Component and Correspondence Analysis
Edited by J.L.A. van Rijckevorsel and J. de Leeuw

Chapter 3

Beyond Homogeneity Analysis

Jan de Leeuw
Department of Psychology and Mathematics
UCLA, Los Angeles, USA
and
Jan L.A. van Rijckevorsel
Department of Statistics, TNO NIPG, Leiden, The Netherlands

1. INTRODUCTION

In Gifi (1981a, 1988) a large number of multivariate analysis methods are organized in a single general framework. The key method in this system is *homogeneity analysis*, also known as *multiple correspondence analysis* (cf. Chapter 1). The Gifi system is inspired by ideas from multidimensional scaling, in particular by the central role of Euclidean distance in the representation of complex multivariate data. The basic data we want to represent geometrically are categorizations of n objects by m variables. Although the assumption that the variables are discrete and assume only a finite number of values is not essential, and can even be made without any practical loss of generality, it is true that in the current versions of homogeneity analysis categorical variables with a small number of categories play a central role. Variables with a large number of possible values, or even 'continuous' variables, can be incorporated in theory, but the implementations of the techniques more or less expect a small number of categories. If the number of categories is very large, say close to the number of objects that are classified, then homogeneity analysis as currently implemented (Gifi, 1981b) does not work very well. It will tend to produce unsatisfactory and highly unstable solutions, in which 'chance capitalization' is a major source of variation (cf. also Chapter 2).

There have been various attempts to make the solutions more stable by imposing restrictions that reflect, in some sense, the prior information we have

about the variables. In De Leeuw (1984a) these restrictions are classified into *rank-restrictions*, *cone-restrictions* and *additivity-restrictions*. Imposing restrictions decreases the number of free parameters. This means, roughly, that there are more data values per parameter, which can consequently be determined in a more stable manner. Rank-restrictions and cone-restrictions make it more easy to deal with variables having a large number of categories, but in several respects their treatment remains somewhat unsatisfactory. In many multidimensional scaling programs there are options for transformation of the variables that are 'smooth' or otherwise 'continuous'. There is no such possibility in the current homogeneity analysis programmes. In this chapter we shall try to extend the basic geometry of homogeneity analysis in such a way that continuous variables fit in more easily using the coding systems as discussed in Chapter 2. A fundamental role in this extension is played by the 'B-spline basis' and its 'fuzzy' generalizations, which is introduced here in a purely geometrical way, that is mainly due to van Rijckevorsel (1987). This additionally indicates more clearly how homogeneity analysis generalizes the various forms of non-metric principal component analysis (cf. Chapter 1). Combination of the various options creates a very flexible new type of homogeneity analysis. It is highly unlikely that all possible combinations will be equally important in practice, in fact we suspect that some of the less restricted forms will again tend to produce very unstable or even 'trivial' solutions. Nevertheless it is satisfactory from a theoretical point of view to show exactly what the choices are that one has to make, even if some of the possible choices may be quite unwise in practical situations.

2. SIMPLE HOMOGENEITY ANALYSIS

We start with a brief recapitalization of the technique of homogeneity analysis introduced in Chapter 1, without any of the frills discussed by Gifi (1981a, 1988) or de Leeuw (1984a, 1984b). The data are m variables on n objects, i.e. there are m functions defined on a common domain $\{1, 2, \ldots, n\}$. We suppose that the range of function j has k_j elements, and we code function j by using the $n \times k_j$ *indicator matrix* G_j. Matrix G_j is binary, it has exactly one element equal to one in each row, indicating into which element of the range the object corresponding to this row is mapped. Thus the rows of G_j add up to one, and the matrix $D_j = G_j'G_j$ is diagonal, and contains the univariate marginals. $G_{jl} = G_j'G_l$ is the cross-table of variables j and l and contains the bivariate marginals. This notation is illustrated in detail in Chapter 1.

The purpose of homogeneity analysis is to map both objects and variables into low-dimensional Euclidean space R^p (where p is *dimensionality*, chosen by the user). We want to do this in such a way that both objects and categories of the variables are represented as points, and in such a way that an object is relatively close to a category it is in, and relatively far from the categories it is not in. Of

rse this implies, by the triangle inequality, that objects mostly scoring in the
e categories tend to be close, while categories sharing mostly the same objects
l to be close too. The extent to which a particular representation X of the
cts and particular representations Y_j of the categories, satisfy the desiderata
omogeneity analysis is measured by a least squares loss function. This is
ned as

$$\sigma(X; Y_1, \ldots, Y_m) = \sum_j \operatorname{tr}(X - G_j Y_j)'(X - G_j Y_j). \tag{2.1}$$

rder to prevent certain obvious trivialities we require that the $n \times p$ matrix of
cts scores X is normalized by $u'X = 0$ and $X'X = nI$. Here u is a vector with all
nents equal to one, and I is the identity matrix. We do not normalize the m
rices of *category quantifications* Y_j, which are of order $k_j \times p$. Using (2.1) and
normalization conventions we can now give a more precise definition of
nogeneity analysis. It is to choose a normalized X and $Y_1, \ldots, Y_m$ in such a
that (2.1) is minimized. For additional interpretations of the loss function, in
ns of consistency discrimination and homogeneity, we refer to Gifi (1981a)
de Leeuw (1984a). In this Chapter we more or less ignore the algorithmic and
istical aspects of the homogeneity analysis techniques, and concentrate on the
metry on which the loss function is based.

3. PICTURES OF LOSS

Table 3.1 we have presented a small example with ten objects and three
ables. The objects are ten cars, the variables are price (in \$1000), gas
sumption (litres per 100 km, on the expressway) and weight (in 100 kg). The
a are taken from Chapter 6, Table 6.1. In order to prevent possible
understandings we must emphasize that Table 3.1 in this chapter is not at all
resentative for data usually analysed with homogeneity analysis. In fact, in
st practical applications of the technique, the number of objects and the

Table 3.1. Car data

	Price	*Gas*	*Weight*
vette	5.6	6.9	9.7
ge Colt	5.7	5.1	8.8
nouth Horizon	6.3	5.5	9.9
Mustang	7.6	6.7	12.0
tiac Phoenix	8.6	6.9	12.1
ge Diplomat	9.4	10.2	15.5
vrolet Impala	10.1	7.5	16.9
k Regal	10.5	7.8	15.0
C Eagle	10.7	11.7	15.7
smobile 98	13.3	8.7	18.3

number of variables is much larger. Moreover in our small example all variables are numerical, which is also not typical for most homogeneity analysis applications.

The data in Table 3.1 cannot be used directly in homogeneity analysis. They must first be made discrete or categorical. This is done by grouping the values of the variables into discrete categories, which can, of course, be chosen in many different ways. One possible, fairly crude, categorization is given in Table 3.2. Observe that there are three cars with *profile* (1, 1, 1), and two cars with (2, 1, 2). Thus there are only seven different profiles for these ten cars, out of possible $3 \times 3 \times 4 = 36$ profiles. A finer discretization would give more possible profiles, more different actual profiles, and also more 'empty cells', i.e. more profiles that do not occur. The finest discretizations is the *ranking* given in Table 3.3. Here there are $10^3 = 1000$ possible profiles, of which only 10 are in use. Thus 99 per cent of the cells are empty. Observe that in constructing Table 3.3 from Table 3.1 we have arbitrarily broken a tie in variable 2 (Chevette and Pontiac Phoenix both score 6.9 in gas consumption).

