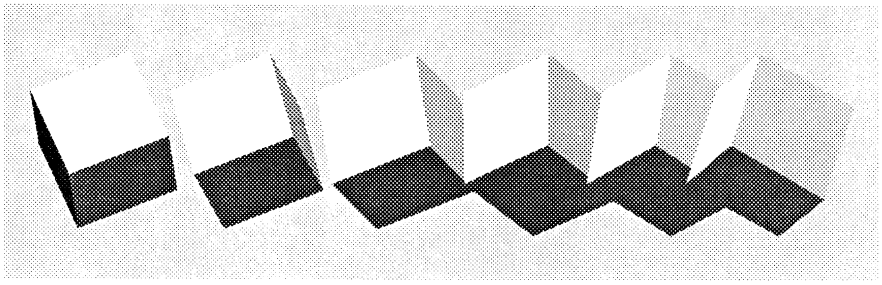

Chapter 2

Generalized Procrustes analysis

2.1 Introduction

In this chapter the first and most simple model of the MATCHALS program will be discussed. Following Van de Geer's analogy (1984), the problem in this model may be visualised as follows. Suppose we have two configurations $\mathbf{X}_1$ and $\mathbf{X}_2$, both configurations containing the coordinates of the same p stimuli in two dimensions. Further suppose that the stimulus points of $\mathbf{X}_1$ are drawn on a transparent sheet, and that the stimulus points of $\mathbf{X}_2$ are drawn on another transparent sheet. The two sheets are arbitrarily placed on top of each other. The problem to be solved in GPA, the abbreviation of Generalized Procrustes Analysis that we use throughout this book, is: How should the two sheets be moved such that the sum of the squared distances between corresponding pairs of points on the two sheets becomes as small as possible? This criterion in GPA for the evaluation of the match between configurations $\mathbf{X}_1$ and $\mathbf{X}_2$ is illustrated in Figure 2.1, where the stimulus points are the vertices of two pentagons. To minimize the sum of squared distances between corresponding pairs of points, the two transparent sheets may be manipulated in a number of ways. However, there is one important restriction: We only consider those transformations that leave the relative distances between stimulus points within each of the two configurations unchanged. The implications of this restriction are illustrated for configuration $\mathbf{X}_1$ in Figure 2.2, where the stimulus points of $\mathbf{X}_1$ are numbered from

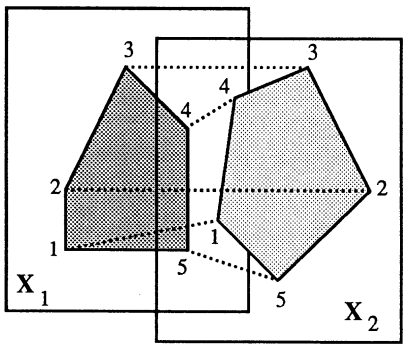

Figure 2.1 Illustration of match criterion in GPA for two configurations. Sum of squared lengths of dotted lines must be as small as possible.

one to five before transformation and from one prime to five prime after transformation.

The first transformation illustrated in Figure 2.2 shows that the stimulus points in a configuration may all be moved into the same direction and along an equal distance. This transformation is called a translation. The second admissible transformation is to turn over the sheet with $\mathbf{X}_1$ on it, known as a reflection. The third admissible transformation is to stretch or shrink the sheet containing $\mathbf{X}_1$ in a uniform fashion, also called a central dilation or uniform rescaling. The last kind of manipulation that

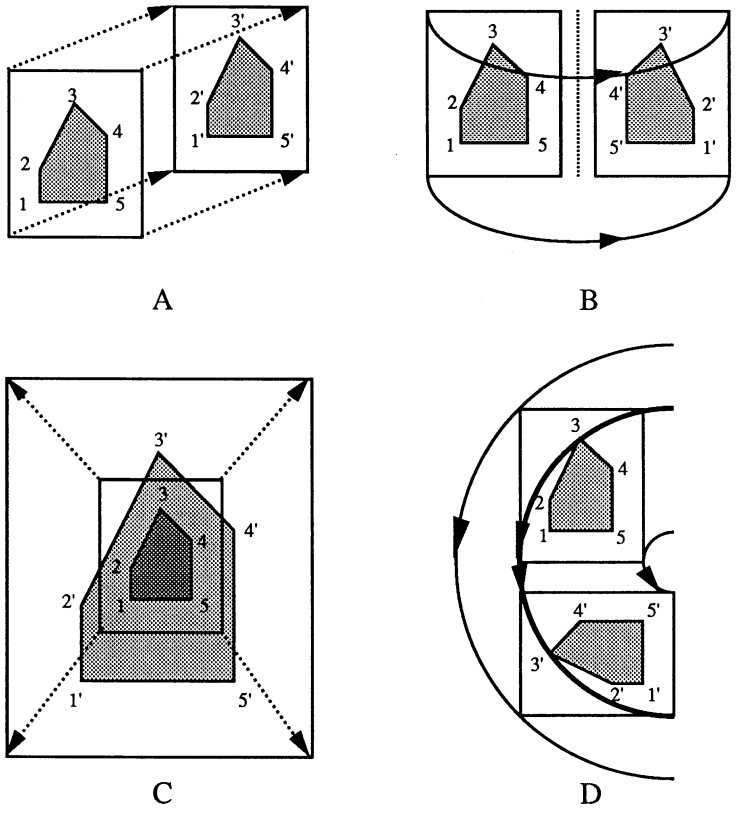

Figure 2.2 Admissible transformations in GPA. A: translation, B: reflection, C: uniform rescaling and D: rotation.

the two sheets are allowed to undergo in GPA in order to enhance their match is a rotation. In Figure 2.2 configuration $\mathbf{X}_1$ is rotated counterclockwise through an angle of 90°. Since, in practice, the problem of rotating and reflecting a configuration can be solved simultaneously, in the sequel these two transformations will jointly be referred to as an *orthonormal transformation.*

None of the four manipulations of the sheet representing configuration $\mathbf{X}_1$ in Figure 2.2 affects the ratios of the distances between the five stimulus point of $\mathbf{X}_1$. These same four transformations may be applied to configuration $\mathbf{X}_2$ in order to reduce the sum of squared distances between the corresponding pairs of points in $\mathbf{X}_1$ and $\mathbf{X}_2$.

In this very simple example consisting of the two configurations $\mathbf{X}_1$ and $\mathbf{X}_2$ shown in Figure 2.1 the reduction in sum of squared distances could roughly be brought about by just moving the two sheets around until the sum of squared distances seems to be as small as possible. But even in this simple case it is difficult to visualise how the two sheets should be stretched or shrunk uniformly in order to enhance their match. Moreover, in practice there will be more than five stimulus points to consider, there will be more than two configurations to match, and last but not least the configurations may be more than two-dimensional.

The groundwork for GPA was laid by Green (1952), Cliff (1966) and Schönemann (1966) who solved the problem of orthonormally transforming (i.e., rotating and reflecting) a matrix such that the sum of squared distances between pairs of points of the latter matrix and another matrix is minimized given that the two configurations have an equal number of columns. The solutions of Green and Cliff only work for matrices of full column rank, while Schönemann generalized the solution to matrices of deficient column rank. Schönemann and Carroll (1970) investigated and solved the problem of fitting one matrix onto another matrix using not only an orthonormal transformation, but also a translation and uniform rescaling.

Kristof and Wingersky (1971) extended orthonormal transformations to the case where more than two configurations with equal numbers of columns are involved. Gower (1975) again solved the problem of fitting more than two configurations using orthonormal transformations as well as translations and central dilations when the configurations have an equal number of columns. He was also the first to coin the term Generalized Procrustes Analysis. In 1977 Ten Berge corrected Gower's method for the determination of the central dilations, and improved Gower's procedure for the

calculation of the orthonormal transformations. Ten Berge and Knol (1984) extended the problem to the case where two or more configurations have a different number of columns (although only for orthonormal transformations). Very recently, Peay (1988) has solved the problem of fitting more than two configurations with differing dimensionalities under the general rule illustrated in Figure 2.2, that is including translations and central dilations.

All these latter extensions of GPA typically are concerned with data sets consisting of matrices having unequal numbers of columns. In this chapter we assume that the matrices have equal column orders, but generalize GPA to the case where the configurations have a different number of *rows*, that is where information about some stimuli in some of the configurations is missing. Previous work in this area has been done by Everitt and Gower (1981) in the context of what they called a 'weighted generalized Procrustes method', of which the problem discussed in this chapter is a special case. However, they did not incorporate central dilations in this method (while these are essential as will be discussed in section 2.3.5), neither did they provide a rigorous treatment of translations. Moreover, a more efficient procedure than Everitt and Gower's method will be proposed in this chapter for the estimation of the orthonormal transformations. Other work on GPA and incomplete configurations has been done by De Leeuw and Meulman (1986) in the context of a Jackknife procedure. They investigated how to match a number of configurations when each configuration has missing information about only one stimulus. De Leeuw and Meulman's solution is a special case of the matching procedure to be developed in this chapter, since we consider the case where the configurations may have any pattern of missing data.

