CHAPTER 4

Nonlinear Principal Component Analysis and Related Techniques

Jan de Leeuw

CONTENTS

4.1 Introduction 107
4.2 Linear PCA 108
4.3 Least-squares nonlinear PCA 110
4.3.1 Introduction 110
4.3.2 Aspects 113
4.3.3 Algorithm 114
4.3.4 Relation with multiple correspondence analysis 117
4.3.5 Relation with multiple regression 118
4.3.6 Bilinearizability 119
4.3.7 Complete bilinearizability 120
4.3.8 Examples of NLPCA 121
4.4 Logistic NLPCA 127
4.4.1 Perfect fit 129
4.4.2 Geometry of combination rules 129
4.4.3 Example 130
4.5 Discussion and conclusions 132
4.6 Software Notes 132

4.1 Introduction

Principal component analysis (PCA) is a multivariate data analysis technique used for many different purposes and in many different contexts. PCA is the basis for low-rank least-squares approximation of a

data matrix, for finding linear combinations with maximum or minimum variance, for fitting bilinear biplot models, for computing factor-analysis approximations, and for studying regression with errors in variables. It is closely related to simple correspondence analysis (CA) and multiple correspondence analysis (MCA), which are discussed in Chapters 1 and 2 of this book, respectively.

PCA is used wherever large and complicated multivariate data sets have to be reduced to a simpler form. We find PCA in microarray analysis, medical imaging, educational and psychological testing, survey analysis, large-scale time series analysis, atmospheric sciences, high-energy physics, astronomy, and so on. Jolliffe (2002) provides a comprehensive overview of the theory and applications of classical PCA.

4.2 Linear PCA

Suppose we have measurement of n objects or individuals on m variables, collected in an $n \times m$ matrix $\mathbf{X} = \{x_{ij}\}$. We want to have an approximate representation of this matrix in p-dimensional Euclidean space. There are many seemingly different, but mathematically equivalent, ways to define PCA. We shall not dwell on each and every one of them, but we consider the one most relevant for the nonlinear generalizations of PCA we want to discuss.

Our definition of PCA is based on approximating the elements of the data matrix $\mathbf{X}$ by the inner products of vectors in p-dimensional R^p. We want to find n vectors $\mathbf{a}_i$ corresponding with the objects and m vectors $\mathbf{b}_j$ corresponding with the variables such that $x_{ij} \approx \mathbf{a}_i^\top \mathbf{b}_j$. The elements of the $n \times p$ matrix $\mathbf{A}$ are called *component scores*, while those of the $m \times p$ matrix $\mathbf{B}$ are *component loadings*.

We measure degree of approximation by using the least-squares loss function

$$\sigma(\mathbf{A},\mathbf{B}) = \sum_{i=1}^{n}\sum_{j=1}^{m}(x_{ij} - \mathbf{a}_i^\top \mathbf{b}_j)^2 \tag{4.1}$$

PCA is defined as finding the scores $\mathbf{A}$ and the loadings $\mathbf{B}$ that minimize this loss function. Another way of formulating the same problem is that we want to find p new unobserved variables, collected in the columns of $\mathbf{A}$, such that the observed variables can be approximated well by linear combinations of these unobserved variables.

As originally shown by Householder and Young (1938), the solution to this problem can be found by first computing the singular-value

decomposition (SVD) $\mathbf{X} = \mathbf{K}\Lambda\mathbf{L}^T$, where Λ is a diagonal matrix and $\mathbf{K}^T\mathbf{K} = \mathbf{L}^T\mathbf{L} = \mathbf{I}$. The general solution can be established by truncating the SVD by keeping only the largest p singular values Λ_p and corresponding singular vectors $\mathbf{K}_p$ and $\mathbf{L}_p$, and then setting $\hat{\mathbf{A}} = \mathbf{K}_p\Lambda_p^{1/2}\mathbf{S}$ and $\hat{\mathbf{B}} = \mathbf{L}_p\Lambda_p^{1/2}\mathbf{T}$, where $\mathbf{S}$ and $\mathbf{T}$ are any two nonsingular matrices of order p satisfying $\mathbf{S}\mathbf{T}^T = \mathbf{I}$. The minimum value of the loss function is equal to

$$\sigma(\hat{\mathbf{A}}, \hat{\mathbf{B}}) = \sum_{s=p+1}^{m} \lambda_s^2(\mathbf{X}) \tag{4.2}$$

where the $\lambda_s(\mathbf{X})$ are the ordered singular values of $\mathbf{X}$ (so that the λ_s^2 are the ordered eigenvalues of both $\mathbf{X}^T\mathbf{X}$ and $\mathbf{X}\mathbf{X}^T$).

We illustrate this with an example, similar to the box problem in Thurstone (1947: 140). We use 20 rectangles and describe them in terms of seven variables (base, height, diagonal, area, circumference, ratio of base to height, and ratio of height to base). The data matrix, in which base and height are uncorrelated, is given in Table 4.1. The PCA model

Table 4.1 Rectangles.

Base	Height	Diag.	Area	Circumf.	Base/Height	Height/Base
1	1	1.41	1	4	1.00	1.00
2	2	2.82	4	8	1.00	1.00
3	3	4.24	9	12	1.00	1.00
4	4	5.66	16	16	1.00	1.00
5	5	7.07	25	20	1.00	1.00
6	6	8.49	36	24	1.00	1.00
7	7	9.90	49	28	1.00	1.00
8	8	11.31	64	32	1.00	1.00
9	9	12.73	81	36	1.00	1.00
10	10	14.14	100	40	1.00	1.00
11	10	14.87	110	42	1.10	0.91
12	9	15.00	108	42	1.33	0.75
13	8	15.26	104	42	1.63	0.62
14	7	15.65	98	42	2.00	0.50
15	6	16.16	90	42	2.50	0.40
16	5	16.76	80	42	3.20	0.31
17	4	17.46	68	42	4.25	0.23
18	3	18.24	54	42	6.00	0.17
19	2	19.10	38	42	9.50	0.11
20	1	20.02	20	42	20.00	0.05

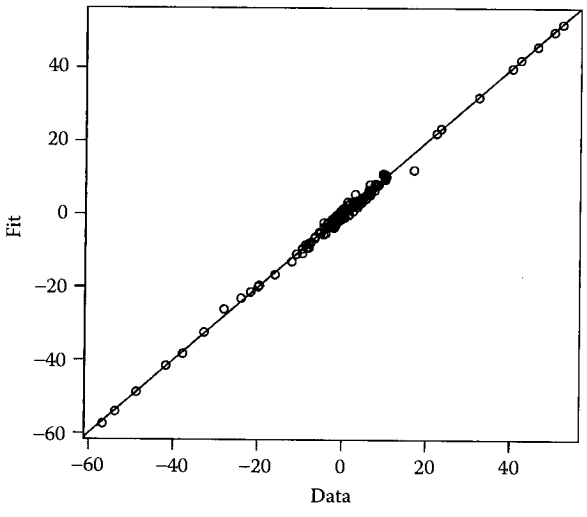

Figure 4.1 PCA fit for rectangles.

fits excellently in two dimensions (99.6% of the sum of squares is "explained"). A plot of the data and the fitted values is in Figure 4.1.

The representation in Figure 4.2 nicely reproduces the V shape of the base–height plot. In Figure 4.2 we have followed the biplot conventions from Gower and Hand (1996), in which loadings are plotted as directions on which we can project the scores. We see, for example, that the last ten rectangles have the same projection on the circumference direction, and that the base/height and height/base directions are very similar because these two variables have a high negative correlation of –0.74.