Table 3.2. Car data, discrete

	Price	*Gas*	*Weight*
Chevette	1	1	1
Dodge Colt	1	1	1
Plymouth Horizon	1	1	1
Fort Mustang	2	1	2
Pontiac Phoenix	2	1	2
Dodge Diplomat	2	3	2
Chevrolet Impala	3	2	3
Buick Regal	3	2	2
AMC Eagle	3	3	2
Oldsmobile 98	4	2	3

Table 3.3. Car data, ranked

	Price	*Gas*	*Weight*
Chevette	1	4	2
Dodge Colt	2	1	1
Plymouth Horizon	3	2	2
Fort Mustang	4	3	4
Pontiac Phoenix	5	5	5
Dodge Diplomat	6	9	7
Chevrolet Impala	7	6	9
Buick Regal	8	7	6
AMC Eagle	9	10	8
Oldsmobile 98	10	8	10

Now suppose we choose object scores X in two dimensions, and category ıantifications Y_j also in two dimensions. We have plotted the objects scores we ve chosen as ten points in Figure 3.1. Also given in Figure 3.1 are the three ints corresponding with the categories of variable 1, price. To make a picture of ss, for variable 1, we have connected all objects with the category point they long to, according to variable 1. Loss-component 1 is simply the sum of uares of the line-lengths drawn in Figure 3.1. We can make a similar picture for riable 2, if we also choose Y_2. It is important to realize that we have chosen X

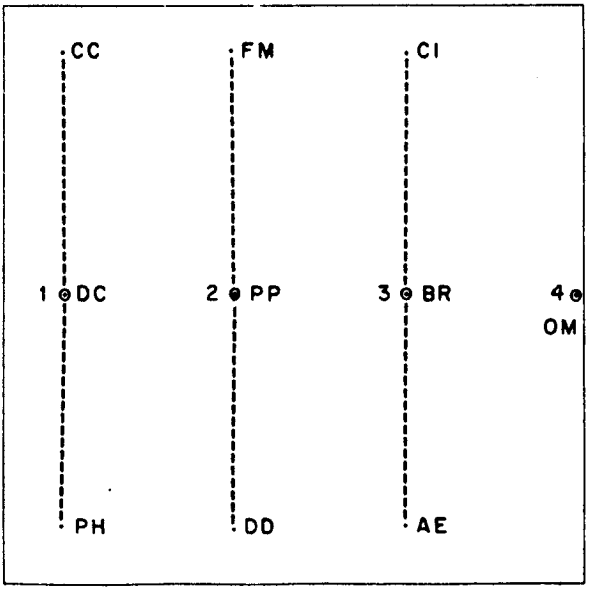

Figure 3.1. Loss variable 1, arbitary solution

ıd Y_1 completely arbitrary, and not by any optimality considerations. They are ot, in any sense, the solutions given by homogeneity analysis. In fact they are erely candidates for the solutions, and it is the purpose of the technique to find tter candidates. Another important point is that we can also make 'dual' ctures, in which we plot all Y_j as points together with a single object point. The ss 'due to object i' can now be represented by drawing lines from the object int to all category points it is in. Such plots, as well as the plot in Figure 3.1, are ıb-plots' of a large plot which contains all object points and all category points, ıd which has a line for each element equal to one in each indicator matrix. This ıper-plot' will generally look somewhat messy, so it is better to present it in ıyers'. In Figure 3.2 we have presented the optimal solution computed by ɔmogeneity analysis, i.e. the optimal object scores and the optimal quantifica-ɔns of the categories of variable 1. It is clear that the line lengths are shorter for

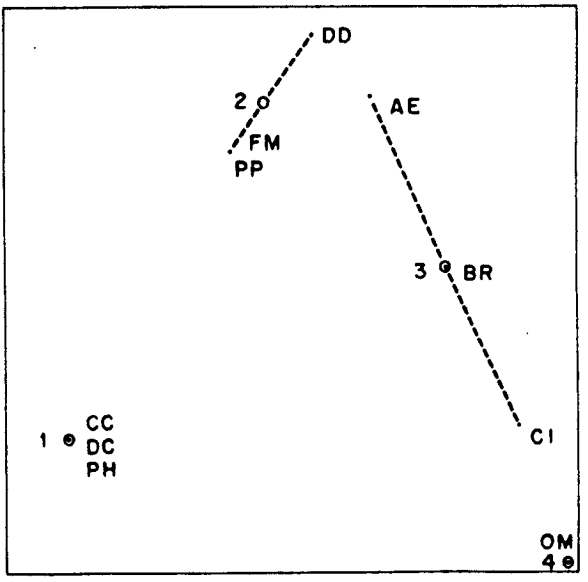

Figure 3.2. Loss variable 1, optional solution

the optimal solution. For other types of plots useful in homogeneity analysis we refer to Gifi (1981a, 1981b, 1988).

4. RANK RESTRICTIONS

In simple homogeneity analysis category quantifications can be anywhere in p-space. From equation (2.1) it follows that optimal category quantifications are centroids of objects points in the categories. This is illustrated in Figure 3.2. In fact in Figure 3.1 category quantifications of variable 1 are also optimal for the given object scores, only the object scores are very far from optimal in this case. Because of the *centroid-property* of optimal category quantifications it follows that their weighted average, with weights equal to the marginal frequencies, is the origin. This is the only restriction on the relative position of the quantifications of the categories within a variable. Now consider the situation in which variables have a range which is ordinal or even numerical. This constitutes a form of prior information which is not used by simple homogeneity analysis, and which consequently may get lost in the representation computed by homogeneity analysis. If we look at Figure 3.2 the categories of variable 1 are represented in the 'correct' order. This is true if we measure order along the horizontal axis, and even more clearly true if we measure order along the 'horseshoe' on which all objects lie. For variable 2, gas consumption, the situation is quite different, however. Only Dodge Diplomat and AMC Eagle are in category 3, which means

at the optimal quantification of the category will be the midpoint of the line nnecting DD and AE. Category 2 contains CI, OM and BR, and will be antified close to CI. Category 1 will be between cluster CC, DC, PH and cluster P, FM. Thus both on the horseshoe and on the line the categories will project in e order 1–3–2, which is contrary to our prior information. In this chapter we scuss geometrically inspired methods which both prevent the horseshoe and ake it possible to impose our prior information.

A familiar way to get rid of the horseshoe is to do this by imposing rank-one strictions (van Rijckevorsel, 1987). By this we mean that we require all category antifications of a variable to be on a line through the origin of p-space, with ch variable having its own line. In matrix notation this means that we require $=z_ja_j'$, i.e. the $k_j \times p$ matrix Y_j must be of rank-one. In Chapter 1, in order to stinguish the various types of category quantifications that result from this idea e Y_j are called *multiple category quantifications*, while the z_j are called *single tegory quantifications*. The a_j are the loadings of variable j. We now minimize e loss function (2.1), with the provision that for some variables (but not cessarily for all) we use the restrictions $Y_j=z_ja_j'$. Variables for which the strictions are imposed are called *single variables*, variables without restrictions e *multiple variables*. A program for homogeneity analysis with mixed multiple d single variables is discussed by Gifi (1982).

In order to study the geometry of single variables we expand the corresponding ss component first. This gives

$$\begin{aligned}\operatorname{tr}(X-G_jY_j)'(X-G_jY_j) &= \operatorname{tr}(X-G_jz_ja_j')'(X-G_jz_ja_j') \\ &= np-2a_j'X'G_jz_j+(z_j'G_j'G_jz_j)(a_j'a_j). \end{aligned} \tag{4.1}$$

ow let $q_j=G_jz_j$, and normalize z_j such that $u'q_j=0$ and $q_j'q_j=n$. Such ormalization is used merely for identification purposes, because z_j only occurs in e product z_ja_j'. Using the normalization we find

$$\operatorname{tr}(X-G_jY_j)'(X-G_jY_j)=n(p-1)+(q_j-Xa_j)'(q_j-Xa_j). \tag{4.2}$$

his shows, in the first place, that single loss cannot possible be zero if p is larger an one. It is always at least $n(p-1)$. It is equal to $n(p-1)$ if all objects in a tegory project in the same point on the line through the origin and a_j. Or, to put differently, if categories define parallel hyperplanes orthogonal through the line efining the variable. All objects in a category must be located in the hyperplane f the category. The elements of z_j are the signed distances to the origin of the tegory hyperplanes, i.e. the location of the projections on the line defining the ariable. In the case of non-perfect fit the loss is simply the distance of each object oint from its category hyperplane, or, more precisely, the squared distance. igure 3.3 illustrates this for a particular choice of X, z_1, and a_1 in our small car xample. Again no optimality considerations are used here, in fact we have not ven paid attention to the appropriate normalizations. It is clear that rank-one