The solution proposed in this chapter is useful in the following general situations. A number of researchers have collected data on the same topic and submitted their data to one or another form of multidimensional scaling or multivariate analysis (like factor or principal components analysis). Each researcher thereby produces a configuration of the stimuli that he or she is studying. However, different researchers not necessarily use the same stimuli when investigating the same topic: some stimulus sets may overlap, others may not. The same thing may happen, of course, if someone repeats an experiment in time. Then too, he or she may choose to use varying numbers of stimuli. In both cases it is of interest to investigate whether the resulting configurations have an identical structure, notwithstanding the fact that the number of stimuli may vary from configuration to configuration.

2.2 The direct and the centroid approach: geometry and algebra

We embark on the solution of the problem of developing the GPA model for configurations containing missing data by giving, without proof, the following identity (see, e.g., Kristof & Wingersky, 1971; Gower, 1975) which is at the core of much that is to follow in this chapter.

Theorem 1. Given n points in m-dimensional space, the sum of the squared distances between these n points always equals n times the sum of the squared distances between the n points and their centroid.

If we let $\mathbf{x}_j$ be the column vector containing the coordinates of point j (j = 1, ..., n) in m-dimensional space, and $\mathbf{z}$ the column vector consisting of the coordinates of the centroid of the n points, that is,

$$\mathbf{z} = (1/n) \sum_{j=1}^{n} \mathbf{x}_j,$$

then Theorem 1 can be expressed algebraically as follows:

$$\sum_{j<k} (\mathbf{x}_j - \mathbf{x}_k)'(\mathbf{x}_j - \mathbf{x}_k) = n \sum_{j=1}^{n} (\mathbf{x}_j - \mathbf{z})'(\mathbf{x}_j - \mathbf{z}) = n \sum_{j=1}^{n} \mathbf{x}_j'\mathbf{x}_j - n^2\, \mathbf{z}'\mathbf{z}.$$

A straightforward application of Theorem 1 in the context of GPA is illustrated in Figure 2.3, which contains the plot of four two-dimensional configurations and their corresponding centroids. The coordinates of all points in Figure 2.3 are given in Table 2.1. The numerical value 99 in Table 2.1 indicates that information about a stimulus in

Table 2.1 Coordinates of stimulus points plotted in Figure 2.3.

stimulus	$\mathbf{X}_1$	$\mathbf{X}_2$	$\mathbf{X}_3$	$\mathbf{X}_4$	$\mathbf{Z}$
1	1 3	-2 3	0 6	-3 8	-1 5
2	99 99	1 5	5 2	3 5	3 4
3	4 -1	99 99	99 99	6 -3	5 -2
4	99 99	1 -4	3 -7	2 -7	2 -6
5	-3 -3	-5 -4	-4.5 -6	-3.5 -7	-4 -5

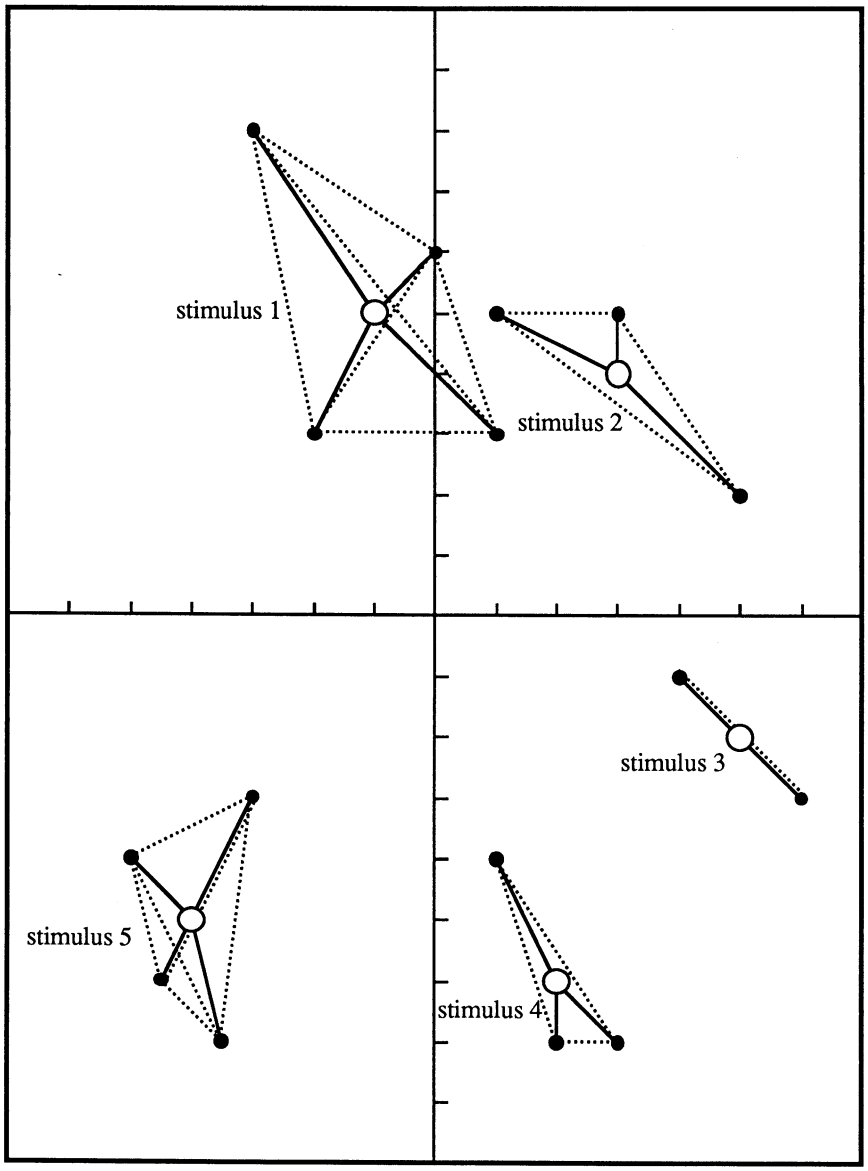

Figure 2.3 Geometry of match for data in Table 2.1.

a configuration is missing. In this example information about stimuli 2 and 4 of configuration $\mathbf{X}_1$ is missing, and stimulus 3 has missing values both in configurations $\mathbf{X}_2$ and $\mathbf{X}_3$.

There are five stimuli in this example and therefore Theorem 1 can be applied five times. The match for stimulus 1 can either be calculated as the sum of squared distances between the four points corresponding to stimulus 1 (dotted lines in Figure 2.3), or as the sum of squared distances between these same four points and their centroid (solid lines in Figure 2.3). For stimulus 1 we have four points to consider because no configuration in this example has missing values in the first row. The match for stimulus 2 may also be evaluated in two ways: either by summing the squared lengths of the dotted lines or by summing the squared lengths of the solid lines for stimulus 2 in Figure 2.3. For stimulus 2 there are only three points to consider since information about this stimulus is missing in configuration $\mathbf{X}_1$. In the same manner the remaining three stimulus points in Figure 2.3 can be evaluated using either match criterion. If we compare the two criteria for each stimulus of the data plotted in Figure 2.3 separately, their ratio is, of course, completely in agreement with Theorem 1. For stimulus 2, for instance, the ratio of the sum of squared lengths of the solid lines and the sum of squared lengths of the dotted lines is 14/42 = 1/3. In general, the ratio between the two criteria will always be $(1/n_i)$, where n_i is the number of non-missing rows in the n configurations $\mathbf{X}_j$ for stimulus i (i = 1, ..., p).

We now address the question how to express these two equivalent match criteria in matrix notation. Given n configurations $\mathbf{X}_j$ (j = 1, ..., n) of order (p x m), let $\mathbf{M}_j$ be the diagonal matrix of order (p x p) with ones on the diagonal if the corresponding rows in $\mathbf{X}_j$ are not missing, and zeroes everywhere else. For instance, $\mathbf{M}_1$ corresponding to $\mathbf{X}_1$ in Table 2.1 is

$$\mathbf{M}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Matrix $\mathbf{M}_j$ is always symmetric and idempotent, and if a configuration j has no missing data $\mathbf{M}_j$ is equal to the identity matrix. In matrix notation the sum of squared lengths of the solid lines in Figure 2.3 is equal to

$$\sum_{j=1}^{n} \text{tr } (\mathbf{X}_j - \mathbf{Z})'\mathbf{M}_j(\mathbf{X}_j - \mathbf{Z}), \tag{2.1}$$

where

$$\mathbf{Z} = (\sum_{j=1}^{n} \mathbf{M}_j)^{-1}(\sum_{j=1}^{n} \mathbf{M}_j\mathbf{X}_j) \tag{2.2}$$

is the (p x m) matrix of centroids. The matrix expression for the sum of squared lengths of the dotted lines in Figure 2.3 is

$$\sum_{j<k} \text{tr } (\mathbf{X}_j - \mathbf{X}_k)'\mathbf{M}_j\mathbf{M}_k(\mathbf{X}_j - \mathbf{X}_k). \tag{2.3}$$

The two criteria (2.1) and (2.3) become identical by correcting them for the ratios $(1/n_i)$ that we discussed above. These ratios are equal to the diagonal of $(\sum_{j=1}^{n} \mathbf{M}_j)^{-1}$.