4.3 Least-squares nonlinear PCA

4.3.1 Introduction

When we talk about nonlinear PCA (NLPCA) in this chapter, we have a specific form of nonlinearity in mind. PCA is a *linear* technique, in the sense that observed variables are approximated by linear combinations of principal components. It can also be a *bilinear* technique, in the sense that elements of the data matrix are approximated by inner products, which are bilinear functions of component scores and component loadings. The nonlinearities in the forms of PCA that we discuss are introduced as nonlinear transformations of the variables, and we still preserve the basic (bi)linearity of the technique. We do not discuss

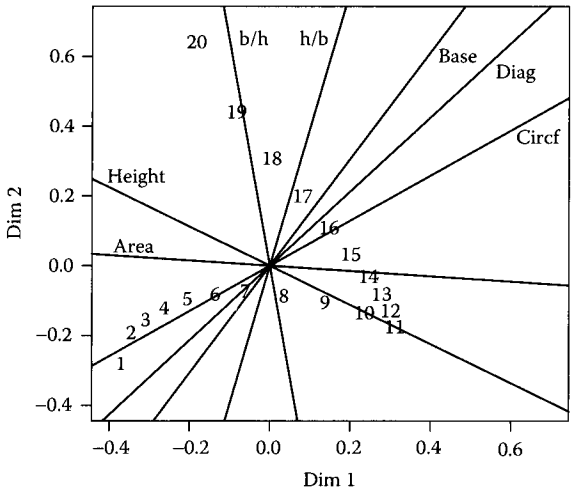

Figure 4.2 PCA solution for rectangles.

more-complicated techniques in which the observed variables are approximated by nonlinear functions of the principal components.

NLPCA can be used, for instance, if we do not have actual numerical values as our data but each variable merely ranks the objects. In other examples, similar to MCA, variables are categorical and partition the objects into a finite number of sets or categories. Binary variables (true/false, yes/no, agree/disagree, and so on) are a very common special case of both ordinal and categorical variables. And in yet other examples, variables may have numerical values, but we want to allow for the possibility of computing transformations to improve the fit of the bilinear model.

Observe that multivariate data matrices in most cases have a property called column conditionality. This means that it makes sense to compare observations within a single column, or variable, but it does not make sense to compare objects from different columns. Each variable orders or measures or classifies the objects into ranges of values specific to the variable, and those ranges may not be comparable. For preference rankings, for instance, the individuals in the experiment order the stimuli, and comparisons are only possible within an individual. This means that, for preference rankings, the individuals are actually the variables ranking the objects. This concept of conditionality is closely related to the classical psychometric distinction between Q and R techniques (Stephenson 1953).

We have seen in the previous section that we evaluate fit of PCA in p dimensions by computing the sum of squares of the residual singular values of $\mathbf{X}$ (or the sum of the residual eigenvectors of the product moment matrix $\mathbf{X}^\top\mathbf{X}$). This makes it natural to look for transformations or quantifications of the variables that minimize the same criterion. Thus, we do not minimize loss merely over component scores $\mathbf{A}$ and component loadings $\mathbf{B}$, but also over the admissible transformations of the columns of $\mathbf{X}$. The loss function becomes

$$\sigma(\mathbf{A},\mathbf{B},\mathbf{X}) = \sum_{i=1}^{n}\sum_{j=1}^{m}\left(x_{ij} - \mathbf{a}_i^\top\mathbf{b}_j\right)^2 \tag{4.3}$$

and we minimize, in addition, over $\mathbf{x}_j \in X_j$, where $X_j \subseteq \mathcal{R}^n$ are the admissible transformations for variable j. By using Equation 4.2, this is the same as finding

$$\min_{\mathbf{x}_j \in X_j} \sum_{s=p+1}^{m} \lambda_s(\mathbf{X}) \tag{4.4}$$

This form of NLPCA, in the special case of monotone transformations, has been proposed by, among others, Lingoes and Guttman (1967), Kruskal and Shepard (1974), and Roskam (1968).

The notion of admissible transformation needs some additional discussion. We have already mentioned the class of monotone transformations as an important example. But other examples can also be covered. We could, for instance, allow low-order polynomial transformations for all or some of the variables. Or, combining the two ideas, monotone polynomials. We could also look for convex or concave transformations, increasing or not. Or we could look for low-order splines on a given knot sequence, which again may or may not be restricted to be monotone. For categorical variables with a small number of categories we may simply allow the class of all possible transformations, which is also known as the class of *quantifications*, in which category labels are replaced by real numbers, as in MCA. NLPCA has been extended to these wider classes of admissible transformations by Young et al. (1978) and Gifi (1990).

All special cases of transformations mentioned so far are covered by the general restriction that the transformed variable must be in a convex cone $\mathcal{R}$ in $\mathcal{R}^n$. Convex cones are defined by the conditions that $\mathbf{x} \in \mathcal{K}$ implies $\alpha\mathbf{x} \in \mathcal{K}$ for all real $\alpha \geq 0$ and $\mathbf{x} \in \mathcal{K}$ and $\mathbf{y} \in \mathcal{K}$ implies $\mathbf{x} + \mathbf{y} \in \mathcal{K}$. It is easy to see that all classes of transformations discussed

above are indeed convex cones. In fact some of them, such as the low-order polynomials and splines, are linear subspaces, which are special cones for which $\mathbf{x} \in \mathcal{K}$ implies $\alpha\mathbf{x} \in \mathcal{K}$ for all real α.

It is also clear that if a transformation $\mathbf{x}$ is in one of the cones mentioned above, then a positive linear function $\alpha\mathbf{x} + \beta$ with $\alpha \geq 0$ is in the cone as well. As a consequence of this we need to normalize our transformations, both to identify them and to prevent the trivial solution in which all transformations are identically set to zero. Another way of saying this is that we redefine our cones to consist only of centered vectors, and we want all transformations $\mathbf{x}$ to be on the unit sphere. Thus, the sets of admissible transformations X_j are of the form $\mathcal{K}_j \cap S$, where $\mathcal{K}_j$ is a convex cone of centered vectors.

The use of normalizations implies that the product moment matrix $\mathbf{X}'\mathbf{X}$ is actually the *correlation matrix* of the variables. Thus, the optimization problem for NLPCA in p dimensions is to find admissible transformations of the variables in such a way that the sum of the $n - p$ smallest eigenvalues of the correlation matrix is minimized or, equivalently, such that the sum of the p largest eigenvalues is maximized. We write our NLPCA problem in the final form as

$$\max_{x_j \in \mathcal{K}_j \cap S} \sum_{s=1}^{p} \lambda_s(\mathbf{R}(\mathbf{X})) \tag{4.5}$$

where $\mathbf{R}(\mathbf{X})$ is the correlation matrix of the transformed variables in $\mathbf{X}$. This seems a natural and straightforward way to generalize PCA. Allowing for nonlinear transformations of the variables makes it possible to concentrate more variation in the first few principal components. Instead of looking at high-dimensional projections, we can look at low-dimensional projections together with plots of the nonlinear transformations that we compute (de Leeuw and Meulman 1986).

4.3.2 Aspects

Instead of tackling the optimization problem (Equation 4.5) directly, as is done in most earlier publications, we embed it in a much larger family of problems for which we construct a general algorithm. Let us look at Equation 4.5 in which we maximize any convex function ϕ of the correlation matrix $\mathbf{R}(\mathbf{X})$—not just the sum of the p largest eigenvalues. We call any convex real-valued function defined on the space of correlation matrices an *aspect* of the correlation matrix (de Leeuw 1988, 1990).