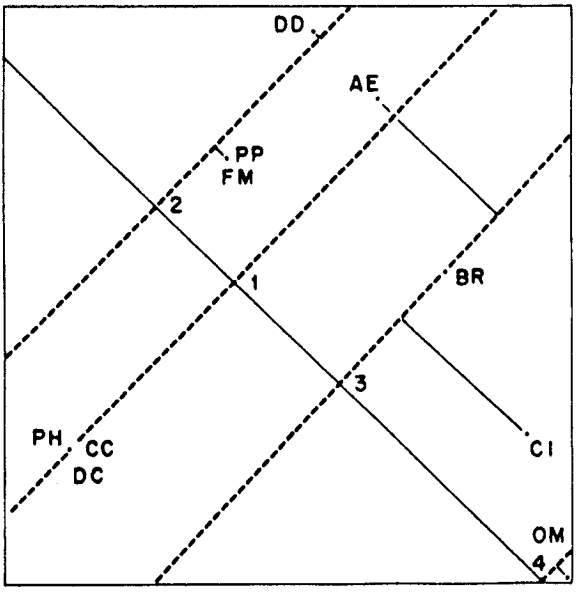

Figure 3.3. Single nominal loss, variable 1, arbitrary solution

restrictions will tend to make horseshoes impossible, or at least highly unlikely. It may not be clear yet how they can be used to impose ordinal or numerical prior information. Before we proceed to explaining this, remember that the use of single variables is related to performing a *principal component analysis* as in Chapter 1.

5. CONE RESTRICTIONS

Rank-one restrictions induce an order on the categories of the variable, even if we do not know the order beforehand. The induced order is given by the projections on the variable vector, or by the order of the category hyperplanes. In fact the category hyperplanes even introduce a single numerical scale for the categories of a variable, given in the vector z_j. Now the induced ordinal or numerical information may or may not correspond with our prior knowledge. We use *cone restrictions* if we impose the constraint that the induced order must be the same as our prior order, and the induced scale must be the same as our prior scale. Numerically these are restrictions on the elements of z_j. Either they must be in the 'correct' order, for *single ordinal* variables, or they must be equal to a given normalized vector, for *single numerical* variables. Observe that the type of a variable refers to the constraints we impose, it does not reflect some intrinsic property of the variable. We use the term 'cone restrictions' because the feasible choices for z_j form a polyhedral convex cone in k_j-space for ordinal variables, and

one-dimensional subspace, which is a sort of degenerate cone, for numerical riables. It is also possible, as is done in Chapters 4, 5 and 6, to formulate our trictions in terms of $q_j = G_j z_j$, i.e. in n-space or in the scalar-product space of ctors q_j $(j = 1, \ldots, m)$. No restrictions on z_j, defining *single nominal* variables, ines a k_f-dimensional subspace in n-space. Ordinal and numerical restrictions ines subcones and subspaces of this k_f-dimensional subspace.

f the z_j are completely given, by restrictions taken together with normaliza-ns, then homogeneity analysis becomes identical with principal component alysis, cf. Chapter 1. This is, in a sense, one of the endpoints of the continuum of mogeneity analysis techniques. All variables are single numerical; the other dpoint has all variables *multiple nominal*. This is what we have described earlier simple homogeneity analysis or multiple correspondence analysis. In Figure we give a two-dimensional principal component analysis representation of small example, using the geometry or homogeneity analysis.

Figure 3.4 results from analysing Table 3.2. It is clear, of course, that the alysis of Tables 3.1 and 3.3 would give different results in general. Table 3.3 is te interesting in this respect. For Table 3.3 the indicator matrices G_j are mutation matrices. If we substitute them in (2.1) it is obvious that loss can ays be made equal to zero by letting X be an arbitrary $n \times p$ matrix, and by ing $Y_j = G_j' X$. Then $G_j Y_j = G_j G_j' X = X$. In the same way single nominal iables can always be fitted perfectly. Choose X and a_j arbitrarily, and set $G_j' X a_j$. Then $q_j = X a_j$, and loss is minimized by (4.2). In other words:

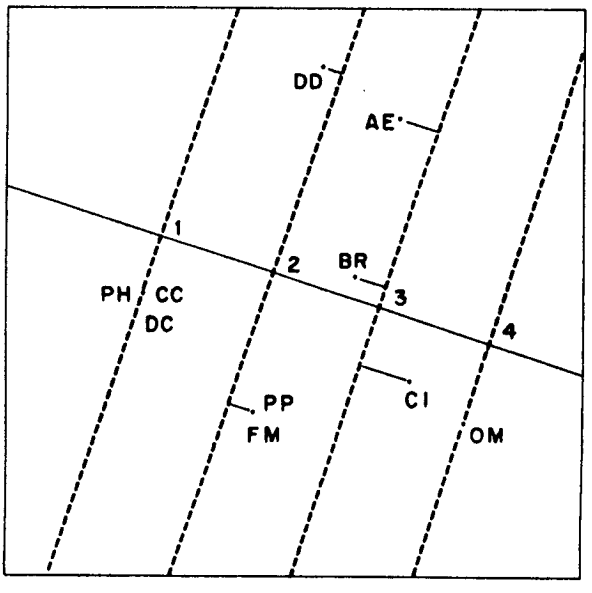

Figure 3.4. Single numerical loss, variable 1, optimal solution

non-trivial analysis of rankings is possible only if we make all variables either single ordinal or single numerical. It is also interesting to compare the single quantifications in $q_j = G_j z_j$ with the original scores in Table 3.1. Clearly plotting the elements of q_j versus the original scores will give a step-function. We have discreticized our variables, and as a consequence every object in the same discretization interval gets the same quantification in the q-vector of the variable. The more intervals, the less crude the transformation given by the step-function will be, but no matter how fine we choose the discretization, the transformation will always be a step function. And step functions do have several drawbacks as we can verify in Chapter 2. This is one of the main reasons why we say that homogeneity analysis as currently implemented by Gifi (1981b, 1982) has a discrete bias. Step-functions are perfectly natural for variables which have a small number of possible values to start with, or for purely nominal variables for which we have no prior numerical information. For 'continuous' numerical variables, such as the three variables in our example, transformation by step-functions ignores the prior information that our variable was originally continuous, and can also assume all intermediate values between the end-points. Thus we now know how to incorporate numerical and ordinal information, but we do not know yet how to incorporate 'smoothness' into homogeneity analysis. This problem will be discussed below, but first we have to fill a number of gaps that have been left open in the combination of various options we have discussed up to now.

6. GAPS IN GIFI

In the previous sections we have discussed single numerical, single ordinal, single nominal, and multiple nominal variables. We did not discuss multiple ordinal and multiple numerical. If only for aesthetic reasons it is interesting to investigate if these remaining types of variables can also be given a simple meaning. Moreover we have distinguished single and multiple variables. For single variables we required that rank (Y_j) was less than or equal to one, for multiple variables there were no rank restrictions, which means that we 'required' that rank (Y_j) was less than or equal to min $(p, k_j - 1)$. It is $k_j - 1$ and not k_j in this upper bound, because of the fact that the rows of Y_j have a weighted mean of zero. Now if $p = 1$ there is no difference between multiple and single. If $p = 2$ then for variables with more than two categories single requires that rank (Y_j) is less than or equal to one and multiple that rank (Y_j) is less than or equal to two. There is no gap between the two options. But for $p = 3$, and k_j larger than three, single requires rank one and multiple requires rank three as the upper bound. Thus there is a gap. We can insert another option, which requires rank (Y_j) to be less than or equal to two. This general rank restriction, which can be between single and multiple, was already discussed in de Leeuw (1976), but it was not incorporated in the subsequent developments of the Gifi system.