Theorem 2. The generalization of Theorem 1 for n configurations $\mathbf{X}_j$ containing missing data is

$$\sum_{j=1}^{n} \text{tr } (\mathbf{X}_j - \mathbf{Z})'\mathbf{M}_j(\mathbf{X}_j - \mathbf{Z}) = \sum_{j<k} \text{tr } (\mathbf{X}_j - \mathbf{X}_k)'\mathbf{M}_j\mathbf{M}_k(\sum_{j=1}^{n} \mathbf{M}_j)^{-1}(\mathbf{X}_j - \mathbf{X}_k), \tag{2.4}$$

with $\mathbf{Z}$ as defined in (2.2).

Proof. Expanding the right side of equation (2.4) gives

$$\sum_{j<k} \text{tr } (\mathbf{X}_j - \mathbf{X}_k)'\mathbf{M}_j\mathbf{M}_k(\sum_{j=1}^{n} \mathbf{M}_j)^{-1}(\mathbf{X}_j - \mathbf{X}_k)$$

$$= \sum_{j=1}^{n} \text{tr } \mathbf{X}_j'\mathbf{M}_j(\sum_{i\neq j} \mathbf{M}_i)(\sum_{j=1}^{n} \mathbf{M}_j)^{-1}\mathbf{X}_j - 2\sum_{j<k} \text{tr } \mathbf{X}_j'\mathbf{M}_j\mathbf{M}_k(\sum_{j=1}^{n} \mathbf{M}_j)^{-1}\mathbf{X}_k. \tag{2.5}$$

But since

$$\text{tr } \mathbf{X}_j'\mathbf{M}_j(\sum_{i\neq j} \mathbf{M}_i)(\sum_{j=1}^{n} \mathbf{M}_j)^{-1}\mathbf{X}_j = \text{tr } \mathbf{X}_j'\mathbf{M}_j\mathbf{X}_j - \text{tr } \mathbf{X}_j'\mathbf{M}_j(\sum_{j=1}^{n} \mathbf{M}_j)^{-1}\mathbf{X}_j,$$

expression (2.5) may be written as

$$\sum_{j=1}^{n} \text{tr } \mathbf{X}_j'\mathbf{M}_j\mathbf{X}_j - \sum_{j=1}^{n} \text{tr } \mathbf{X}_j'\mathbf{M}_j(\sum_{j=1}^{n} \mathbf{M}_j)^{-1}\mathbf{X}_j - 2\sum_{j<k} \text{tr } \mathbf{X}_j'\mathbf{M}_j\mathbf{M}_k(\sum_{j=1}^{n} \mathbf{M}_j)^{-1}\mathbf{X}_k$$

$$= \sum_{j=1}^{n} \text{tr } \mathbf{X}_j'\mathbf{M}_j\mathbf{X}_j - \text{tr } (\sum_{j=1}^{n} \mathbf{M}_j\mathbf{X}_j)'(\sum_{j=1}^{n} \mathbf{M}_j)^{-1}(\sum_{j=1}^{n} \mathbf{M}_j\mathbf{X}_j). \tag{2.6}$$

It further follows from (2.2) that

$$(\sum_{j=1}^{n} \mathbf{M}_j\mathbf{X}_j) = (\sum_{j=1}^{n} \mathbf{M}_j)\mathbf{Z}, \tag{2.7}$$

and substitution of (2.7) in (2.6) yields

$$\sum_{j=1}^{n} \text{tr } \mathbf{X}_j'\mathbf{M}_j\mathbf{X}_j - \text{tr } \mathbf{Z}'(\sum_{j=1}^{n} \mathbf{M}_j)\mathbf{Z}$$

$$= \sum_{j=1}^{n} \text{tr } \mathbf{X}_j'\mathbf{M}_j\mathbf{X}_j + \text{tr } \mathbf{Z}'(\sum_{j=1}^{n} \mathbf{M}_j)\mathbf{Z} - 2 \text{ tr } \mathbf{Z}'(\sum_{j=1}^{n} \mathbf{M}_j)\mathbf{Z}$$

$$= \sum_{j=1}^{n} \text{tr } (\mathbf{X}_j - \mathbf{Z})'\mathbf{M}_j(\mathbf{X}_j - \mathbf{Z}),$$

which completes the proof.

It may be noted that thus far nothing has been said about the optimization of the match between the n configurations. Equation (2.4) only shows that, wherever in space the configurations are located with respect to one another and in whatever position, the match in GPA can be measured in two different ways yielding exactly the same result. In order to optimize the match between the n configurations the parameters corresponding to the four admissible transformations illustrated in Figure 2.2 must be inserted into the formula of either the first criterion as in (2.1) or the second criterion as on the right side of (2.4).

Let $\mathbf{1}$ be a (p x 1) vector consisting of ones, $\mathbf{u}_j$ an unknown translation vector of order (m x 1), s_j an unknown uniform scaling factor and $\mathbf{R}_j$ an unknown orthonormal rotation and reflection matrix of order (m x m). Also, let the n unknown translation vectors $\mathbf{u}_j$ be collected in the (m x n) matrix $\mathbf{U}$, the uniform scaling factors in the (n x 1) vector $\mathbf{s}$, and the unknown orthonormal matrices in the (nm x m) supermatrix $\mathbf{R}$, then for the first criterion the following least squares loss function must be minimized for unknown translations, uniform scaling factors and orthonormal transformations:

$$f(\mathbf{U},\mathbf{s},\mathbf{R}) = \sum_{j=1}^{n} \text{tr } [s_j(\mathbf{X}_j - \mathbf{1}\mathbf{u}_j')\mathbf{R}_j - \mathbf{Z}]'\mathbf{M}_j[s_j(\mathbf{X}_j - \mathbf{1}\mathbf{u}_j')\mathbf{R}_j - \mathbf{Z}], \tag{2.8}$$

where $\mathbf{Z}$ is defined as the (p x m) centroid matrix, that is,

$$\mathbf{Z} = (\sum_{j=1}^{n} \mathbf{M}_j)^{-1}(\sum_{j=1}^{n} s_j\mathbf{M}_j(\mathbf{X}_j - \mathbf{1}\mathbf{u}_j')\mathbf{R}_j). \tag{2.9}$$

Analogously, in GPA the following least squares loss function must be minimized if the second criterion is used

$$g(\mathbf{U},\mathbf{s},\mathbf{R}) = \sum_{j<k} \text{tr}\, [\tilde{\mathbf{X}}_j - \tilde{\mathbf{X}}_k]'\mathbf{M}_j\mathbf{M}_k(\sum_{j=1}^{n} \mathbf{M}_j)^{-1}[\tilde{\mathbf{X}}_j - \tilde{\mathbf{X}}_k], \tag{2.10}$$

where $\tilde{\mathbf{X}}_j = s_j(\mathbf{X}_j - \mathbf{1}\mathbf{u}_j')\mathbf{R}_j$ and $\tilde{\mathbf{X}}_k = s_k(\mathbf{X}_k - \mathbf{1}\mathbf{u}_k')\mathbf{R}_k$, and again $\mathbf{u}_j$ and $\mathbf{u}_k$ are unknown translation vectors, s_j and s_k are unknown central dilations, and $\mathbf{R}_j$ and $\mathbf{R}_k$ are unknown orthonormal rotation and reflection matrices.

To be able to distinguish between loss functions (2.8) and (2.10), henceforth we will call (2.8) the *centroid approach* in GPA, since in this approach the amount of loss is measured as the sum of squared distances between corresponding stimulus points and their centroid. Loss function (2.10), on the other hand, from here on will be called the *direct approach*, which name expresses the fact that the loss is directly measured as the sum of squared distances between corresponding points and not through the intermediary of their centroid.

2.2.1 Overview of the solutions

In the literature on GPA without missing data one finds that the minimization of the direct approach (2.10) is more efficient than the minimization of the centroid approach (2.8) (cf. Ten Berge, 1977), because in that case the translation vectors are easily eliminated from (2.10) yielding a function of only two sets of parameters: $\mathbf{s}$ and $\mathbf{R}$. In the case of missing data, however, it seems more fruitful to minimize the centroid approach (2.8), since both the translation vectors *and the centroid configuration* can be eliminated from the latter function, while we have found it impossible to eliminate the translation vectors from (2.10).

At this point, the reader probably wonders: how can one possibly eliminate the centroid configuration $\mathbf{Z}$ from (2.8), when $\mathbf{Z}$ is already defined by (2.9)? First, it is important to note that definition (2.9) has a very special property: *it already yields the*

global minimum of (2.8), when considered as a function of **Z**, for fixed **U**, **s**, and **R**. This immediately follows from the fact that the smallest sum of squared distances between n given points and another variable point is obtained by setting the variable point equal to the centroid of the n given points. Because of this property of (2.9), nothing changes by including the matrix of centroids as a variable in (2.8). Moreover, this creates the possibility to eliminate **U** and **Z** from (2.8), and thus to reduce the unknown parameters in (2.8) to **R** and **s**. Recall that no harm is done by treating **Z** as a variable, since, as will be proved below, at the optimum of the centroid loss function it still satisfies (2.9), even after the elimination of **U** from (2.8).