Of course, we first have to show that, indeed, the sum of the p largest eigenvalues is a convex function of the correlation matrix. For this we use the very useful lemma that if $f(\mathbf{x},\mathbf{y})$ is convex in $\mathbf{x}$ for every $\mathbf{y}$, then $g(\mathbf{x}) = \max_{\mathbf{y}} f(\mathbf{x},\mathbf{y})$ is also convex in $\mathbf{x}$. The sum of the p largest eigenvalues of a matrix $\mathbf{R}$ is the maximum of tr $\mathbf{L}'\mathbf{R}\mathbf{L}$ over all $m \times p$ matrices $\mathbf{L}$ with $\mathbf{L}'\mathbf{L} = \mathbf{I}$. Thus, the aspect is the pointwise maximum of a family of functions that are linear, and thus convex, in $\mathbf{R}$, and the lemma applies.

We take the opportunity to give some additional examples of convex aspects that illustrate the considerable generality of our approach. A very simple aspect is the sum of the correlation coefficients. It does not use eigenvalues to measure how closely variables are related, but it does measure the strength of the overall relationships. Related aspects are the sum of even powers of the correlation coefficients, or the sum of odd powers of the absolute values of the correlation coefficients. Observe that the sum of squares of the correlation coefficients is actually equal to the sum of squares of the eigenvalues of the correlation matrix. Because the sum of the eigenvalues is a constant, maximizing the sum of squares is the same as maximizing the variance of the eigenvalues. This aspect gives another way to concentrate as much of the variation as possible in the first few principal components.

4.3.3 Algorithm

The algorithm we propose is based on the general *principle of majorization*. Majorization methods are discussed extensively in de Leeuw (1994), Heiser (1995), Lange et al. (2000), and de Leeuw and Michailidis (in press). We give only a very brief introduction.

In a majorization algorithm the goal is to minimize a general real-valued function $g(\mathbf{x})$ over $\mathbf{x} \in X$, with. Of course, maximization of g is the same as minimization of $-g$, so the introduction below also applies to maximization problems.

Majorization requires us to construct a function $f(\mathbf{x},\mathbf{y})$, defined on $X \times X$, that satisfies

$$f(\mathbf{x},\mathbf{y}) \geq g(\mathbf{x}) \quad \text{for all } \mathbf{x},\mathbf{y} \in X \tag{4.6a}$$

$$f(\mathbf{x},\mathbf{x}) = g(\mathbf{x}) \quad \text{for all } \mathbf{x} \in X \tag{4.6b}$$

Thus, for a fixed $\mathbf{y}$, $f(\mathbf{x},\mathbf{y})$ is above $g(\mathbf{x})$, and it touches $g(\mathbf{x})$ at the point $(\mathbf{y},g(\mathbf{y}))$. We say that f majorizes g.

Majorizing functions are used to construct the following iterative algorithm for minimizing $g(\mathbf{x})$. Suppose we are at step k.

Step 1 Given a value $\mathbf{x}^{(k)}$, construct a majorizing function $f(\mathbf{x},\mathbf{x}^{(k)})$.

Step 2 Set $\mathbf{x}^{(k+1)} = \underset{\mathbf{x}\in X}{\operatorname{argmin}}\, f(\mathbf{x},\mathbf{x}^{(k)})$.

Step 3 If $\| \mathbf{x}^{(k+1)} - \mathbf{x}^{(k)} \| < \varepsilon$ for some predetermined $\varepsilon > 0$, stop; else go to Step 1.

Now $g(\mathbf{x}^{(k+1)}) \leq f(\mathbf{x}^{(k+1)},\mathbf{x}^{(k)})$ because of majorization, and $f(\mathbf{x}^{(k+1)},\mathbf{x}^{(k)}) < f(\mathbf{x}^{(k)},\mathbf{x}^{(k)})$ because $\mathbf{x}^{(k+1)}$ minimizes the majorization function. But $f(\mathbf{x}^{(k)},\mathbf{x}^{(k)}) = g(\mathbf{x}^{(k)})$ because the majorization function touches at the current point. Thus, we have the *sandwich inequality*, which says

$$g(\mathbf{x}^{(k+1)}) \leq f(\mathbf{x}^{(k+1)},\mathbf{x}^{(k)}) < f(\mathbf{x}^{(k)},\mathbf{x}^{(k)}) = g(\mathbf{x}^{(k)})$$

and a majorization step consequently always decreases the loss function. For this algorithm to be of practical use, the majorizing function f needs to be easy to maximize, otherwise nothing substantial is gained by following this route.

We now apply majorization theory to maximizing our convex aspect, ϕ. Because we are maximizing, we need to find a minorization function. The convexity of the aspect, and the fact that a convex function is always above its tangents, gives the inequality

$$\phi(\mathbf{R}(\mathbf{X})) \geq \phi(\mathbf{R}(\mathbf{Y})) + \sum_{1\leq i\neq j\leq n}\sum \left.\frac{\partial\phi}{\partial r_{ij}}\right|_{\mathbf{R}=\mathbf{R}(\mathbf{Y})} \left(\mathbf{x}_i^\top \mathbf{x}_j - \mathbf{y}_i^\top \mathbf{y}_j\right) \tag{4.7}$$

for all matrices $\mathbf{X}$ and $\mathbf{Y}$ of normalized admissible transformations. The normalization ensures that the diagonal terms in the double sum on the right disappear.

Each step in the majorization algorithm requires us to maximize the right-hand side of Equation 4.7. We do this by *block relaxation*, that is, by maximizing over one transformation at a time, keeping the other transformations fixed at their current values (de Leeuw 1994). Thus in each iteration we solve m of these optimal scaling problems, transforming or quantifying each of the variables in turn.

By separating out the part of Equation 4.7 that depends only on $\mathbf{x}_j$, we find that each optimal scaling problem amounts to solving a least-squares problem of the form

$$\min_{\mathbf{x}_j\in\mathcal{K}_j\cap S} \left(\mathbf{x}_j - \tilde{\mathbf{x}}_j^{(k)}\right)^\top \left(\mathbf{x}_j - \tilde{\mathbf{x}}_j^{(k)}\right) \tag{4.8}$$

Here $\tilde{\mathbf{x}}_j^{(k)}$ is the current *target*, defined by

$$\tilde{\mathbf{x}}_j^{(k)} = \sum_{\ell<j} g_{j\ell}^{(k,j)} \mathbf{x}_\ell^{(k+1)} + \sum_{\ell<j} g_{j\ell}^{(k,j)} \mathbf{x}_\ell^{(k)}$$

and the matrices $\mathbf{H}^{(kj)}$ are the partial derivatives, evaluated while updating variable j in iteration k. Thus,

$$h_{j\ell}^{(k,j)} = \left.\frac{\partial \phi}{\partial r_{j\ell}}\right|_{\mathbf{R}=\mathbf{R}(\mathbf{x}_1^{(k+1)},\cdots,\mathbf{x}_{j-1}^{(k+1)},\mathbf{x}_j^{(k)}\cdots,\mathbf{x}_m^{(k)})}$$

The formula looks complicated, but the only thing it does is keep track of the iteration indices. If we have an expression for the partial derivatives and a way to solve the least-squares problem in Equation 4.8, then we have a simple and general way to maximize the corresponding aspect. From the software point of view, we can write a high-level algorithm that uses as arguments subroutines to compute aspects and their partial derivatives. Thus, with relatively little extra work, users can plug in their own aspects.

If the aspect we use is the sum of the correlation coefficients, then all elements of $\mathbf{H}^{(kj)}$ are equal to +1, and thus, the target is just the sum of all variables (except for the one we are updating). If the aspect is a single correlation coefficient in the matrix, say $\mathbf{r}_{j\ell}$, then the target when updating $\mathbf{x}_j$ will be $\mathbf{x}_\ell$ and vice versa. In the general case, we have to recompute the correlations and the partials after updating each variable. This can be expensive computationally. If our aspect is the classical NLPCA sum of the p largest eigenvalues, for instance, then

$$\frac{\partial \phi}{\partial \mathbf{R}} = \mathbf{L}\mathbf{L}^\top$$

with $\mathbf{L}$ the normalized eigenvectors corresponding with the p largest eigenvalues of $\mathbf{R}$. Computing the partials means solving an eigenvalue problem. De Leeuw (1990) discusses some (minor) variations of the algorithm that allow for updating all variables before recomputing the correlations and the partial derivatives.