The loss function, with general rank constraints, can be written as

$$\sigma(X; Y_1, \ldots, Y_m) = \sum_j \operatorname{tr}(X - G_j Z_j A_j')'(X - G_j Z_j A_j'). \tag{6.1}$$

re Z_j is $k_j \times r_j$, and A_j is $p \times r_j$. The r_j are the required ranks for variable j. ometrically the constraint means, of course, that the category quantifications st be in a r_j-dimensional hyperplane through the origin. If $Z_j'D_jZ_j = nI$, then for variable j satisfies

$$\sigma_j(X, Y_j) = n(p - r_j) + \operatorname{tr}(XA_j - G_jZ_j)'(XA_j - G_jZ_j). \tag{6.2}$$

$= (A_1 \mid \ldots \mid A_m)$ and $Q = (Q_1 \mid \ldots \mid Q_m) = (G_1Z_1 \mid \ldots \mid G_mZ_m)$, then

$$\sigma(X; Y_1, \ldots, Y_m) = nm(p - r) + \operatorname{tr}(XA - Q)'(XA - Q). \tag{6.3}$$

s looks very similar to (4.2), but remember that in (6.3) each Q_j consists of r_j nogonal quantifications of the same variable, i.e. of r_j *copies* (compare de uw, 1984a; Tijssen, 1984; de Leeuw and Tijssen, 1984). Again, geometrically, have minimum loss if the category points are in an r_f-plane, and all object nts are on lines perpendicular to the plane, which cross the plane in the ategory points.

General rank restrictions now make it possible to define r_j-nominal, in which re are no further restrictions on Z_j. There is also r_j-numerical, in which the r_j umns of Z_j are known orthogonal k_j-vectors. And, finally, there is r_j-ordinal, in ch all columns of Z_j must be in the appropriate order. For r_j-nominal and umerical we can require, without loss of generality, that $Z_j'D_jZ_j = nI$. For r_j-inal such a constraint cannot be imposed, and we have to refrain from malizing Z_j and/or A_j. It is clear, of course, that general rank constraints, pled with *measurement restrictions*, generalize our previous notions of single l numerical, and fill the gaps in the system. In fact it opens completely new sibilities: we can require that the first 'copy' in Z_j is ordinal, while the aining copies are nominal, and so on. Again we do not know how practical se new options are. We have discussed them because they fit naturally into the s, and also because they can be incorporated without much ado into the nogeneity analysis algorithms that are already there.

7. PSEUDO-INDICATORS

Chapter 2 it is illustrated that a more satisfactory analysis of continuous iables becomes possible if we generalize the notion of an indicator matrix. ppose we continue to use the same notion of loss, with the same types of trictions on the category transformations, but we do not suppose that the G_j indicator matrices. They must still be known $n \times k_j$ matrices, but they need not binary any more. In a sense we have already gone a step in this direction. If a iable is r_j-numerical, then $Y_j = G_j(Z_jA_j') = (G_jZ_j)A_j'$. Suppose, for instance, that

the Z_j are polynomials, orthogonal with respect to the marginals. Then G_jZ_j are orthogonal polynomials in n-space, and we can interpret our analysis as an unrestricted analysis using an $n \times r_j$ basis of orthogonal polynomials instead of the indicator matrix G_j. Although this is clearly a valid interpretation, it is not exactly what we have in mind.

In this section we concentrate on so-called *fuzzy codings*, collected in *pseudo-indicator* matrices. Indicator matrices are characterized as pseudo-indicators with bandwidth unity, cf. Chapter 2. Piecewise linear B-splines define pseudo-indicators with bandwidth two, and so on. In this chapter we do not care about the origin of the pseudo-indicators, for this we refer to Chapter 2. We simply assume that data are coded in this way, and we look for the geometrical interpretations of such a coding. In Table 3.4 we have a fuzzy coding of our small

Table 3.4. Piecewise linear coding car data

	Price			*Gas*			*Weight*			
Chevette	0.88	0.12	0.00	0.62	0.38	0.00	0.06	0.94	0.00	0.00
Dodge Colt	0.86	0.14	0.00	0.98	0.02	0.00	0.24	0.76	0.00	0.00
Plymouth Horizon	0.74	0.26	0.00	0.90	0.10	0.00	0.02	0.98	0.00	0.00
Fort Mustang	0.48	0.52	0.00	0.66	0.34	0.00	0.00	0.60	0.40	0.00
Pontiac Phoenix	0.28	0.72	0.00	0.62	0.38	0.00	0.00	0.58	0.42	0.00
Dodge Diplomat	0.12	0.88	0.00	0.00	0.96	0.04	0.00	0.00	0.90	0.10
Chevrolet Impala	0.00	0.98	0.02	0.50	0.50	0.00	0.00	0.00	0.62	0.38
Buick Regal	0.00	0.90	0.10	0.44	0.56	0.00	0.00	0.00	1.00	0.00
AMC Eagle	0.00	0.86	0.14	0.00	0.66	0.34	0.00	0.00	0.86	0.14
Oldsmobile 96	0.00	0.34	0.66	0.26	0.74	0.00	0.00	0.00	0.34	0.66

example, which is actually the result of piecewise linear coding. The idea behind our generalization of homogeneity analysis now is, that we can combine all our previous options and restrictions with this new coding as well.

In particular we can impose rank-constraints, and impose ordinal or numerical restrictions.

Because $p=2$ in our example it suffices to distinguish single and multiple. Consider multiple nominal. The loss component for variable j vanishes if $X=G_jY_j$. In the coding used in Table 3.4 each X corresponds with two categories, because the bandwidth in our example is two. The two category quantifications are the endpoints of a line segment, all line segments for a particular variable are connected. The object scores must be on the line segment corresponding to the categories they are in. And not only must they be on the segment, they must also be in a precise location on the segment, where the location is dictated by the masses of the endpoints in the coding. This is indicated in Figure 3.5, which is not an optimal solution of any kind, but it is used to illustrate the loss of variable 1 in the coding of Table 3.4. The points on the two line segments indicate where the

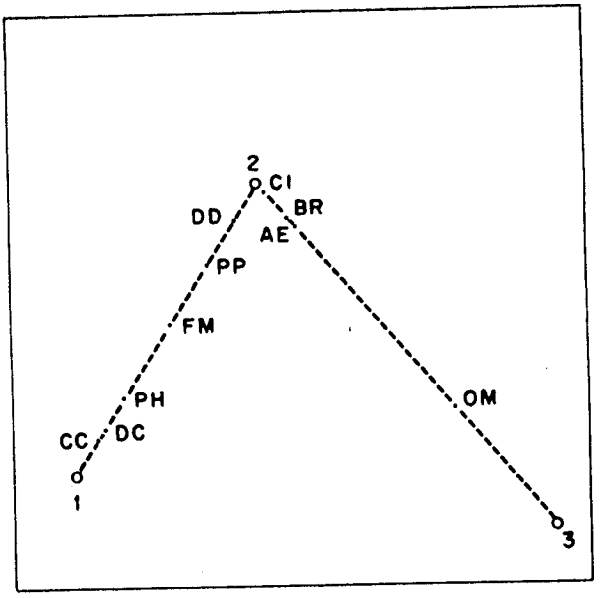

Figure 3.5. Loss for multiple piecewise linear, variable 1, arbitrary solution

cars must be given the coding, and given the location of the endpoints of the segments. In the single case the endpoint must be on the same straight line, and the object points must project on the places fixed by the coding. Thus there are parallel lines perpendicular to the line connecting the category points, which intersect this line at the appropriate places. In the ordinal case the endpoints must be ordered along the line, such that both within-category and between-category quantification is ordered (which makes this a somewhat peculiar option, perhaps).

The geometry of fuzzy or smooth homogeneity analysis with several applications is extensively treated in van Rijckevorsel (1987). If we study the transformation which considers $q_j = G_j z_j$ as a function of the original data values, then transformations from pseudo-indicators as discussed in Chapter 2, will indeed be more smooth than those from indicators. The precise nature of the smoothness depends on the nature of the pseudo-indicators, for instance on the bandwidth. In our example the transformations are continuous and piecewise linear. If we use piecewise quadratic splines, joined in such a way that they are differentiable at the endpoints, then we get more smoothness (and a bandwidth of three). The geometry becomes more complicated, because object scores must be at the appropriate places in the triangle spanned by three endpoints. Successive triangles are interlocked, because they have one side in common. And so on, for larger bandwidths, and/or in higher dimensions. The relationship of the object scores with the multiple quantifications (Y-configuration) and its interpretation

are to be reconsidered. Several questions do arise: What happens to the *principe barycentrique* in fuzzy homogeneity analysis? Are the original (= not coded) data reproducible from the final configuration? What is the geometrical significance of the goodness of fit parameters?