Therefore, in section 2.3 we completely develop the centroid approach instead of the direct approach. In section 2.3.1 we first of all determine the optimal translation vectors $\mathbf{u}_j$ in (2.8), because this allows us to simplify the loss function considerably. In section 2.3.2 we investigate how the optimal centroid configuration **Z** can be determined in the loss function from which the translation vectors have been eliminated. In section 2.3.3 it is shown how the matrix of centroids **Z** can also be eliminated from the simplified loss function (2.8), and in sections 2.3.4 and 2.3.5 the resulting function is minimized with respect to the only remaining unknowns: the orthonormal transformation matrices and the scaling factors. To this end a number of restrictions will be added to the simplified loss function, so as to guarantee that the optimal rotation and reflection matrices $\mathbf{R}_j$ are orthonormal and that the ensuing algorithm does not converge to the trivial solution $s_j = 0$ for each j.

In section 2.4 an algorithm is presented for GPA of configurations with and without missing data. In section 2.5 a number of measures of fit in GPA are discussed and it is shown how the total sum of squares in GPA may be partitioned in such a way that the relative contributions of individual configurations, stimuli and dimensions to the GPA solution can be identified.

In section 2.6 we discuss the relations between the GPA match criterion and Van de Geer's (1984) match criteria MAXBET, MAXNEAR and MAXDIFF when the configurations contain missing data, and in section 2.7, finally, two examples are presented of the generalized Procrustes analysis of configurations containing missing data, the first example consisting of a constructed data set and the second example of empirical data.

2.3 The centroid approach

2.3.1 Translations

In this section the least squares loss function according to the centroid approach

$$f(\mathbf{U},\mathbf{s},\mathbf{R},\mathbf{Z}) = \sum_{j=1}^{n} \text{tr}\ [s_j(\mathbf{X}_j - \mathbf{1}\mathbf{u}_j')\mathbf{R}_j - \mathbf{Z}]'\mathbf{M}_j[s_j(\mathbf{X}_j - \mathbf{1}\mathbf{u}_j')\mathbf{R}_j - \mathbf{Z}] \qquad (2.8)$$

is minimized with respect to the unrestricted translation vectors $\mathbf{u}_j$ for fixed $\mathbf{s}$, $\mathbf{R}$, and $\mathbf{Z}$. It may be noted that $\mathbf{Z}$ has now been included in the loss function as a variable. Defining $\mathbf{A}_j = s_j\mathbf{X}_j\mathbf{R}_j - \mathbf{Z}$, and only considering one particular translation vector $\mathbf{u}_j$, function (2.8) can be rewritten as

$$f(\mathbf{u}_j) = s_j^2\mathbf{u}_j'\mathbf{u}_j\mathbf{1}'\mathbf{M}_j\mathbf{1} - 2\ s_j\mathbf{u}_j'\mathbf{R}_j\mathbf{A}_j'\mathbf{M}_j\mathbf{1} + d_j = c_j^2\ \mathbf{u}_j'\mathbf{u}_j - 2\ c_j\mathbf{u}_j'\mathbf{y}_j + d_j, \quad (2.11)$$

where d_j is a term independent of $\mathbf{u}_j$, $\mathbf{y}_j \equiv \mathbf{R}_j\mathbf{A}_j'\mathbf{M}_j\mathbf{1}/\sqrt{\mathbf{1}'\mathbf{M}_j\mathbf{1}}$, and $c_j \equiv s_j\sqrt{\mathbf{1}'\mathbf{M}_j\mathbf{1}}$. Since

$$f(\mathbf{u}_j) = \|c_j\mathbf{u}_j - \mathbf{y}_j\|^2 + d_j - \mathbf{y}_j'\mathbf{y}_j \ \geq\ d_j - \mathbf{y}_j'\mathbf{y}_j,$$

this lower bound, and hence the unique global minimum, is attained for $c_j\mathbf{u}_j - \mathbf{y}_j = \mathbf{0}$, and thus for

$$\mathbf{u}_j = c_j^{-1}\mathbf{y}_j. \qquad (2.12)$$

Re-expressing (2.12) in the original terms yields

$$\mathbf{u}_j = \frac{(s_j\mathbf{X}_j - \mathbf{Z}\mathbf{R}'_j)'\mathbf{M}_j\mathbf{1}}{s_j\mathbf{1}'\mathbf{M}_j\mathbf{1}}. \qquad (2.13)$$

Substitution of (2.13) in $\{s_j(\mathbf{X}_j - \mathbf{1}\mathbf{u}_j')\mathbf{R}_j - \mathbf{Z}\}$, which is a part of (2.8), gives

$$s_j(\mathbf{X}_j - \frac{\mathbf{1}\mathbf{1}'\mathbf{M}_j(s_j\mathbf{X}_j - \mathbf{Z}\mathbf{R}'_j)}{s_j\ \mathbf{1}'\mathbf{M}_j\mathbf{1}})\mathbf{R}_j - \mathbf{Z} = \mathbf{J}_j(s_j\mathbf{X}_j\mathbf{R}_j - \mathbf{Z}),$$

with $\mathbf{J}_j = \mathbf{I} - \{(\mathbf{1}\mathbf{1}'\mathbf{M}_j)/(\mathbf{1}'\mathbf{M}_j\mathbf{1})\}$.

Hence, the complete loss function can be written as

$$f(\mathbf{s},\mathbf{R},\mathbf{Z}) = \sum_{j=1}^{n} \text{tr } (s_j\mathbf{X}_j\mathbf{R}_j - \mathbf{Z})'\mathbf{J}_j'\mathbf{M}_j\mathbf{J}_j(s_j\mathbf{X}_j\mathbf{R}_j - \mathbf{Z}). \tag{2.14}$$

Result (2.14) implies that, if one wants to match n configurations according to the centroid approach in GPA, the translation problem is simply taken care of by centering the non-missing elements in each column of $\mathbf{X}_j$ *and* the corresponding elements in each column of $\mathbf{Z}$ on the origin of m-dimensional space. In the case of missing data, the geometrical implication of (2.14) is that the matrix of centroids $\mathbf{Z}$ is translated optimally, and thus *differently*, for each configuration j. For each stimulus i, the error is no longer measured as the sum of squared distances between the n_i points in the $\mathbf{X}_j$ and one fixed centroid point (as in Figure 2.3), but as the sum of squared distances between the n_i points in the $\mathbf{X}_j$ centered on the origin and a centroid point that is idiosyncratically translated for each j, and therefore optimally 'moved around' in m-dimensional space. The situation in Figure 2.3 is immediately restored, however, by calculating $\mathbf{u}_j$ for each j according to (2.13) and then plotting $s_j\mathbf{M}_j(\mathbf{X}_j - \mathbf{1}\mathbf{u}_j')\mathbf{R}_j$ for each j. In that case $\mathbf{Z}$ is again at the fixed centroid of the optimally transformed configurations $\mathbf{X}_j$, while the amount of loss in (2.8) is identical to the amount of loss in (2.14).

The nice effect of the substitution of (2.13) in (2.8) is that the translation vectors are eliminated from loss function (2.8) and are implicitly optimized in the simplified loss function (2.14).

We end this section by noting that $\mathbf{M}_j\mathbf{J}_j = \mathbf{J}_j'\mathbf{M}_j\mathbf{J}_j$ is symmetric and idempotent, as is readily verified. It is also important to see that $\mathbf{J}_j\mathbf{1} = \mathbf{0}$, showing that $\mathbf{J}_j$ is always a singular matrix. Since the matrix product $\mathbf{M}_j\mathbf{J}_j = \mathbf{J}_j'\mathbf{M}_j\mathbf{J}_j$ occurs very often in this book, in the sequel we will replace it by the (p x p) matrix $\mathbf{C}_j$. This matrix has the effect of centering the non-missing elements in each column of configuration $\mathbf{X}_j$ *and* the corresponding elements in each column of $\mathbf{Z}$ on the origin. The loss function from which the translation vectors have been eliminated can thus be written as

$$f(\mathbf{s},\mathbf{R},\mathbf{Z}) = \sum_{j=1}^{n} \text{tr } (s_j\mathbf{X}_j\mathbf{R}_j - \mathbf{Z})'\mathbf{C}_j(s_j\mathbf{X}_j\mathbf{R}_j - \mathbf{Z}), \tag{2.15}$$

and henceforth this simplified loss function is used to determine the remaining unknown parameters.

2.3.2 The configuration of centroids Z

In section 2.2 we showed how to calculate the coordinates of the centroids of corresponding points in loss function (2.8) by using (2.9), and discussed that (2.9) also yields the global minimum of (2.8) if **Z** is treated as a variable. After elimination of the translation vectors from (2.8), however, (2.9) only optimizes (2.15) if the optimal translation vectors are calculated using (2.13), and then substituted in (2.9). But this is not what we are after, of course, since the latter procedure would re-introduce the translation vectors that we just eliminated from (2.8). Hence, in this section we investigate the problem of determining the matrix of centroids **Z** optimizing (2.15) without taking recourse to (2.13).

