1978). Dynamic-programming algorithm optimization for spoken word recognition. *IEEE Transactions on acoustics speech and signal processing 26*, 43–49.

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

Tenenbaum, J., V. de Silva, and J. Langford (2000). A global geometric framework for nonlinear dimensionality reduction. *Science 290*, 2319–2323.

Trosset, M. and C. Priebe (2008). The out-of-sample problem for classical multidimensional scaling. *Computational Statistics & Data Analysis 52*(10), 4635–4642.

Trosset, M. and M. Tang (2010, November). The out-of-sample problem for classical multidimensional scaling: Addendum. Technical Report 10-03, Department of Statistics, Indiana University.

Wang, H. and J. Marron (2007). Object Oriented Data Analysis: Sets of Trees. *Annals of Statistics 35*(5), 1849–1873.

Wang, K. and T. Gasser (1997). Alignment of curves by dynamic time warping. *Annals Of Statistics 25*, 1251–1276.

Williams, C. (2002). On a connection between kernel PCA and metric multidimensional scaling. *Machine Learning 46*, 11–19.