7.1. The representation of basis functions

First order B-splines (crisp coding) collapse all values within an interval into a point. Second order B-splines force all values within one interval to be on a line. Third order B-splines transform all values to be on the face of a triangle.

The regular polygon is often used as a triangle of reference for the position of data points in the basis, cf. Le Foll (1979), Gallego (1980), Greenacre (1984) and van Rijckevorsel (1987). In this way we can represent at most three (orthogonal) dimensions, i.e. basis functions, in a plane. This implicitly uses the property that all fuzzy codes of one data point add up to one. The use of triangular coordinates is limited to the representation of three basis functions at the time.

This tool enables us to show the differences between various low dimensional forms of fuzzy coding in terms of triangular coordinates. The triangle of reference with the vertices (1, 0, 0), (0, 1, 0) and (0, 0, 1) is also known as the triangular or barycentric coordinate system. The number of coordinates $\neq 0$ is maximally three per data point. In Chapter 2 this number is also called the bandwidth of the set of basis functions.

In the cases (a) and (b) in Table 3.5, one triangle suffices to represent the whole transformation function because there exist only three basis functions, cf. Figure 3.6: A, B_1 and B_2.

The crisp codes in Figure 3.6 coincide with the vertices. The codes in the fuzzy areas around the knots are the points on the sides of the triangle. The transition from first order fuzzy coding into second order codes is clearly because in the latter case all codes are between the vertices. This automatically leads to second order B-splines, where all points are on the sides between the vertices and only coinciding with a vertex, if the data-point coincides with a knot.

Table 3.5. Dimensionality, bandwidth and order of low order fuzzy coding with two interior knots

Type of coding	*Order*	*Dimension*	*Bandwidth*
(a) Crisp coding	1	3	1
(b) First order fuzzy coding	1	3	2
(c) Second order B-splines	2	4	2
(d) Third order B-splines	3	5	3

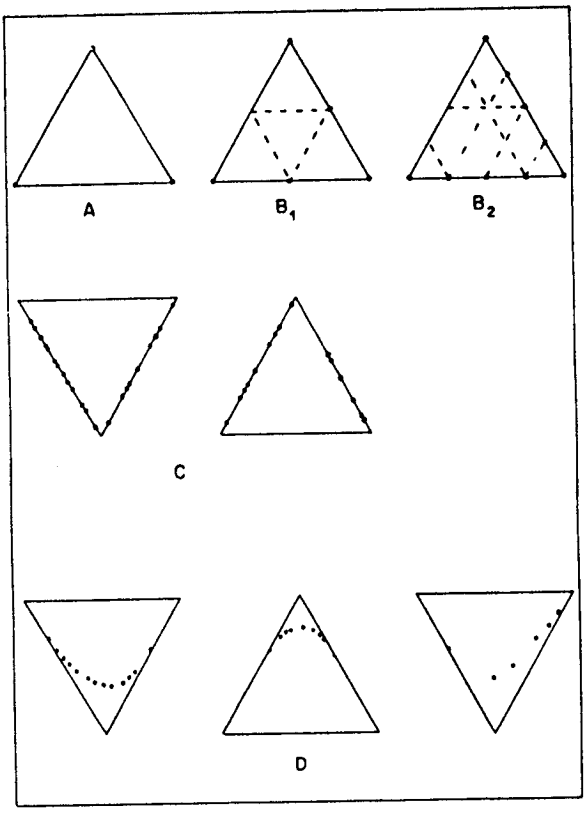

Figure 3.6. The Public spending in the Netherlands between 1951 and 1981 coded by crisp coding (A), demi-discrete (B), trapezoidal (B2), piecewise linear (C) and quadratic coding (D), all represented by triangular coordinates after van Rijckevorsel (1987), see also Chapter 2

The first degree fuzzy coding represents three intervals that have two coordinates $\neq 0$ on four basis-functions, cf. Figure 3.6 (C). One triangle is not sufficient because of the dimensionality of the basis. One way of solving this is by using an additional triangular coordinate system that has one dimension, i.e. one side between two vertices, in common with the first triangle, in order to maintain the simplicity and parsimony of this approach. The first triangle covers the first two (out of three) intervals and the second triangle the last two (out of three) intervals. There exists an overlap of one interval, and they have two vertices in common; i.e. the triangles are interlocked.

The second degree coding of Figure 3.6 (D), is a code with three different coordinates $\neq 0$, that add up to one, which is represented by a triangle of reference in Figure 3.7.

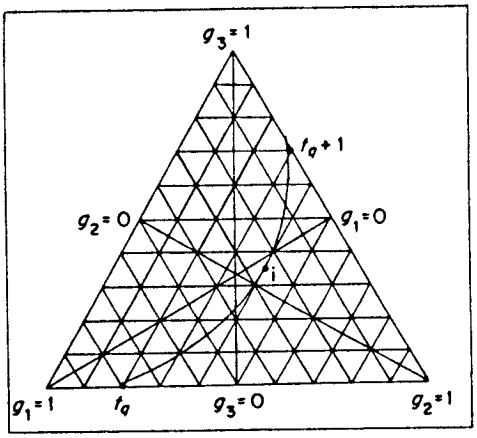

Figure 3.7. The fuzzy coding of data point i by second degree B-splines: $\{g_1=0.25, g_2=0.40, g_3=0.35\}$, represented by the barycentric coordinate system

It is clear that, if one of the three coordinates equals zero, the corresponding point is on the side of the triangle. The coordinates of the knots are t_q: $\{g_1=0.8, g_2=0.2, g_3=0.0\}$ and t_{q+1}: $\{g_1=0.0, g_2=0.7, g_3=0.3\}$. Note that in every knot one of the codes is equal to zero. *Ergo* the knots are on the sides of the triangle and the points within the interval are on the face of the triangle. All data values between t_q and t_{q+1} are on the quadratic curve between t_q and t_{q+1}. Each basis vector is a quadratic function and hence the triangular representation is a quadratic as well.

In this way the generalization to second degree codes, cf. Figure 3.6 (D), is easy to understand. The restricting parameter has evolved from a point, via a line segment, to the face of a triangle. A second degree B-spline is a quadratic function on the face of the triangle of reference smoothly joining at two sides.

7.2. The build-up of a transformation function

In crisp coding data points are represented as grouped points, by first degree B-splines as individual points on line segments and by second degree B-splines as points on curves. We know from Chapter 2 that the global transformation function is equal to a piecewise function.

Say, we use a triangle of reference to represent the coding of a variable. Then we can observe the same phenomena, i.e. point, line segment and curve in the barycentric representation. If we inspect the functional coefficients in the space of the object scores, which is the usual way of inspecting the parameters in homogeneity analysis, we observe the same phenomena: point, line segment and curve. This is to be expected; the sets of functional coefficients span the subspaces that geometrically restrict the object scores. In case of perfect fit the object scores

collapse, not into the functional coefficients, but into the regular polytopes spanned by these coefficients. They span points, line segments and curves on the face of a triangle. This means that a perfect fit in fuzzy homogeneity analysis is trivial in a different way from a perfect fit in crisp homogeneity analysis. The latter demands that all data points collapse perfectly into a few categories. This seems less realistic than demanding that all data points within an interval should form a linear or quadratic function, which permits the expression of a considerably larger amount of variation in a controlled way (see Figure 3.8).

Note that the transformation G_jY_j coincides with the functional coefficients in crisp coding, it connects the coefficients in first degree B-splines by straight lines and in second degree B-splines it forms a quadratic curve that contains only those functional coefficients that correspond to the exterior knots.