It is also shown in de Leeuw (1990) that Equation 4.8 can be minimized by first projecting on the cone, thus ignoring the normalization constraint, and then normalizing afterward. Generally, such cone projection problems are simple to solve. In the categorical case, for instance, we merely have to compute category averages. In the

monotone case, we must perform a monotone regression to project the target on the cone (de Leeuw 2005). In the polynomial case, we must solve a polynomial regression problem.

4.3.4 Relation with multiple correspondence analysis

MCA is a special case of our general aspect approach. It corresponds with maximizing the largest eigenvalue of the correlation matrix (and with the case in which all variables are categorical). As shown in Chapter 2, MCA solves the generalized eigenproblem for the Burt matrix. This corresponds with finding the stationary values of the ratio

$$\lambda(\mathbf{a}) = \frac{\sum_{j=1}^{m}\sum_{\ell=1}^{m}\mathbf{a}_j'\mathbf{C}_{j\ell}\mathbf{a}_\ell}{m\sum_{j=1}^{m}\mathbf{a}_j'\mathbf{C}_{jj}\mathbf{a}_j}$$

Change variables by letting $\mathbf{a}_j = v_j\mathbf{y}_j$, where $\mathbf{y}_j'\,\mathbf{C}_{jj}\mathbf{y}_j = 1$. Then

$$\lambda(\mathbf{v},\mathbf{y}) = \frac{\mathbf{v}'\mathbf{R}(\mathbf{y})\mathbf{v}}{m\mathbf{v}'\mathbf{v}}$$

where $\mathbf{R}(\mathbf{y})$ is the correlation matrix induced by the quantifications in $\mathbf{a}$. It follows that

$$\max_{\mathbf{y}}\max_{\mathbf{v}}\lambda(\mathbf{v},\mathbf{y}) = \max_{\mathbf{y}}\lambda_{max}(\mathbf{R}(\mathbf{y}))$$

which is what we wanted to show.

Thus, the dominant MCA solution gives us the quantifications maximizing the largest eigenvalue aspect. And the largest eigenvalue of the induced correlation matrix is the largest eigenvalue of the MCA problem. But what about the remaining MCA solutions? They provide additional solutions of the stationary equations for maximizing the largest eigenvalue aspect, corresponding with other nonglobal minima, local maxima, and saddle points. As was pointed out very early on by Guttman (1941) the first MCA solution should be distinguished clearly from the others, because the others correspond with suboptimal solutions of the stationary equations. In fact, each MCA eigenvector has its own associated induced correlation matrix. And each MCA eigenvalue is an eigenvalue (and not necessarily the largest one) of the correlation matrix induced by the corresponding MCA eigenvector.

It goes without saying that simple CA is the special case in which we have only two variables, and both are categorical. The correlation matrix has only one nonconstant element, and all reasonable aspects will be monotone functions of that single correlation coefficient. Maximizing the aspect will give us the maximum correlation coefficient, and the CA solutions will be the transformations solving the stationary equations of the maximum correlation problem.

4.3.5 Relation with multiple regression

Multiple regression and PCA are quite different techniques, but nevertheless there are some important relationships. Consider the PCA problem of maximizing the sum of the $m - 1$ largest eigenvalues of the correlation matrix. This is the same, of course, as minimizing the smallest eigenvalue, and thus, it can be interpreted as looking for a singularity in the transformed data matrix. This form of regression analysis dates back to Pearson (1901). It is a form of regression analysis, except that in the usual regression analysis we single out one variable as the outcome and define the rest as the predictors, and we measure singularity by finding out whether and how far the outcome variable is in the space spanned by the predictors.

More precisely, the squared multiple correlation coefficient of variable j with the remaining $m - 1$ variables can be written as

$$\phi(\mathbf{R}(\mathbf{X})) = \max_{\mathbf{b}}(1 - \mathbf{b}^\top \mathbf{R}\mathbf{b})$$

where the vector $\mathbf{b}$ is restricted to have $b_j = 1$. By the lemma we used previously, this is a convex function of $\mathbf{R}$, which can be maximized by our majorization algorithm. The partials are simply

$$\frac{\partial \phi}{\partial \mathbf{R}} = -\mathbf{b}\mathbf{b}^\top$$

This aspect can be easily extended to the sum of all m squared multiple correlation coefficients of each variable with all others, which has been discussed in the context of factor analysis by Guttman (1953) and others.

So far, we have written down our theory for the case in which we are maximizing a convex aspect. As we noted previously, the same results apply for minimizing a concave aspect. Some aspects are more

naturally discussed in this form. Consider, for example, the determinant of the correlation matrix. Minimizing the determinant can also be thought of as looking for a singularity, i.e., as yet another way of approaching regression. The representation

$$\log \| \mathbf{R} \| = \min_{\Gamma \gtrsim 0} \log \| \mathbf{\Gamma} \| + \text{tr} \mathbf{\Gamma}^{-1} \mathbf{R} - m,$$

where $\mathbf{\Gamma} \geq 0$ means we require $\mathbf{\Gamma}$ to be positive semidefinite, shows that the logarithm of the determinant is a concave function of the correlation matrix. Also

$$\frac{\partial \phi}{\partial \mathbf{R}} = \mathbf{R}^{-1}$$

which means that the target for updating a variable is its *image*, in the sense of Guttman (1953), the least-squares prediction of the variable from all others. Minimizing the determinant can be done by sequentially projecting images on cones of admissible transformations.

4.3.6 Bilinearizability

There is more that can be said about the relationship between MCA and maximizing the correlation aspects of NLPCA. Most of the theory we discuss here is taken from de Leeuw (1982), Bekker and de Leeuw (1988), de Leeuw (1988), and de Leeuw et al. (1999).

Let us start by looking at the condition of *bilinearizability* of regressions. This means that we can find transformation of the variables (in our class of admissible transformations) such that all bivariate regressions are exactly linear. In the case of m categorical variables with Burt table $\mathbf{C}$, this means that the system of bilinearizability equations

$$\mathbf{C}_{j\ell} \mathbf{y}_\ell = r_{j\ell} \mathbf{C}_{jj} \mathbf{y}_j \tag{4.9}$$

has a solution, normalized by $\mathbf{y}_j' \mathbf{C}_{jj} \mathbf{y}_j = 1$ for all j. The corresponding induced correlation matrix $\mathbf{R}(\mathbf{y})$ has m eigenvalues λ_s and m corresponding normalized eigenvectors $\mathbf{v}_s$. We can now define the m vectors $\mathbf{a}_{js} = v_{js} \mathbf{y}_j$, and we find

$$\sum_{l=1}^m \mathbf{C}_{j\ell} \mathbf{a}_{\ell s} = \sum_{l=1}^m \mathbf{C}_{j\ell} \mathbf{y}_\ell v_{\ell s} = \mathbf{C}_{jj} \mathbf{y}_j \sum_{l=1}^m r_{j\ell} v_{\ell s} = \lambda_s v_{js} \mathbf{C}_{jj} \mathbf{y}_j = \lambda_s \mathbf{C}_{jj} \mathbf{a}_{js}$$

In other words, for each s the vector $\mathbf{a}_s$ defines a solution to the MCA problem, with eigenvalue λ_s, and each of these m solutions induces the same correlation matrix.