Since in this problem we consider **s** and **R** to be fixed, let $\mathbf{A}_j = s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j$. Also let the matrices $\mathbf{A}_j$ be collected in the supermatrix **A** of order (np x m), and let the centering matrices $\mathbf{C}_j$ be collected in the supermatrix **B** of order (np x p). Then loss function (2.15) may be rewritten as

$$f(\mathbf{Z}) = \text{tr}\,(\mathbf{A} - \mathbf{BZ})'(\mathbf{A} - \mathbf{BZ}). \tag{2.16}$$

To minimize (2.16) with respect to **Z** is the classical multivariate multiple regression problem. The general solution of this problem is well-known, and follows from the normal equations corresponding to (2.16):

$$\mathbf{B}'\mathbf{BZ} = \mathbf{B}'\mathbf{A}. \tag{2.17}$$

Since the matrices $\mathbf{C}_j$ are symmetric and idempotent, re-expressing (2.17) in the original matrices and scalars yields

$$\mathbf{CZ} = \sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j, \tag{2.18}$$

where $\mathbf{C} \equiv (\sum \mathbf{C}_j)$. Hence, the global minimum of (2.15) with respect to **Z** is attained for

$$\mathbf{Z} = \mathbf{C}^{-}(\sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j), \tag{2.19}$$

where $\mathbf{C}^{-}$ is the Moore-Penrose generalized inverse of the sum of the $\mathbf{C}_j$'s.

The reason that a generalized inverse must be determined in order to solve (2.18) for unknown $\mathbf{Z}$ is that

$$\mathbf{C1} = \sum_{j=1}^{n} \mathbf{M}_j\mathbf{J}_j\mathbf{1} = \sum_{j=1}^{n} \mathbf{M}_j\mathbf{0} = \mathbf{0}$$

(see also section 2.3.1), indicating that the sum of the $\mathbf{C}_j$'s has no proper inverse. The Moore-Penrose inverse is calculated as follows. Because $\mathbf{C}_j$ is always a symmetric matrix, the sum of the $\mathbf{C}_j$'s is also symmetric. Therefore, let

$$\mathbf{C} = \mathbf{Q}\mathbf{\Phi}\mathbf{Q}'$$

be an eigenvalue-eigenvector decomposition of the sum of the centring matrices. Matrix $\mathbf{\Phi}$ does not contain any negative eigenvalues, since the sum of the $\mathbf{C}_j$'s is positive semi-definite, each $\mathbf{C}_j$ being positive semi-definite. Let $\mathbf{\Phi}^*$ be the submatrix of $\mathbf{\Phi}$ only containing the positive eigenvalues, and let $\mathbf{Q}^*$ be the matrix of eigenvectors corresponding to $\mathbf{\Phi}^*$. The Moore-Penrose generalized inverse is then determined by

$$\mathbf{C}^- = \mathbf{Q}^*\mathbf{\Phi}^{*-1}\mathbf{Q}^{*\prime}.$$

(see, e.g., Van de Geer, 1986).

Geometrically, the matrix $\mathbf{Z}$ calculated with (2.19) does *not* contain the simple averages of corresponding points in the optimally transformed $s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j$'s. This is related to the fact that in (2.15) matrix $\mathbf{Z}$ is optimally translated but this translation is different for each j. Calculating the optimal translation vectors (2.13) for each j after (2.19) has been computed, however, and substituting these optimal $\mathbf{u}_j$'s in (2.9) guarantees that $\mathbf{Z}$ computed according to (2.19) is exactly equal to $\mathbf{Z}$ calculated using (2.9). We will now prove that this is indeed the case.

It follows from (2.9) that

$$(\sum_{j=1}^{n} \mathbf{M}_j)\mathbf{Z} = \sum_{j=1}^{n} s_j\mathbf{M}_j(\mathbf{X}_j - \mathbf{1}\mathbf{u}_j')\mathbf{R}_j,$$

and substitution of (2.13), the optimal translation vectors, in the latter expression gives

2.3.3 The elimination of Z

The loss function corresponding to the centroid approach from which the translation vectors have been eliminated is (2.15), that is,

$$f(\mathbf{s},\mathbf{R},\mathbf{Z}) = \sum_{j=1}^{n} \text{tr } (s_j\mathbf{X}_j\mathbf{R}_j - \mathbf{Z})'\mathbf{C}_j(s_j\mathbf{X}_j\mathbf{R}_j - \mathbf{Z}),$$

and the optimum of this function for unknown $\mathbf{Z}$ is found where

$$\mathbf{Z} = \mathbf{C}^{-}(\sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j). \tag{2.19}$$

The key procedure to change (2.15) into a loss function that does not explicitly contain the unknown centroid configuration is to substitute (2.19) into (2.15):

$$f(\mathbf{s},\mathbf{R}) =$$

$$\sum_{j=1}^{n} \text{tr } [s_j\mathbf{X}_j\mathbf{R}_j - \mathbf{C}^{-}(\sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j)]'\mathbf{C}_j[s_j\mathbf{X}_j\mathbf{R}_j - \mathbf{C}^{-}(\sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j)] =$$

$$\sum_{j=1}^{n} s_j^2 \text{ tr } \mathbf{R}_j'\mathbf{X}_j'\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j - \text{tr } (\sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j)'\mathbf{C}^{-}(\sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j). \tag{2.22}$$

If all configurations are complete $\mathbf{C}_j = \mathbf{I} - (\mathbf{11'/1'1}) \equiv \mathbf{J}$ for each j, and (2.22) may be written as

$$f(\mathbf{s},\mathbf{R}) = \sum_{j=1}^{n} s_j^2 \text{ tr } \mathbf{R}_j'\mathbf{X}_j'\mathbf{J}\mathbf{X}_j\mathbf{R}_j - \text{tr } (\sum_{j=1}^{n} s_j\mathbf{J}\mathbf{X}_j\mathbf{R}_j)'(n\mathbf{J})^{-}(\sum_{j=1}^{n} s_j\mathbf{J}\mathbf{X}_j\mathbf{R}_j)$$

$$= \sum_{j=1}^{n} s_j^2 \text{ tr } \mathbf{R}_j'\mathbf{X}_j'\mathbf{J}\mathbf{X}_j\mathbf{R}_j - (1/n) \text{ tr } (\sum_{j=1}^{n} s_j\mathbf{J}\mathbf{X}_j\mathbf{R}_j)'(\sum_{j=1}^{n} s_j\mathbf{J}\mathbf{X}_j\mathbf{R}_j)$$

$$= (1/n) \sum_{j<k} \text{tr } (s_j\mathbf{J}\mathbf{X}_j\mathbf{R}_j - s_k\mathbf{J}\mathbf{X}_k\mathbf{R}_k)'(s_j\mathbf{J}\mathbf{X}_j\mathbf{R}_j - s_k\mathbf{J}\mathbf{X}_k\mathbf{R}_k).$$

This is the well-known GPA format for the direct approach without translations if all configurations are complete (see, e.g., Gower, 1975; Ten Berge, 1977), and, again, confirms the equivalence between the centroid and the direct approach in GPA.

With (2.22) we are in a position to determine the optimal orthonormal transformations and uniform scaling factors without the slackening effect of $\mathbf{Z}$, and therefore to generalize Ten Berge's (1977) efficient method to the case of missing

$$(\sum_{j=1}^{n} \mathbf{M}_j)\mathbf{Z} = \sum_{j=1}^{n} s_j\mathbf{M}_j(\mathbf{X}_j - \frac{\mathbf{11'M}_j(s_j\mathbf{X}_j - \mathbf{Z}\mathbf{R}_j')}{s_j\mathbf{1'M}_j\mathbf{1}})\mathbf{R}_j$$

$$= \sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j + (\sum_{j=1}^{n} \mathbf{M}_j\mathbf{11'M}_j/\mathbf{1'M}_j\mathbf{1})\mathbf{Z}. \qquad (2.20)$$

Since it is always true that

$$\mathbf{CZ} = [\sum_{j=1}^{n} \mathbf{M}_j(\mathbf{I} - \mathbf{11'M}_j/\mathbf{1'M}_j\mathbf{1})]\mathbf{Z} = (\sum_{j=1}^{n} \mathbf{M}_j)\mathbf{Z} - (\sum_{j=1}^{n} \mathbf{M}_j\mathbf{11'M}_j/\mathbf{1'M}_j\mathbf{1})\mathbf{Z},$$

the following also holds:

$$(\sum_{j=1}^{n} \mathbf{M}_j\mathbf{11'M}_j/\mathbf{1'M}_j\mathbf{1})\mathbf{Z} = (\sum_{j=1}^{n} \mathbf{M}_j)\mathbf{Z} - \mathbf{CZ}. \qquad (2.21)$$

Substitution of (2.21) in (2.20) gives

$$(\sum_{j=1}^{n} \mathbf{M}_j)\mathbf{Z} = \sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j + (\sum_{j=1}^{n} \mathbf{M}_j)\mathbf{Z} - \mathbf{CZ},$$

and therefore

$$(\sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j) = \mathbf{CZ}.$$

This completes the proof of the equality of **Z** in (2.19) and (2.9) for optimal translation vectors (2.13).

Due to this equality it also follows that, although **Z** in (2.19) does not contain the averages of corresponding points in the $s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j$, it does contain the averages in the $s_j\mathbf{M}_j(\mathbf{X}_j - \mathbf{1u}_j')\mathbf{R}_j$ after updating **U**, indicating that Theorem 2 still holds after elimination of the translation vectors from (2.8), and with matrix **Z** treated as an unknown variable.

data. In the next two sections (2.22) is minimized with respect to the remaining unknowns **R** and **s**.