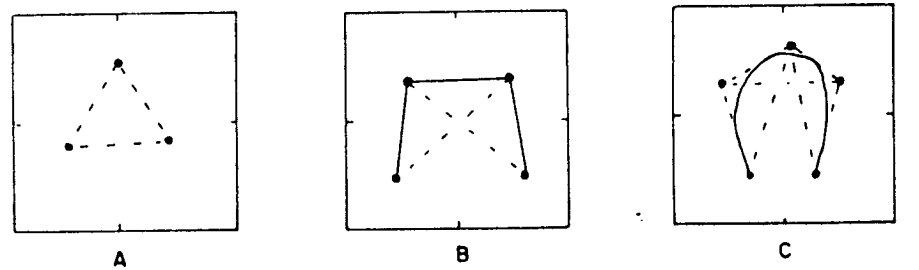

Figure 3.8. The functional coefficients Y_j (=dots), the weighted triangles of reference (=dotted lines) and the global multiple transformation functions G_jY_j (=solid lines) in the X-space. Represented for crisp coding (A), first degree B-splines (C) and second degree B-splines (D)

7.3. Goodness of fit

The goodness of fit of point i on variable j is defined as is the custom in homogeneity analysis: the squared euclidean distance between the observation score x_i and the corresponding value on the global multiple transformation function $g_{ij}Y_j$. See the dotted lines in Figure 3.9.

The subspaces, spanned by the functional coefficients, can be interpreted geometrically as restrictions for the corresponding object scores. Using crisp codes, the objects scores should be as close as possible to the point Y_{jk}; using hat codes (=second order B-splines), the object scores should be as close as possible to the corresponding points on the line segments between the points $Y_{j,k}$ and $Y_{j,k+1}$; as for the bell codes (=third order B-splines) the object scores should be as close as possible to the curve within the face of the triangle, spanned by the functional coefficients $Y_{j,k}$, $Y_{j,k+1}$ and $Y_{j,k+2}$. We can now develop a schematic geometrical account of what happens to a variable in fuzzy homogeneity analysis.

The data are coded respectively by all five nearly-orthogonal fuzzy codes discussed in Chapter 2. The symbols used in the Figures 3.10 to 3.13 are A (crisp),

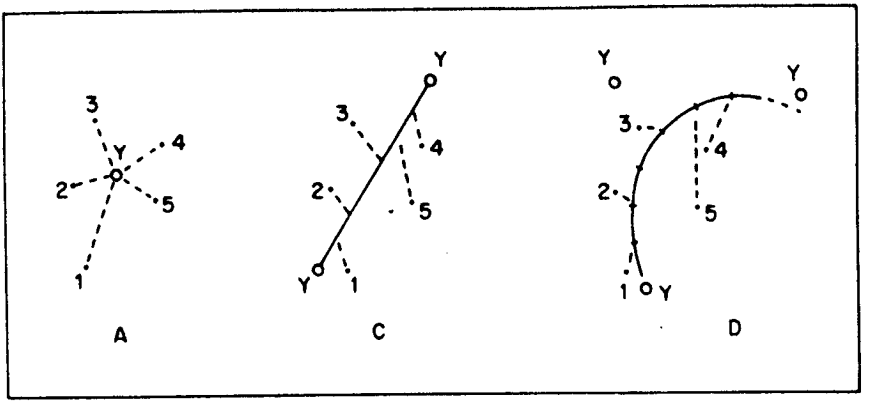

Figure 3.9. The euclidean distances between x-points and the global multiple transformation function in one interval for three different ways of coding

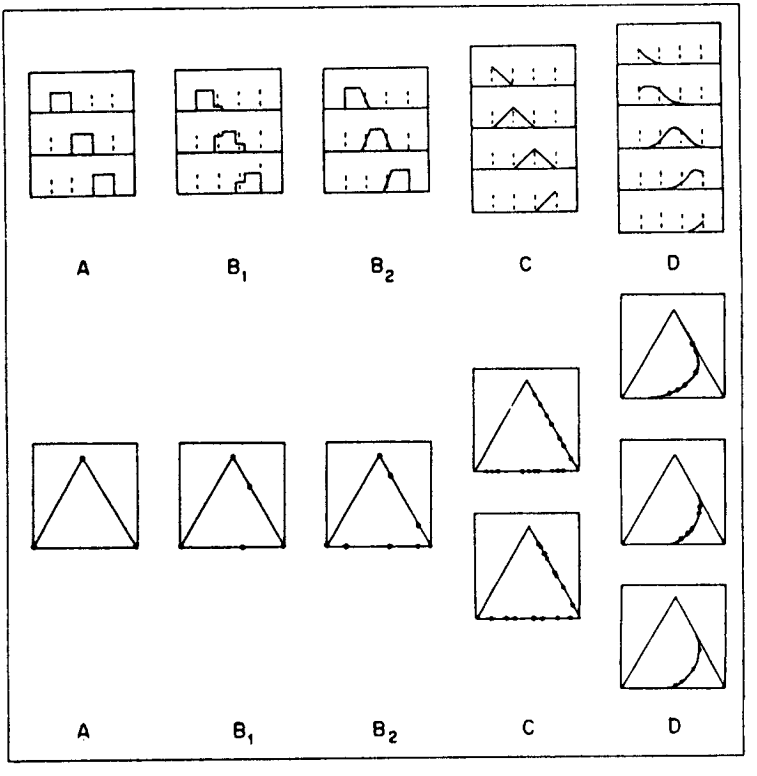

Figure 3.10. Five different types of fuzzy coding functions with the representations by triangular coordinates; see the text for the interpretation of the labels

B1 (semi-discrete), B2 (trapezoidal), C (first degree B-spline) and D (second degree B-spline). The obtained basis vectors are represented by triangles of reference: maximally three basis vectors per triangle. A triangle of reference can represent three intervals in case of zero degree codes, two intervals in case of first degree codes and one interval in case of second degree codes (and no intervals for higher degree codes).

The basis vectors, and thus the vertices of the triangles of reference, are weighted with respect to the p-dimensional X configuration for maximal homogeneity by the least squares estimates $Y{:}G_jY_j$, while $G_{jk}Y_j$ is the quantification of data points in the kth interval, see Figure 3.11.

The weighted basis functions expressed by the sides of triangles, form together the global multiple transformation function in the p-dimensional space, see Figure 3.12. Nothing new is introduced here. The procedure is extensively discussed in Chapter 2 and in this chapter. Each side of the triangle is geometrically speaking, separately stretched respectively shrunk, by the least squares estimation.

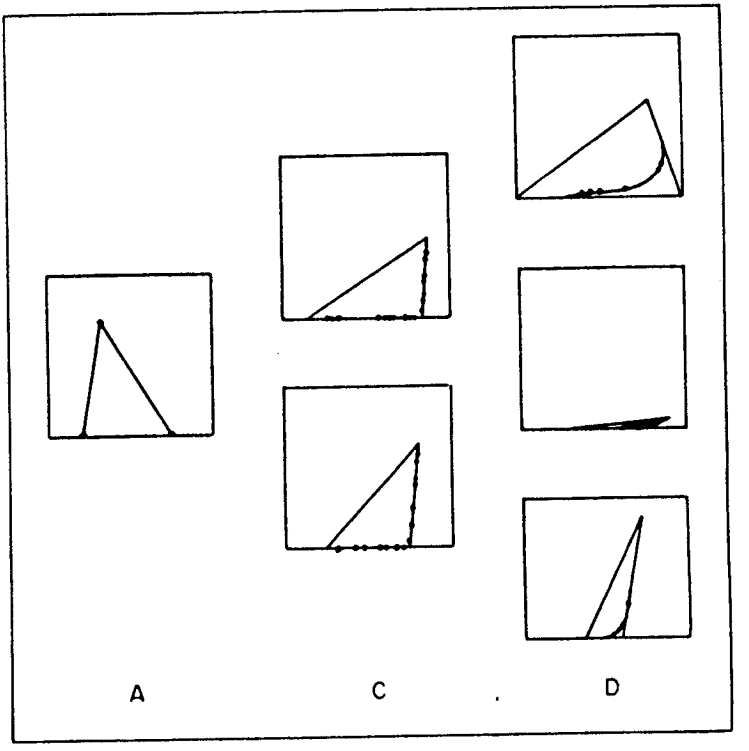

Figure 3.11. The weighted triangular representations of three different types of fuzzy coding: see the text for the interpretation of the labels