Bilinearizability has some other important consequences. A system of transformations that linearizes all regressions solves the stationary equations for any aspect of the correlation matrix. Thus, in a multivariate data matrix with bilinearizability, it does not matter which aspect we choose, because they will all give the same transformations. Another important consequence of bilinearizability is that the correlation coefficients computed by maximizing an aspect have the same standard errors as the correlation coefficients computed from known scores. This means, for example, that we can apply the asymptotically distribution-free methods of structural equation programs to optimized correlation matrices, and they will still compute the correct tests and standard errors if the data are bilinearizable (or a sample from a bilinearizable distribution).

4.3.7 Complete bilinearizability

It may be the case that there is a second set of transformations $\bar{\mathbf{y}}_j$ that satisfies Equation 4.9. Again, such a set generates m additional MCA solutions, all inducing the same correlation matrix. Moreover, $\mathbf{y}_j^\top \mathbf{C}_{jj} \bar{\mathbf{y}}_j = 0$ for all j, so the second set is orthogonal to the first for each variable separately. And there may even be more sets. If bilinearizability continues to apply, we can build up all MCA solutions from the solutions to Equation 4.9 and the eigenvectors of the induced correlation matrices. Another way of thinking about this is that we solve $\binom{m}{2}$ simple CA problems for each of the subtables of the Burt matrix. Equation 4.9 then says that if we have complete bilinearizability, we can patch these CA solutions together to form the MCA solution.

More precisely, suppose $\mathbf{C}$ is a Burt matrix and $\mathbf{D}$ is its diagonal. We have complete bilinearizability if there are matrices $\mathbf{K}_j$ such that $\mathbf{K}_j^\top \mathbf{C}_{jj} \mathbf{K}_j = \mathbf{I}$ for each j and $\mathbf{K}_j^\top \mathbf{C}_{j\ell} \mathbf{K}_\ell$ is diagonal for each j and ℓ. Remember that the direct sum of matrices stacks those matrices in the diagonal submatrices of a large matrix, which has all its nondiagonal submatrices equal to zero. If $\mathbf{K}$ is the direct sum of the $\mathbf{K}_j$, then $\mathbf{K}^\top \mathbf{D} \mathbf{K} = \mathbf{I}$ while $\mathbf{E} = \mathbf{K}^\top \mathbf{C} \mathbf{K}$ has the same structure as the Burt matrix, but all submatrices $\mathbf{E}_{j\ell}$ are now diagonal. This means there is a permutation matrix $\mathbf{P}$ such that $\mathbf{P}^\top \mathbf{K}^\top \mathbf{C} \mathbf{K} \mathbf{P}$ is the direct sum of correlation matrices. The first correlation matrix contains all (1,1) elements of the $\mathbf{E}_{j\ell}$, the second correlation matrix contains all (2,2) elements, and so on.

By making $\mathbf{L}$ the direct sum of the matrices of eigenvectors of these correlation matrices, we see that $\mathbf{L}'\mathbf{P}'\mathbf{K}'\mathbf{CKPL}$ is diagonal, while $\mathbf{L}'\mathbf{P}'\mathbf{K}'\mathbf{DKPL} = \mathbf{I}$. Thus the matrix $\mathbf{KPL}$ contains all the MCA solutions and gives a complete eigendecomposition of the Burt matrix.

This may be somewhat abstract, so let us give an important application. Suppose we perform an MCA of a standard multivariate normal, with correlation matrix Γ. Because all bivariate regressions are linear, the linear transformations of the variables are a bilinearizable system, with correlation matrix Γ. But the quadratic Hermite–Chebyshev polynomials are another bilinearizable system, with correlation matrix $\Gamma^{(2)}$, the squares of the correlation coefficients, and so on. Thus we see that applying MCA to a multivariate normal will give m solutions consisting of polynomials of degree d, where the eigenvalues are those of $\Gamma^{(d)}$, for all $d = 1,2,....$

In standard MCA we usually order the eigenvalues and keep the largest ones, often the two largest ones. The largest eigenvalue for the multivariate normal is always the largest eigenvalue of Γ, but the second largest eigenvalue can be either the second largest eigenvalue of Γ or the largest eigenvalue of $\Gamma^{(2)}$. If the second largest eigenvalue in the MCA is the largest eigenvalue of $\Gamma^{(2)}$, then for each variable the first transformation will be linear and the second will be quadratic, which means we will find horseshoes (Van Rijckevorsel 1987) in our scatter plots. There is an example in Gifi (1990: 382–384) where two-dimensional MCA takes both its transformations from Γ, which means it finds the usual NLPCA solution.

Our analysis shows clearly what the relationships are between MCA and NLPCA. In PCA we find a single set of transformations and a corresponding induced correlation matrix that is optimal in terms of an aspect. In MCA we find multiple transformations, each with its own corresponding induced correlation matrix. Only in the case of complete bilinearizability (such as is obtained in the multivariate normal) can we relate the two solutions because they are basically the same solution. MCA, however, presents the solution in a redundant and confusing manner. This gives a more precise meaning to the warning by Guttman (1941) that the additional dimensions beyond the first one in MCA should be interpreted with caution.

4.3.8 Examples of NLPCA

Our first data set is the GALO (*Groninger Afsluitingsonderzoek Lager Onderwijs*) data, taken from Peschar (1975). The objects (individuals) are 1290 school children in the sixth grade of an elementary school in

the city of Groningen (the Netherlands) in 1959. The four variables and their categories are

1. Gender: (a) boys, (b) girls
2. IQ: values between 60 and 144, categorized into nine subcategories
3. Teacher's advice: (a) no further education, (b) extended primary education, (c) manual-labor education, (d) agricultural education, (e) general education, (f) secondary school for girls, (g) pre-university
4. Father's profession: (a) unskilled labor, (b) skilled labor, (c) lower white collar, (d) shopkeepers, (e) middle white collar, (f) professional

We use these data to maximize a large number of different aspects of the correlation matrix. All variables are categorical, and no monotonicity or smoothness constraints are imposed. Results are in Table 4.2. Each row of the table corresponds with a different aspect that we optimize, and thus with a different correlation matrix. The table gives the four eigenvalues of the induced correlation matrix, and in the final column the induced correlation coefficient between IQ and Advice, which are the two

Table 4.2 GALO example with eigenvalues of correlation matrices induced by maximizing different aspects.

Aspect	λ_1	λ_2	λ_3	λ_4	r_{23}
Sum of correlations	2.147	0.987	0.637	0.229	0.767
Sum of squared correlations	2.149	0.998	0.648	0.204	0.791
Sum of cubed correlations	2.139	0.934	0.730	0.198	0.796
Largest eigenvalue	2.157	0.950	0.682	0.211	0.784
Sum of two largest eigenvalues	1.926	1.340	0.535	0.198	0.795
Sum of three largest eigenvalues	1.991	1.124	0.688	0.196	0.796
Squared multiple correlation with advice	2.056	1.043	0.703	0.196	0.796
Sum of squared multiple correlations	1.961	1.302	0.538	0.199	0.795
Determinant	2.030	1.220	0.551	0.199	0.796

dominant variables in the GALO example. In the fourth row, for example, we find the eigenvalues of the correlation matrix induced by maximizing the largest eigenvalue aspect (which is also the correlation matrix induced by the first MCA dimension). And in the last row we find the eigenvalues of the correlation matrix induced by minimizing the determinant.

The largest possible eigenvalue is 2.157 (from the fourth row) and the smallest possible one is 0.196 (from the sixth row). The regression-type solutions, seeking singularities, tend to give a small value for the smallest eigenvalue. In general, the pattern of eigenvalues is very similar over the different aspects, suggesting approximate bilinearizability. We give the transformations for the aspect that maximizes the largest eigenvalue, that is, for the MCA solution, in Figure 4.3.