2.3.4 Orthonormal transformations

The minimization of loss function (2.22), that is, of

$$f(\mathbf{s},\mathbf{R}) = \sum_{j=1}^{n} s_j^2 \operatorname{tr} \mathbf{R}_j'\mathbf{X}_j'\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j - \operatorname{tr} (\sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j)'\mathbf{C}^{-}(\sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j)$$

with respect to the unknown matrix **R** for fixed **s** is equivalent to the maximization of

$$g(\mathbf{R}) = \operatorname{tr} (\sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j)'\mathbf{C}^{-}(\sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j) \tag{2.23}$$

given the constraint

$$\mathbf{R}_j'\mathbf{R}_j = \mathbf{R}_j\mathbf{R}_j' = \mathbf{I}_m, \qquad \text{for } j = 1, \ldots, n. \tag{2.24}$$

If we let $\mathbf{A}_j = s_j\mathbf{C}_j\mathbf{X}_j$ and consider one particular $\mathbf{R}_j$, function (2.23) can be written as

$$g(\mathbf{R}_j) = \operatorname{tr} \mathbf{R}_j'\mathbf{A}_j'\mathbf{C}^{-}\mathbf{A}_j\mathbf{R}_j + 2 \operatorname{tr} \mathbf{R}_j'\mathbf{A}_j'\mathbf{C}^{-}(\sum_{i\neq j} \mathbf{A}_i\mathbf{R}_i) + d_j, \tag{2.25}$$

where d_j is a term independent of $\mathbf{R}_j$. Therefore, the maximization of (2.25) is equivalent to the maximization of

$$h(\mathbf{R}_j) = \operatorname{tr} \mathbf{R}_j'\mathbf{A}_j'\mathbf{C}^{-}(\sum_{i\neq j} \mathbf{A}_i\mathbf{R}_i) \tag{2.26}$$

under constraint (2.24). Let

$$\mathbf{A}_j'\mathbf{C}^{-}(\sum_{i\neq j} \mathbf{A}_i\mathbf{R}_i) = \mathbf{P}_j\mathbf{\Phi}_j\mathbf{Q}_j' \tag{2.27}$$

be a singular value decomposition of $\mathbf{A}_j'\mathbf{C}^{-}(\sum_{i\neq j} \mathbf{A}_i\mathbf{R}_i)$. Substitution of (2.27) in (2.26) gives

$$h(\mathbf{R}_j) = \operatorname{tr} \mathbf{R}_j'\mathbf{P}_j\mathbf{\Phi}_j\mathbf{Q}_j' = \operatorname{tr} \mathbf{Q}_j'\mathbf{R}_j'\mathbf{P}_j\mathbf{\Phi}_j = \operatorname{tr} \mathbf{B}_j\mathbf{\Phi}_j,$$

where $\mathbf{B}_j = \mathbf{Q}_j'\mathbf{R}_j'\mathbf{P}_j$ is an orthonormal matrix of order (m x m), being the product of orthonormal matrices $\mathbf{Q}_j'$, $\mathbf{R}_j'$ and $\mathbf{P}_j$. Therefore, it is always true that

$$h(\mathbf{R}_j) = \text{tr } \mathbf{B}_j\mathbf{\Phi}_j \leq \text{tr } \mathbf{\Phi}_j, \tag{2.28}$$

and this upper limit is attained for

$$\mathbf{B}_j = \mathbf{Q}_j'\mathbf{R}_j'\mathbf{P}_j = \mathbf{I}_m,$$

and thus for

$$\mathbf{R}_j = \mathbf{P}_j\mathbf{Q}_j'. \tag{2.29}$$

Solution (2.29) obviously satisfies (2.24), and holds regardless of singularity of (2.27). Inequality (2.28) is due to Kristof (1970) and, in a more general form, Ten Berge (1983).

Since (2.23) can not be solved for the n $\mathbf{R}_j$'s simultaneously, an alternative least squares algorithm must be used where (2.23) is consecutively being maximized for j = 1, 2, ..., n and $\mathbf{R}$ is updated after each step. This process is repeated until n steps jointly fail to raise (2.23) above some threshold value.

2.3.5 Scaling constants

In this section we will show how to minimize (2.22) with respect to the unknown uniform scaling constants. When we are dealing with incomplete configurations it is essential to determine optimal scaling constants in GPA. We will illustrate this point with a simple example. The example consists of only two configurations in two dimensions with coordinates given on the left of Table 2.2 and drawn on the left of Figure 2.4. Configuration $\mathbf{X}_1$ has four stimuli that form the vertices of a square. Configuration $\mathbf{X}_2$ is identical to configuration $\mathbf{X}_1$, except that information about the second stimulus is assumed to be missing in $\mathbf{X}_2$. Both configurations have been centered on the origin. Notice that the distances between stimuli 1, 3, and 4 in $\mathbf{C}_1\mathbf{X}_1$ are identical to the distances between stimuli 1, 3, and 4 in $\mathbf{C}_2\mathbf{X}_2$. Loosely speaking the two configurations $\mathbf{C}_1\mathbf{X}_1$ and $\mathbf{C}_2\mathbf{X}_2$ are 'of the same size'. Translations and orthonormal transformations of these two configurations therefore suffice to obtain a perfect solution. In practice this situation will rarely occur, however. Usually the

Table 2.2 Coordinates of the configurations in Figure 2.4 before (left) and after (right) unit normalization.

stimulus	$\mathbf{C}_1\mathbf{X}_1$	$\mathbf{C}_2\mathbf{X}_2$	$\mathbf{C}_1\tilde{\mathbf{X}}_1$	$\mathbf{C}_2\tilde{\mathbf{X}}_2$
1	1 0	1 -1/3	1/2 0	.612 -.204
2	0 -1	99 99	0 -1/2	99 99
3	-1 0	-1 -1/3	-1/2 0	-.612 -.204
4	0 1	0 2/3	0 1/2	0 .408

situation will be as is illustrated on the right of Figure 2.4. There, the same two configurations are drawn after having been unit normalized (i.e., $\text{tr}\,\tilde{\mathbf{X}}_1\mathbf{C}_1\tilde{\mathbf{X}}_1 = \text{tr}\,\tilde{\mathbf{X}}_2\mathbf{C}_2\tilde{\mathbf{X}}_2 = 1$). The coordinates of the unit normalized configurations are given at the right side of Table 2.2. Due to the unit normalization, the distances between stimuli 1, 3, and 4 in $\mathbf{C}_1\tilde{\mathbf{X}}_1$ are no longer identical to the distances between stimuli 1, 3, and 4 in $\mathbf{C}_2\tilde{\mathbf{X}}_2$. Obviously, in this case it is impossible to find a perfect solution if the only admissible transformations are translations, rotations and reflections. This shows that, if the configurations contain missing values, and if no care is taken to correct for differences in 'size' between the configurations, neither the loss function according to the centroid approach nor the loss function using the direct approach is in general bounded from below by zero (i.e., a perfect solution is never found, even if it

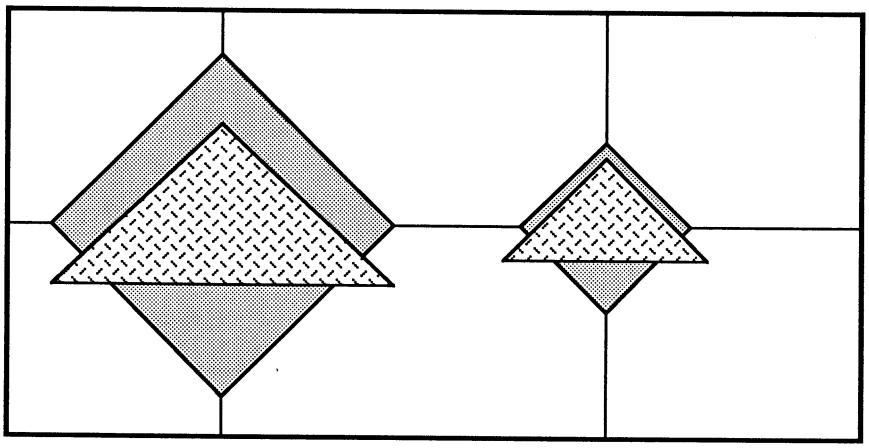

Figure 2.4 Configurations $\mathbf{X}_1$ and $\mathbf{X}_2$ from Table 2.2, both before (left) and after (right) unit normalization.

exists). In the case of missing data, therefore, it is mandatory to implement scaling factors in the corresponding loss functions.

Some kind of restriction must be imposed upon the scaling factors that minimize (2.22) because the ensuing algorithm otherwise will converge to the trivial solution $\mathbf{s} = \mathbf{0}$. The scaling factors will be restricted to satisfy

$$\sum_{j=1}^{n} s_j^2 \operatorname{tr} \mathbf{X}_j'\mathbf{C}_j\mathbf{X}_j = \sum_{j=1}^{n} \operatorname{tr} \mathbf{X}_j'\mathbf{C}_j\mathbf{X}_j. \tag{2.30}$$

This side condition is a generalization of the constraint that Gower (1975) used in the case without missing data. Restriction (2.30) expresses that the sum of squares about the origin of the scaled configurations must remain equal to the sum of squares of the original configurations centered on the origin.