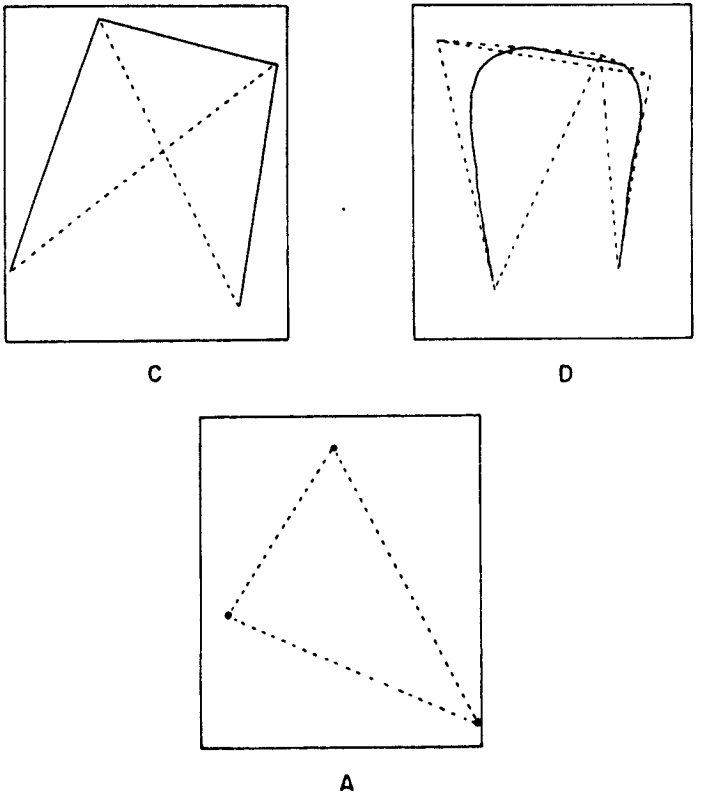

Figure 3.12. Weighted triangular representations (dotted) and resulting multiple transformation functions (solid) in the first two dimensions of fuzzy homogeneity analysis, based on three different types of fuzzy coding

The weighted triangles have a distinct relationship with the X-configuration: It follows from this picture that bandwidth three or more does not combine naturally with single quantification, because single quantification makes the triangles degenerate to straight lines. This is no problem analytically, but it makes the geometry of loss far less interesting. In general we think that for practical purposes a bandwidth larger than two is probably not very interesting, unless data are very well behaved indeed.

8. RELATED WORK ON FUZZY HOMOGENEITY ANALYSIS

The combination of homogeneity analysis and fuzzy coding is fairly recent. The development of fuzzy set theory, comprehensively reviewed by Bezdek (1987), took place independently from the development of homogeneity analysis. See also van Rijckevorsel (1987). Fuzzy coding itself is introduced by Zadeh (1965)

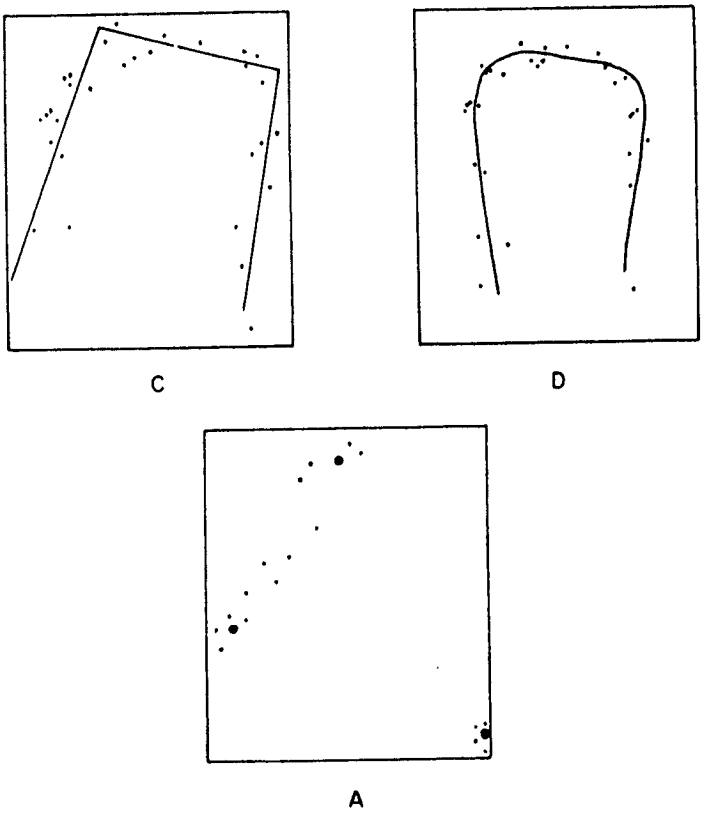

Figure 3.13. Multiple transformation functions and X-points in the plane of the first two dimensions of fuzzy homogeneity analysis based on three different types of fuzzy coding

and Ruspini (1969) and the first proposals for fuzzy coding in homogeneity analysis are by Bordet (1973), Guitonneau and Roux (1977) and Ghermani, Roux and Roux (1977). These early papers mainly adapt continuous data to the discrete mould of homogeneity analysis and hence try to smooth the inaccuracy of crisp coding around the knots. The application within homogeneity analysis seems to be rather accidental; theoretically speaking it could have been any other crisp technique. In the same vein Martin (1980) derives a probabilistic framework to this end. See also Chapter 5. The first attempts to incorporate fuzzy coding and homogeneity analysis are by Le Foll (1979) and Gallego (1980). They both concentrate on piecewise linear coding. Le Foll generalizes to a larger class of coding strategies, which he defines as *codages complets* to be used in a variety of techniques. The larger part of his work, however, is devoted to an application of piecewise linear coding within homogeneity analysis on ecological data, referring to the surface water pollution in the larger surroundings of Paris (France).

Gallego (1980) concentrates on an application of piecewise semi exponential coding within several techniques in order to smooth seasonal macroeconomic data. In this context he discusses linear PCA (on recoded data) and cluster analysis but predominantly homogeneity analysis.

Some French work on fuzzy coding is marked by the desire to analyse real valued and categorical data simultaneously within one analysis. Fuzzy coding is thus a means to incorporate real valued data by coding them into a format that conforms with homogeneity analysis. And, consequently, because fuzzy coding in itself increases inertia, much attention is given on methods how this should be corrected for, prior to further analysis. cf. Guitonneau and Roux (1977) Benzécri (1980), Gallego (1980) and Greenacre (1984, p. 162).

Another group of French authors prefer to work with a probabilistic interpretation of fuzzy coding in non-linear analysis, cf. Chapter 5. Martin (1980) is most outspoken and other work often uses his definition of probability coding, that a fuzzy code is mainly a transition probability between an observed and an unobserved variable, cf. Besse and Vidal (1982), Gautier and Saporta (1982) and Mallet (1982). The idea is that the unobserved random variable is reconstructed by coding the observed variable. Besse and Vidal (1982) extend this idea to both variables, observed and unobserved, being categorical and restrict the coding to the bivariate coding of pairs of variables (not to be confused with the bivariate coding of pairs of intervals as mentioned in Chapter 2). Ramsay (1982) and Besse and Ramsay (1986) discuss the (smooth) PCA of data which are functions, cf. Chapter 4. This work should not be confused with the work of Winsberg and Ramsay (1983) who consider the isotone polynomial spline transformations of separate variables as a kind of (probabilistic) optimal scaling. See also the ACE methodology applied in this context by Koyak (1985), and the work of Winsberg and Kruskal (1986). Chapter 6 deals separately with the latter.