We can also use this example to illustrate the difference between MCA and NLPCA. Figure 4.4 has the two principal components from an MCA solution. The components come from different correlation matrices, one corresponding with linear transformations and one corresponding with quadratic ones. Thus the component scores form a horseshoe. The NLPCA solution for the same data is shown in Figure 4.5. Both components come

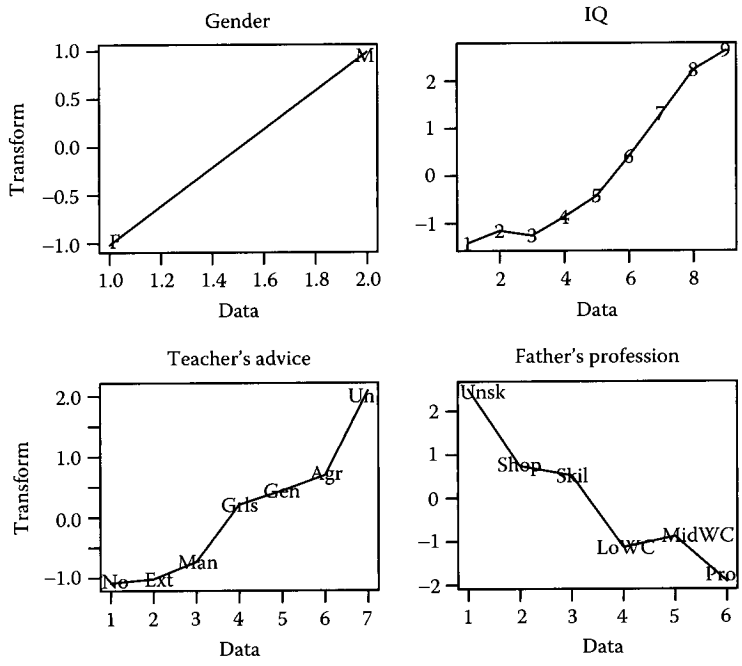

Figure 4.3. GALO data: Transformations.

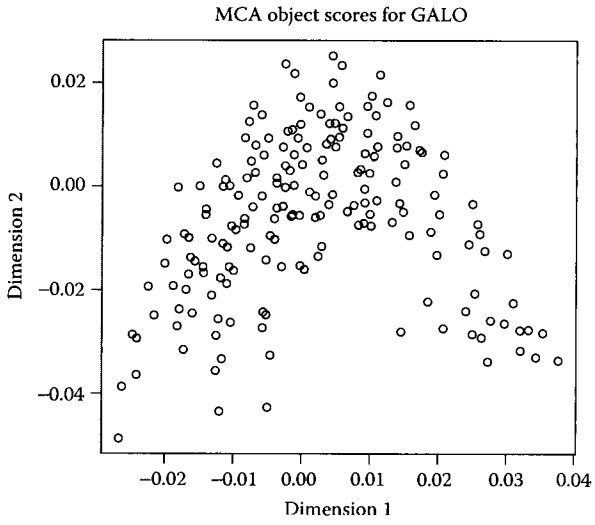

Figure 4.4 GALO data: MCA.

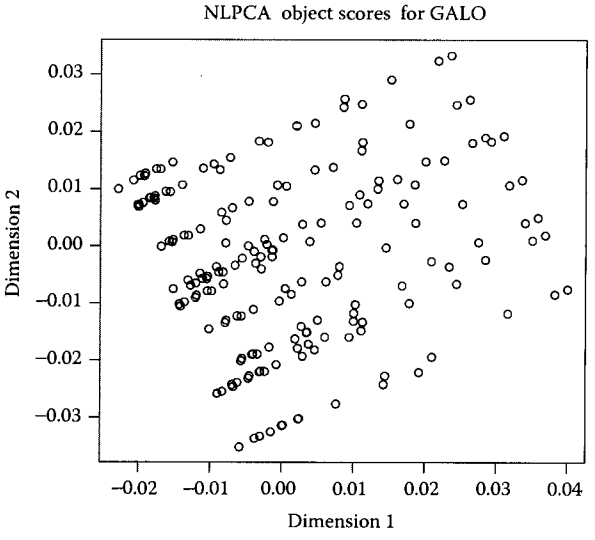

Figure 4.5 GALO data: Nonlinear PCA.

from the correlation matrix induced by the transformations in Figure 4.3. We see a completely different plot, without horseshoe, in which the discrete parallel strips of points come about because the dominant variables IQ and Advice have only a small finite number of values.

The second example of NLPCA is from Roskam (1968: 152). The Department of Psychology at the University of Nijmegen has, or had, nine different areas of research and teaching. Each of the 39 psychologists working in the department ranked all nine areas in order of relevance for their work. The areas are given in Table 4.3, and the data in Table 4.4. These are preference rank orders, and thus conditionality dictates we compute correlation coefficients among the 39 psychologists.

We first perform a linear PCA on the rank numbers, which is sometimes known as Tucker's Preference Analysis (Tucker 1960). The first two eigenvalues of (1/9) **R** are 0.374 and 0.176, which means the first two principal components capture 55% of the variation in the rank numbers. We now optimize the sum of the first two eigenvalues over all monotone transformations of the 39 rank orders. The eigenvalues increase to 0.468 and 0.297, and thus the two principal components capture 76.6% of the transformed rank numbers. For completeness, we also note that maximizing the largest eigenvalue gives 0.492 and maximizing the sum of the first three eigenvalues brings the percentage of captured variance up to 87.2%.

If we look at the plots of eigenvectors (scaled by the square roots of the eigenvalues) for the two-dimensional solution in Figure 4.6 and Figure 4.7, we see that the linear PCA produces groupings that are somewhat counterintuitive, mostly because there is so much variation left in the third and higher dimensions. The grouping in the NLPCA is clearer: psychologists in the same area are generally close together,

Table 4.3 Nine psychology areas.

Area	Plot Code
Social psychology	SOC
Educational and developmental psychology	EDU
Clinical psychology	CLI
Mathematical psychology and psychological statistics	MAT
Experimental psychology	EXP
Cultural psychology and psychology of religion	CUL
Industrial psychology	IND
Test construction and validation	TST
Physiological and animal psychology	PHY

Table 4.4 Roskam psychology subdiscipline data.

	SOC	EDU	CLI	MAT	EXP	CUL	IND	TST	PHY
1	1	5	7	3	2	4	6	9	8
2	1	3	2	7	6	4	5	8	9
3	1	6	5	3	8	2	4	7	9
4	1	5	4	7	6	2	3	8	9
5	7	1	4	3	6	8	2	9	5
6	6	1	2	5	3	7	8	4	9
7	2	1	4	5	3	8	6	7	9
8	4	1	2	8	3	5	9	6	7
9	4	1	3	5	7	6	8	2	9
10	3	1	2	4	6	8	9	7	5
11	4	1	8	3	7	6	2	5	9
12	3	2	1	5	6	8	7	4	9
13	2	9	1	6	8	3	4	5	7
14	2	7	1	4	3	9	5	6	8
15	7	2	1	3	5	8	9	4	6
16	5	7	8	1	3	9	4	2	6
17	5	9	8	1	2	7	6	3	4
18	9	6	5	1	3	7	8	2	4
19	9	6	7	2	1	8	3	4	5
20	8	3	7	2	1	9	4	5	6
21	7	2	8	5	1	9	6	4	3
22	8	7	6	3	1	9	2	5	4
23	8	6	5	2	1	9	4	7	3
24	8	7	5	2	1	9	6	4	3
25	7	3	6	2	1	9	8	4	5
26	4	7	9	5	1	8	2	3	6
27	5	6	8	2	1	9	4	7	3
28	1	8	9	2	3	7	6	4	5
29	2	5	6	4	8	1	7	3	9
30	2	5	4	3	6	1	8	7	9
31	5	3	2	9	4	1	6	7	8
32	4	5	6	2	8	7	1	3	9
33	5	7	9	3	2	8	1	4	6
34	6	3	7	2	8	5	1	4	9
35	8	5	7	4	2	9	1	3	6
36	2	6	5	4	3	7	1	8	9
37	5	8	9	2	3	7	1	4	6
38	8	7	3	4	2	9	5	6	1
39	5	6	7	2	4	9	8	3	1