Considering $\mathbf{R}$ as fixed, and letting $\mathbf{A}_j = \mathbf{C}_j\mathbf{X}_j\mathbf{R}_j$, the loss function to be minimized is

$$f(\mathbf{s}) = \sum_{j=1}^{n} s_j^2 \operatorname{tr} \mathbf{A}_j'\mathbf{A}_j - \operatorname{tr} (\sum_{j=1}^{n} s_j\mathbf{A}_j)'\mathbf{C}^{-}(\sum_{j=1}^{n} s_j\mathbf{A}_j). \tag{2.22}$$

Because the scaling constants are constrained to satisfy

$$\sum_{j=1}^{n} s_j^2 \operatorname{tr} \mathbf{A}_j'\mathbf{A}_j = \sum_{j=1}^{n} \operatorname{tr} \mathbf{A}_j'\mathbf{A}_j, \tag{2.30}$$

the minimization of (2.22) for unknown scaling constants is equivalent to the maximization of

$$h(\mathbf{s}) = \operatorname{tr} (\sum_{j=1}^{n} s_j\mathbf{A}_j)'\mathbf{C}^{-}(\sum_{j=1}^{n} s_j\mathbf{A}_j). \tag{2.31}$$

Let matrix $\mathbf{Y}$ of order (n x n) be defined as

$$\mathbf{Y} = \begin{bmatrix} \operatorname{tr} \mathbf{A}_1'\mathbf{C}^{-}\mathbf{A}_1 & \operatorname{tr} \mathbf{A}_1'\mathbf{C}^{-}\mathbf{A}_2 & \dots & \operatorname{tr} \mathbf{A}_1'\mathbf{C}^{-}\mathbf{A}_n \\ \operatorname{tr} \mathbf{A}_2'\mathbf{C}^{-}\mathbf{A}_1 & \operatorname{tr} \mathbf{A}_2'\mathbf{C}^{-}\mathbf{A}_2 & \dots & \operatorname{tr} \mathbf{A}_2'\mathbf{C}^{-}\mathbf{A}_n \\ \vdots & \vdots & & \vdots \\ \operatorname{tr} \mathbf{A}_n'\mathbf{C}^{-}\mathbf{A}_1 & \operatorname{tr} \mathbf{A}_n'\mathbf{C}^{-}\mathbf{A}_2 & \dots & \operatorname{tr} \mathbf{A}_n'\mathbf{C}^{-}\mathbf{A}_n \end{bmatrix}. \tag{2.32}$$

Also, define matrix $\mathbf{W}$ of order (n x n) as

$$\mathbf{W} = \begin{bmatrix} \text{tr } \mathbf{A}_1'\mathbf{A}_1 & 0 & \dots & 0 \\ 0 & \text{tr } \mathbf{A}_2'\mathbf{A}_2 & \dots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & \text{tr } \mathbf{A}_n'\mathbf{A}_n \end{bmatrix}. \tag{2.33}$$

Finally, let

$$\mathbf{W}^{-1/2}\mathbf{Y}\mathbf{W}^{-1/2} = \mathbf{P}\boldsymbol{\Phi}\mathbf{P}' \tag{2.34}$$

be an eigenvector-eigenvalue decomposition of $\mathbf{W}^{-1/2}\mathbf{Y}\mathbf{W}^{-1/2}$, and let $\mathbf{p}_1$ be the first column of $\mathbf{P}$.

Theorem 3. The scaling factors $\mathbf{s}$ that maximize (2.31) under constraint (2.30) are

$$\mathbf{s} = (\text{tr } \mathbf{W})^{1/2}\mathbf{W}^{-1/2}\mathbf{p}_1, \tag{2.35}$$

where $\mathbf{W}$ and $\mathbf{p}_1$ are defined in (2.33) and (2.34), respectively.

Proof. Function (2.31) can be written as

$$h(\mathbf{s}) = \mathbf{s}'\mathbf{Y}\mathbf{s}, \tag{2.36}$$

which must be maximized under constraint (2.30) that can be written as

$$\mathbf{s}'\mathbf{W}\mathbf{s} = \text{tr } \mathbf{W}. \tag{2.37}$$

Since $\mathbf{W}^{1/2}\mathbf{W}^{-1/2} = \mathbf{I}_n$, expression (2.36) may be rewritten as

$$h(\mathbf{s}) = \mathbf{s}'\mathbf{W}^{1/2}\mathbf{W}^{-1/2}\mathbf{Y}\mathbf{W}^{-1/2}\mathbf{W}^{1/2}\mathbf{s}. \tag{2.38}$$

Substitution of (2.34) in (2.38) gives

$$h(\mathbf{s}) = \mathbf{s}'\mathbf{W}^{1/2}\mathbf{P}\boldsymbol{\Phi}\mathbf{P}'\mathbf{W}^{1/2}\mathbf{s}, \tag{2.39}$$

and if we let $\mathbf{u} = \mathbf{P}'\mathbf{W}^{1/2}\mathbf{s}$, expression (2.39) simplifies into

$$h(\mathbf{s}) = \mathbf{u}'\boldsymbol{\Phi}\mathbf{u}.$$

But, since

$$h(\mathbf{s}) = \mathbf{u}'\mathbf{\Phi u} \leq \phi_1 \mathbf{u}'\mathbf{u},$$

and because

$$\mathbf{u}'\mathbf{u} = \mathbf{s}'\mathbf{W}^{1/2}\mathbf{P}\mathbf{P}'\mathbf{W}^{1/2}\mathbf{s} = \mathbf{s}'\mathbf{W}\mathbf{s} = \text{tr } \mathbf{W},$$

the following always holds:

$$h(\mathbf{s}) = \mathbf{u}'\mathbf{\Phi u} \leq \phi_1(\text{tr } \mathbf{W}), \tag{2.40}$$

whatever the values of the scaling constants are, as long as they satisfy $\mathbf{s}'\mathbf{W}\mathbf{s} = \text{tr } \mathbf{W}$. The upper bound in (2.40) is attained for (2.35) since

$$\begin{aligned} h((\text{tr } \mathbf{W})^{1/2}\mathbf{W}^{-1/2}\mathbf{p}_1) &= [(\text{tr } \mathbf{W})^{1/2}\mathbf{W}^{-1/2}\mathbf{p}_1]'\mathbf{Y}[(\text{tr } \mathbf{W})^{1/2}\mathbf{W}^{-1/2}\mathbf{p}_1] \\ &= (\text{tr } \mathbf{W})\, \mathbf{p}_1'\mathbf{P}\mathbf{\Phi}\mathbf{P}'\mathbf{p}_1 = \phi_1(\text{tr } \mathbf{W}). \end{aligned} \tag{2.41}$$

The optimal scaling constants in (2.35) satisfy constraint (2.37) because

$$\mathbf{s}'\mathbf{W}\mathbf{s} = [(\text{tr } \mathbf{W})^{1/2}\mathbf{W}^{-1/2}\mathbf{p}_1]'\mathbf{W}[(\text{tr } \mathbf{W})^{1/2}\mathbf{W}^{-1/2}\mathbf{p}_1] = \text{tr } \mathbf{W}.$$

This completes the proof.

Concluding, for fixed $\mathbf{R}$ an analytical solution is available for the unknown scaling constants $\mathbf{s}$ which is guaranteed to yield the global optimum of the GPA loss function. To minimize (2.22) with respect to $\mathbf{R}$ and $\mathbf{s}$, however, an iterative procedure must be used. In the next section we present a two-step algorithm for the generalized Procrustes analysis of configurations containing missing data.

2.4 The algorithm

The algorithm for generalized Procrustes analysis of data with and without missing rows can be set up as follows.

1. Initialization step

For each j (j = 1, 2, ..., n) calculate

$$\mathbf{C}_j = \mathbf{M}_j(\mathbf{I} - \frac{\mathbf{1}\mathbf{1}'\mathbf{M}_j}{\mathbf{1}'\mathbf{M}_j\mathbf{1}})$$

of order (p x p) and determine the Moore-Penrose generalized inverse of the sum of the $\mathbf{C}_j$'s as described in section 2.3.2, that is, determine

$$\mathbf{C}^- = (\sum_{j=1}^{n} \mathbf{C}_j)^- .$$

Normalize the n configurations such that

$$\sum_{j=1}^{n} \text{tr } \mathbf{X}_j'\mathbf{C}_j\mathbf{X}_j = n.$$

It may be noted that this normalization is not strictly necessary. Our main reason for using it is that it helps to keep the precision of the GPA algorithm (and of the algorithms to be developed in subsequent chapters) within more or less uniform bounds. Construct the matrix

$$\mathbf{W}^{-1/2} = \begin{bmatrix} 1/\sqrt{\text{tr } \mathbf{X}_1'\mathbf{C}_1\mathbf{X}_1} & 0 & \dots & 0 \\ 0 & 1/\sqrt{\text{tr } \mathbf{X}_2'\mathbf{C}_2\mathbf{X}_2} & \dots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & 1/\sqrt{\text{tr } \mathbf{X}_n'\mathbf{C}_n\mathbf{X}_n} \end{bmatrix} .$$

Initialize all n scaling constants on $s_j = 1$, and set all n initial $\mathbf{R}_j$'s equal to $\mathbf{I}_m$.