Apart from these developments there exists another tendency to relate fuzzy coding within homogeneity analysis to the theoretical non-linear principal component analysis as defined by Dauxois and Pousse (1976). Fuzzy coding is then regarded as a way to further the convergence of homogeneity analysis to a theoretically completely non-linear generalized canonical analysis, where non-linear variables are non-linearly related. This is linked to the two types of convergence discussed in Chapter 2. Nearly all the French research in this field published after 1976 refers to this form of non-linear generalized canonical analysis. See also Lafaye de Micheaux (1978) and Mallet (1982). The latter conjectures that the empirical analysis of fuzzy-coded variables is a good approximation of the theoretical non-linear analysis.

9. PROCESS

In the developments so far the data were coded as (pseudo)-indicators, and these pseudo-indicators were fixed during the computations. Now let us look at single

ordinal piecewise linear again. We have already seen that the order of the category points on the line is fixed in this case, although their precise location is free. Given the location of the category points, however, the location of the preferred projection of the objects points on the line is fixed by the coding. This is what we mean by fixedness of within-category order. This fixedness is contrary to what is called the *primary approach to ties* in multidimensional scaling literature, and also the *continuous ordinal* option (compare de Leeuw, Young and Takane, 1976, Young, de Leeuw and Takane, 1980, Young, 1981). In this option, which is incorporated in various non-metric principal component programs, we fix the order between categories but not within categories. Or, geometrically, given the line and the location of the category points on the line, the object-point can project *anywhere* between the end-points of its category. Loss only occurs if they project outside their assigned interval.

Given our previous discussion it is easy to see how the idea of continuous ordinal data can be incorporated easily into our form of homogeneity analysis. The elements of the pseudo-indicators are not considered fixed any more, only the location of the non-zero elements is fixed. Thus we know which elements must be non-zero, we also know that they must be non-negative and they must add up to one for each row, but their precise values are additional parameters over which the loss function is minimized. In the single ordinal piecewise linear case this gives exactly continuous ordinal data as treated in PRINCIPALS, for instance (Takane, Young and de Leeuw, 1978). But because we have fitted the possibility of varying the elements of the G_j into our general homogeneity analysis framework, we can combine this option with all other previous options that we already had. It can be combined with multiple quantification, and with single numerical quantification. In this last case it gives the continuous numerical scaling earlier discussed by de Leeuw and Walter (1977).

There is very little need to elaborate on the geometry of the continuous versions. It is basically the same as the discrete geometry, only points are not fixed in intervals, but they can be anywhere in the interval. It becomes perhaps a bit more interesting to use larger bandwidths with single options, because the bandwidth now controls the amount of overlap of the intervals corresponding with the categories. If bandwidth is two, there is no overlap. If bandwidth is three, successive categories have one common subinterval, and so on. Multiple options with bandwidth three, in two dimensions, are interpreted in terms of triangles (or convex hulls). Objects in category 1 must be in the convex hull of category points 1, 2 and 3, objects in category 2 in the convex hull of 2, 3 and 3, and so on. Successive triangles have one side in common, if they degenerate to line segments this becomes the overlapping subinterval. It is not at all clear (yet) if these conceptually very nice options are useful in practice. A theorem in Gifi (1981a, 1988) is useful to illustrate their limitation. It refers to the continuous ordinal option, with all variables single. The results show that with this option *degenerate*

solutions, which locate one object very far away from the others, which are collapsed into a single point, will be quite common. In fact Gifi shows that in the situation in which objects are a random sample the minimum of loss is almost surely equal to zero if the sample size tends to infinity. Van Rijckevorsel (1987) illustrates with real data that the conceptual nicety can be misleading. We do not know yet how devastating the results are in practice, but it certainly indicates that we have to be careful.

Computationally our new options do not introduce any trouble at all. We must introduce a new subproblem into the alternating least squares cycles of homogeneity analysis in which the G_j are adjusted. This is done for each row of each G_j separately, defining a very small special quadratic programming problem. Of course we have to exert a little self-control in combining our options. We have the possibility, in principle, to take a different bandwidth for each object, or a different rank for each Y_j. In fact, looming large in the distance, is the possibility of further generalizations. We can fix the bandwidth of each variable, for instance, and determine the optimum location of the non-zero elements. This is probably very unwise, because the program output will become almost independent of the data.

It is perhaps convenient to relate existing programs to our general form homogeneity analysis, in which we choose (a) quantification rank, (b) measurement level, (c) bandwidth, (d) process for each variable separately. HOMALS (Gifi, 1981b) has quantification rank equal to dimensionality, measurement level nominal, bandwidth unity and process discrete. Of course if bandwidth is unity there is no distinction between discrete and continuous process. Ordinary principal component analysis has quantification rank unity, measurement level numerical, bandwidth unity and process discrete. PRINCALS (Gifi, 1982) has quantification rank either one or dimensionality, and measurement level numerical, ordinal or nominal (but ordinal/numerical cannot occur together with multiple), bandwidth is unity and process is discrete. SPLINALS (van Rijckevorsel, 1982, 1987; Coolen, van Rijckevorsel and de Leeuw, 1982) has quantification rank either one or dimensionality, measurement level nominal, bandwidth either one or two and process discrete. Winsberg and Ramsay (1983) have, with some minor qualifications, measurement level ordinal, quantification rank unity, arbitrary bandwidth, and process discrete. PRINCIPALS (Takane, Young and de Leeuw, 1978) has quantification rank-one, measurement level nominal, ordinal or numerical, bandwidth either one or two, process continuous or discrete. But if the process is continuous the measurement level must be ordinal, and if the process is discrete the bandwidth must be one. It is clear that our new homogeneity analysis program, which only exists in preliminary APL-versions yet, encompasses all these possibilities and has all previous programs as special cases. Of course it will be more expensive in terms of time and storage, and more liable to produce degeneracy.

10. WORDS OF CAUTION

Homogeneity analysis is a dangerous technique. We use very little information from the data, and we do not impose restrictions of a strong type on the representation. This type of program traditionally appeals greatly to many social scientists, who are very unsure about the value of their prior knowledge. They prefer to delegate the decisions to the computer, and they expect programs to generate knowledge. This strategy leads, all too often, to chance capitalization, triviality and degeneracy. Hypotheses are never rejected, and investigators are constantly making errors of the second kind. As a consequence results can, of course, never be replicated. Generalized homogeneity analysis, as we have developed it here, is a very powerful tool which can contribute greatly to a further inflation of social science results. By choosing the least restrictive options we can make the results almost completely independent of the data.

On the other hand it is well known that if we pay too much attention to errors of the second kind, then social scientists can say abolutely nothing. This is also considered to be an undesirable state of affairs. It can be circumvented by concentrating on minute aspects of well-defined small problems, as in laboratory situations, or it can be circumvented by introducing vast quantities of prior knowledge, as in sociology. Of course in most cases the prior knowledge is nothing but prejudice, and it so dominates the investigation that the results become equally independent of the data.

This defines the dilemma of applied empirical social science. According to the canons of scientific respectability we can say almost nothing, and the things we can say are likely to be trivial. There are two ways out of this situation. Either we impose so much prior knowledge on our problem that the data only marginally make a difference. This is the rationalistic solution, popular in sociology. Or we impose so little prior knowledge that the data, including all outliers, stragglers, idiosyncrasies, coding errors, missing data, completely determine the solution. In this case the technique is supposed to generate theory. This is the empiristic and technological approach, popular in applied psychology. Both approaches have, up to now, not produced much of interest.

Homogeneity analysis is firmly in the empiristic and technological tradition. Thus it is clear what dangers we have to guard against especially. If we have reliable prior knowledge, we must incorporate it. It is absolutely necessary to investigate the *stability* of the results (Gifi, 1981a; de Leeuw, 1984b). Observe, however, that stability is not sufficient. A program that responds to any data matrix by drawing the unit circule is very stable indeed. We also need to *gauge* the technique, by comparing analysis with different options on data whose most important properties are known. For some forms of homogeneity analysis this has already been done quite extensively (Gifi, 1981a; de Leeuw, 1984c), but apart from results by van Rijckevorsel (1987) very little is known in this respect about the more general options discussed here. One strategy, that seems promising, is to

analyse the same data with various options, from numerical to ordinal, from bandwidth one to bandwidth two, from discrete to continuous, and so on. In fact, this defines another form of stability analysis, which seems indispensable in situations with little prior knowledge.