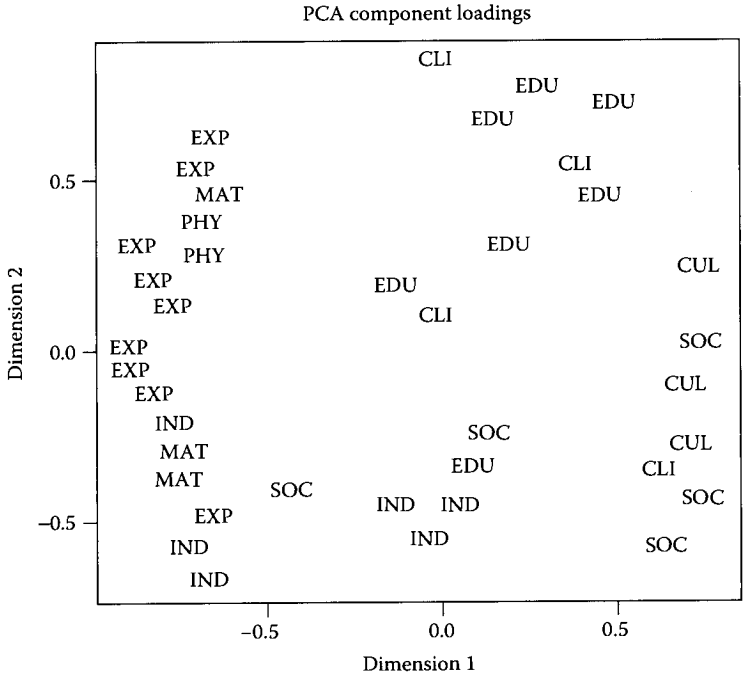

Figure 4.6 Roskam data: Linear PCA.

and there is a relatively clear distinction between qualitative and quantitative areas.

4.4 Logistic NLPCA

In the remainder of this chapter we discuss an entirely different way to define and fit NLPCA. It does not use least squares, at least not to define the loss function. The notion of correlation between variables is not used in this approach because we do not construct numerically quantified or transformed variables.

Suppose the data are categorical, as in MCA, and coded as indicator matrices. The indicator matrix $\mathbf{Z}_j$ for variable j has n rows and k_j columns. Remember that $\sum_{\ell=1}^{k_j} z_{ij\ell} = 1$ for all i and j. As in MCA, we represent both the n objects and the k_j categories of variable j as points $\mathbf{a}_i$ and $\mathbf{b}_{j\ell}$ in low-dimensional Euclidean space.

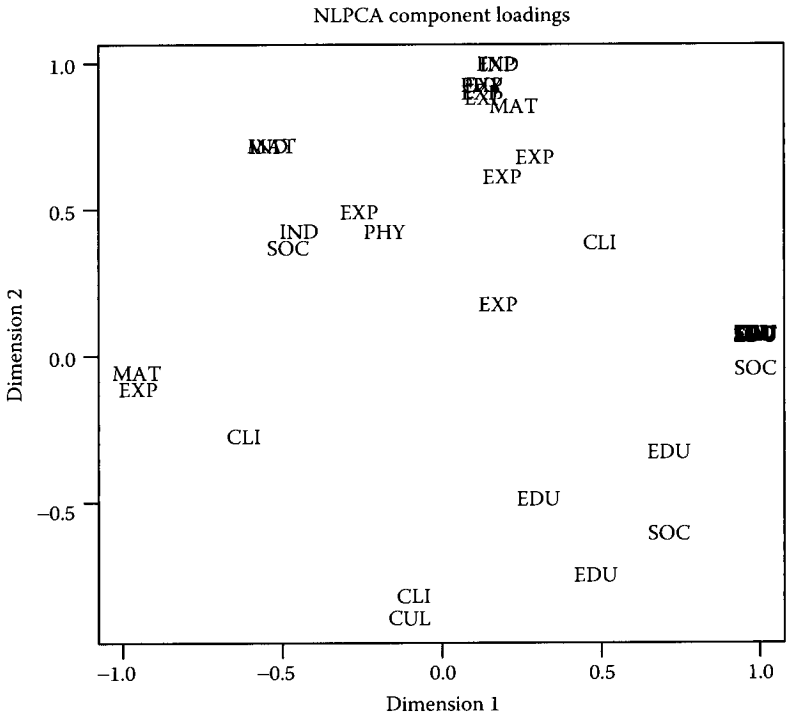

Figure 4.7 Roskam data: Nonlinear PCA.

We measure loss by using the *deviance*, or the negative log-likelihood,

$$\Delta(\mathbf{A},\mathbf{B}) = -\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\ell=1}^{k_j} z_{ij\ell}\,\log \pi_{ij\ell}(\mathbf{A},\mathbf{B})$$

where

$$\pi_{ij\ell}(\mathbf{A},\mathbf{B}) = \frac{\exp(\eta(\mathbf{a}_i,\mathbf{b}_{j\ell}))}{\sum_{v=1}^{k_j}\exp(\eta(\mathbf{a}_i,\mathbf{b}_{jv})}$$

For the time being, we do not specify the *combination rule* η, and we develop our results for a perfectly general combination rule. But to make matters less abstract, we can think of the inner product, $\eta(\mathbf{a}_i,\mathbf{b}_{j\ell}) = \mathbf{a}_i^{\top}\mathbf{b}_{j\ell}$, or the negative distance, $\eta(\mathbf{a}_i,\mathbf{b}_{j\ell}) = -\|\,\mathbf{a}_i - \mathbf{b}_{j\ell}\,\|$.

4.4.1 *Perfect fit*

In general, it will not be possible to find a perfect solution with zero deviance. We discuss under what conditions such a solution does exist. Consider the system of strict inequalities

$$\eta(\mathbf{a}_i, \mathbf{b}_{j\ell}) > \eta(\mathbf{a}_i, \mathbf{b}_{jv}) \tag{4.10}$$

for all (i,j,ℓ,v) for which $z_{ij\ell} = 1$. In other words, for all i and j the largest of the $\eta(\mathbf{a}_i,\mathbf{b}_{jv})$ must be the one corresponding to category ℓ for which $z_{ij\ell} = 1$.

Suppose Equation 4.10 has a solution $(\hat{\mathbf{A}},\hat{\mathbf{B}})$, and suppose our combination rule η is homogeneous in the sense that $\eta(\lambda\mathbf{a}_i,\lambda\mathbf{b}_{j\ell}) = \lambda^r\eta(\mathbf{a}_i,\mathbf{b}_{j\ell})$for some positive power r. Then by letting λ go to infinity, we see that $\pi_{ij\ell}(\lambda\hat{\mathbf{A}},\lambda\hat{\mathbf{B}})$goes to 1 for all $z_{ij\ell} = 1$, and thus $\Delta(\lambda\hat{\mathbf{A}},\lambda\hat{\mathbf{B}})$ goes to zero. We have a perfect solution, but with all points at infinity. While generally Equation 4.10 will not be solvable, we can perhaps expect some points to move to infinity in the actual solutions we compute.

4.4.2 *Geometry of combination rules*

In our further analysis, we concentrate on the particular combination rule using the negative of the distance, $\eta(\mathbf{a}_i,\mathbf{b}_{j\ell}) = -\|\mathbf{a}_i - \mathbf{b}_{jl}\|$. Equation 4.10 says that we want to map objects and categories into low-dimensional space in such a way that each object is closest to the category point in which it falls.

This can be illustrated nicely by using the notion of a *Voronoi diagram* (Okabe et al. 2000). In a Voronoi diagram (for a finite number, say p, points), space is partitioned into p regions, one for each point. The cell containing the point s is the locus of all points in space that are closer to point s than to the other $p - 1$ points. Voronoi cells can be bounded and unbounded, and in the Euclidean case they are polyhedral and bounded by pieces of various perpendicular bisectors. Using the $\mathbf{b}_{j\ell}$, we can make a Voronoi diagram for each variable. Our logistic PCA, for this particular combination rule, says that each object point $\mathbf{a}_i$ should be in the correct Voronoi cell for each variable.

This type of representation is closely related to representation of categorical data in Guttman's MSA-I, discussed by Lingoes (1968). It should also be emphasized that if the data are binary, then the Voronoi diagram for a variable just consists of a single hyperplane partitioning space into two regions. Equation 4.10 now says that the "yes" responses should be on one side of the hyperplane and the "no" responses should

be on the other side. This is a classical version of NLPCA, dating back to at least Coombs and Kao (1955), and used extensively in political science (Clinton et al. 2004).

To minimize the deviance, we use quadratic majorization (Böhning and Lindsay 1988; de Leeuw, 2006). We need the first and the second derivatives for a Taylor expansion of the deviance with respect to the $\eta_{ij\ell}$. We then bound the second derivatives to find the majorization function.

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\ell=1}^{k_j}[\eta_{ij\ell}(\mathbf{A},\mathbf{B})-\tau_{ij\ell}(\tilde{A},\tilde{B})]^2 \tag{4.11a}$$

where the current target is defined by

$$\tau_{ij\ell}(\tilde{A},\tilde{B})=\eta_{ij\ell}(\tilde{A},\tilde{B})-2[z_{ij\ell}-\pi_{ijl}(\tilde{A},\tilde{B})] \tag{4.11b}$$

Thus we can solve the logistic NLPCA problem by using iterative least squares. If we know how to fit $\eta_{ij\ell}(\mathbf{A},\mathbf{B})$ to a matrix by least squares, then we can also fit it logistically by maximum likelihood. In iteration k we compute the current target $\tau(A^{(k)},B^{(k)})$ by Equation 4.11b, and then we minimize (or at least improve) the least-squares loss function (Equation 4.11a) to find $A^{(k+1)}$ and $B^{(k+1)}$.

This implies immediately that for the inner product or bilinear composition rule η, we can use iterated singular-value decomposition, while for the negative distance rule we can use iterated least-squares multidimensional unfolding. In de Leeuw (2006), we give the details, and we show how the approach can easily be extended to deal with probit, instead of logit, loss functions.

As in Gifi (1990), we can construct variations on the basic technique by imposing constraints on the $\mathbf{b}_{j\ell}$. If we constrain them, for example, to be on a straight line through the origin by setting $b_{j\ell s}=z_{j\ell}\alpha_{js}$, then the bisecting hyperplanes will all be perpendicular to this line, and for each variable the space will be divided into parallel strips or bands. Objects should be in the correct strip. This is the form of NLPCA we have already discussed in the least-squares context, except that loss is measured on probabilities instead of correlations.

4.4.3 Example

The four GALO variables have a total of 24 categories, and there are 1290 individuals. Thus the metric unfolding analysis in each majorization step must fit 30,960 distances, using targets τ that can easily be negative. If we make all distances zero, which can be done by collapsing

all points, then the deviance becomes 1290*(log 2 + log 9 + log 6 + log 7) = 8550. This is, in a sense, the worst possible solution, in which all probabilities are equal.

We have written some software to optimize our loss functions. It has not been tested extensively, but so far it seems to provide a convergent algorithm. It starts with the MCA solution. Remember that in MCA (Michailidis and de Leeuw 1998) we want the $\mathbf{a}_i$ to be close in the least-squares sense to the category centroids $\mathbf{b}_{j\ell}$. In the graph-drawing interpretation (de Leeuw and Michailidis 1999), where we connect each category centroid to all the objects having that category in a star pattern, we want the category stars to be small. It seems reasonable to suppose that small stars will correspond with points in the correct Voronoi cell. The MCA solution starts with a negative likelihood of 8490 and improves this to 8315.

In Figure 4.8 we draw the Voronoi cells for IQ (observe that they are all open). The category points for IQ are almost on a circle (the horseshoe closes somewhat), starting with the lowest IQ category at the bottom center, and then proceeding clockwise to the higher categories. Similar plots can be made for the other variables, but we do not present them here.

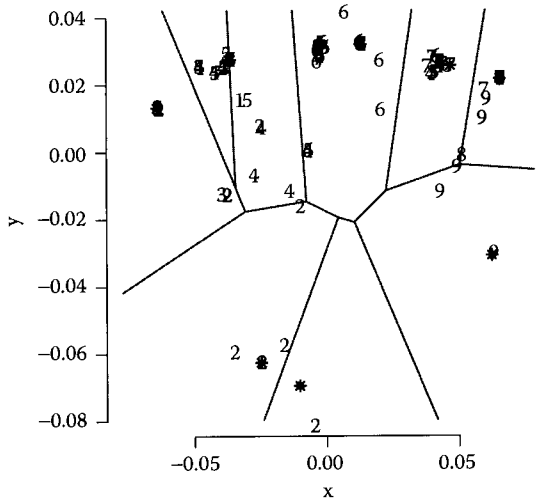

Figure 4.8 GALO data, intelligence, logistic PCA.

4.5 Discussion and conclusions

NLPCA offers many advantages over classical linear PCA because it can incorporate mixed measurement level data with ordinal, nominal, and numerical variables. It offers much flexibility in terms of admissible transformations and in terms of correlation aspects that can be maximized.

We have also seen that NLPCA has distinct advantages over MCA as an optimal scaling method. In the case of multinormal, or approximately multinormal, data, MCA will produce horseshoes and a very redundant representation of the basic information of the data. MCA can also be presented from a geometrical point of view (see Chapter 2), using the notion of chi-squared distance or minimizing the squared line length in a graph drawing of the star plots. There is no immediate generalization of the correlation aspect approach of NLPCA to these geometrical notions, although Gifi (1990) shows that NLPCA can be introduced by imposing restrictions on the location of the category quantifications in the joint MCA plots.

The solution for logistic NLPCA of the GALO data is presented somewhat tentatively because both theory and algorithm are new and will require much research and refinement. It is clear, however, that at least in principle, the basic theory and algorithms of Gifi (1990), which cover MCA, NLPCA, and various forms of nonlinear canonical analysis, can be extended to logit and probit loss functions that optimize aspects of probabilities instead of aspects of correlation coefficients.

4.6 Software Notes

There are quite a number of software options for performing the various forms of NLPCA explained in this chapter. PRINQUAL in SAS (1992) can optimize sums of the largest eigenvalues as well as the sum of correlations and the determinant aspect. Categories (Meulman and Heiser 1999) has CatPCA, which optimizes the classical eigenvalue criteria. In the R contributed packages we find the function homals from the homals package, which can perform NLPCA for categorical variables with or without ordinal constraints using the many options inherent in the Gifi system. There are also programs for NLPCA in the Guttman–Lingoes programs (Lingoes 1973).

The Gifi package for R has functions to optimize arbitrary aspects of the correlation matrix and to do the NLPCA of rank orders we

applied in the Roskam example. It includes the PREHOM program discussed by Bekker and de Leeuw (1988), which finds complete bilinearizable systems of scores if they exist, and the LINEALS program discussed by de Leeuw (1988). The R code is available from the author.

Code for the logistic (and probit) versions of PCA in R is also available. The binary version has been tested quite extensively (Lewis and de Leeuw 2004) and can be compared with similar programs for IRT analysis, written mostly by educational statisticians, and for roll-call analysis, written mostly by political scientists.