2. Orthonormal transformation step

Determine for each j a new orthonormal matrix $\mathbf{R}_j$ using the singular value decomposition (see section 2.3.4)

$$\mathbf{X}_j'\mathbf{C}_j\mathbf{C}^{-}(\sum_{i\neq j} s_i\mathbf{C}_i\mathbf{X}_i\mathbf{R}_i) = \mathbf{P}_j\mathbf{\Phi}_j\mathbf{Q}_j'. \tag{2.27}$$

In practice, the orthonormal transformation procedure in MATCHALS has been programmed as follows. We use the Jacobi method to calculate the eigenvalue-eigenvector decomposition of

$$\mathbf{A}_j'\mathbf{A}_j = \mathbf{Q}_j\mathbf{\Phi}_j^2\mathbf{Q}_j', \tag{2.42}$$

where

$$\mathbf{A}_j \equiv \mathbf{X}_j'\mathbf{C}_j\mathbf{C}^{-}(\sum_{i\neq j} s_i\mathbf{C}_i\mathbf{X}_i\mathbf{R}_i).$$

If $\mathbf{A}_j$ has full column rank (i.e., if $\mathbf{\Phi}_j^2$ has m positive eigenvalues), then the left singular vectors $\mathbf{P}_j$ are computed as

$$\mathbf{P}_j = \mathbf{A}_j\mathbf{Q}_j\mathbf{\Phi}_j^{-1}, \tag{2.43}$$

and the optimal orthonormal $\mathbf{R}_j$ is computed as $\mathbf{R}_j = \mathbf{P}_j\mathbf{Q}_j'$.

If $\mathbf{A}_j$ has rank $r < m$ (i.e., if the diagonal of $\mathbf{\Phi}_j^2$ contains one or more zeroes), $\mathbf{P}_j^*$ of order (m x r) is calculated as

$$\mathbf{P}_j^* = \mathbf{A}_j\mathbf{Q}_j^*\mathbf{\Phi}_j^{*-1}, \tag{2.44}$$

where $\mathbf{\Phi}_j^*$ is the (r x r) submatrix of $\mathbf{\Phi}_j$ only containing the positive singular values, and $\mathbf{Q}_j^*$ is the (m x r) matrix of eigenvectors corresponding to $\mathbf{\Phi}_j^*$. Then (m – r) orthonormal columns collected in the (m x (m – r)) matrix $\bar{\mathbf{P}}_j$ must be appended to $\mathbf{P}_j^*$ satisfying $\mathbf{P}_j^{*\prime}\bar{\mathbf{P}}_j = \mathbf{0}$ and $\bar{\mathbf{P}}_j'\bar{\mathbf{P}}_j = \mathbf{I}_{(m-r)}$.

The nullspace $\bar{\mathbf{P}}_j$ of $\mathbf{P}_j^*$ is calculated as follows. Using a random number generator a vector $\mathbf{y}$ of order (m x 1) containing random numbers is constructed. Then the multiple regression of $\mathbf{y}$ on the r columns of $\mathbf{P}_j^*$ is calculated. If we let $\hat{\mathbf{y}}$ be the vector of predicted values on account of the regression equation, the (m x 1) column vector of residuals $(\mathbf{y} - \hat{\mathbf{y}})$ is calculated and unit normalized. This unit normalized vector of

residuals is guaranteed to be orthogonal to all r columns of $\mathbf{P}_j^*$ and is used as the first column of matrix $\bar{\mathbf{P}}_j$. Then a new (m x 1) vector **y** of random numbers is generated, the multiple regression of this new **y** on the r columns of $\mathbf{P}_j^*$ *and* the first column of $\bar{\mathbf{P}}_j$ is executed, and the unit normalized vector of residuals is used as the second column of $\bar{\mathbf{P}}_j$, etc., etc. This procedure is repeated (m – r) times, yielding an (m x (m – r)) orthonormal nullspace $\bar{\mathbf{P}}_j$ of matrix $\mathbf{P}_j^*$. If we let $\mathbf{P}_j = [\mathbf{P}_j^*, \bar{\mathbf{P}}_j]$, that is, matrix $\mathbf{P}_j^*$ of order (m x r) appended with the (m – r) columns of $\bar{\mathbf{P}}_j$, then the optimal orthonormal transformation is $\mathbf{R}_j = \mathbf{P}_j\mathbf{Q}_j'$.

After the calculation of n new orthonormal matrices $\mathbf{R}_j$ following the above procedure, evaluate loss function (2.22).

3. Rescaling step

Set up matrix **Y** of order (p x p) as defined in section 2.3.5, and determine the eigenvalue-eigenvector decomposition of

$$\mathbf{W}^{-1/2}\mathbf{Y}\mathbf{W}^{-1/2}.$$

To this end use the power method, and only calculate the *first* principal component, that is, only calculate

$$\mathbf{W}^{-1/2}\mathbf{Y}\mathbf{W}^{-1/2} = \mathbf{p}_1\phi_1\mathbf{p}_1' + \mathbf{E},$$

where $\mathbf{p}_1$ is the principal eigenvector, ϕ_1 is the largest eigenvalue and matrix **E** of order (n x n) is a matrix of residuals. The reason that we prefer the power method in this step of the algorithm is that it is more efficient than the Jacobi method as the latter always calculates all eigenvectors and eigenvalues (while only the first one is needed). Then calculate new scaling constants as

$$\mathbf{s} = (\sqrt{n})\mathbf{W}^{-1/2}\mathbf{p}_1. \tag{2.35}$$

Evaluate the loss in this iteration by computing

$$f(\mathbf{s},\mathbf{R}) = n - \mathrm{tr}\,(\sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j)'\mathbf{C}^{-}(\sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j), \tag{2.22}$$

and check whether the difference between the loss in this iteration and the loss in the former iteration is smaller than some preset convergence criterion (0.00001, for example). If so, go to step 4, otherwise go to step 2.

4. Results

After the algorithm has converged, determine the centroid configuration using

$$\mathbf{Z} = \mathbf{C}^{-}(\sum_{j=1}^{n} s_j \mathbf{C}_j \mathbf{X}_j \mathbf{R}_j), \tag{2.19}$$

and then calculate the optimal translation vectors $\mathbf{u}_j$ using (2.13). As is easily verified, the whole solution of GPA is unique up to a simultaneous orthonormal transformation of all optimally rotated, reflected and rescaled configurations. For interpretational purposes as well as to make the solution unique the whole solution is rotated to the principal components of $\mathbf{Z}$. Hence, determine the eigenvalue-eigenvector decomposition of

$$\mathbf{Z}'\mathbf{C}\mathbf{Z} = \mathbf{K}\Lambda\mathbf{K}',$$

and print $\mathbf{ZK}$ (i.e., the centroid configuration rotated to its principal components) and $s_j\mathbf{M}_j(\mathbf{X}_j - \mathbf{1}\mathbf{u}_j')\mathbf{R}_j\mathbf{K}$ for each j (i.e., the optimally translated, rotated, reflected and rescaled configurations rotated to the principal components of their centroid configuration). Also print the optimal scaling constants. Finally, print the history of iterations, that is the loss after the orthonormal transformations step and after the rescaling step in each iteration of the algorithm.

This algorithm must converge, although it can not be proved to converge necessarily to the global minimum. Convergence is very fast in practice, and much faster than the first GPA algorithm proposed by Gower (1975). This is due to the fact that in the algorithm proposed in this section the unknowns can be calculated directly, that is, without having to use the centroid configuration as an intermediary.

2.5 Analysis of variation in GPA with missing data

For reasons of simplicity of notation we assume throughout this section that the n configurations have been normalized such that $\sum \text{tr } \mathbf{X}_j'\mathbf{C}_j\mathbf{X}_j = n$.

In their 1978 article Lingoes and Borg discussed two measures of fit in the context of the generalized Procrustes analysis model without missing data. The first measure is the squared correlation between the elements of the optimally transformed $\mathbf{X}_j$ and the elements of $\mathbf{Z}$. In Lingoes and Borg's notation the fit for each j is $r^2(\tilde{\mathbf{X}}_j,\mathbf{Z})$, where $\tilde{\mathbf{X}}_j$ is the optimally transformed configuration $\mathbf{X}_j$. They used this measure to assess the relative contribution of each configuration j to the solution of GPA. As a measure of total fit Lingoes and Borg calculated the mean of these configuration fits, that is

$$\text{total fit} = (1/n) \sum_{j=1}^{n} r^2(\tilde{\mathbf{X}}_j,\mathbf{Z}).$$

Lingoes and Schönemann (1974) proposed this measure of fit in the context of fitting one configuration to another if only two configurations are involved. The advantages of this measure are that it is a symmetric measure and that it is not sensitive to differences in scale of the two configurations.

The second measure of fit that Lingoes and Borg briefly discussed is the squared norm of the centroid configuration $\mathbf{Z}$ (i.e., tr $\mathbf{Z}'\mathbf{Z}$), which can be shown to satisfy

$$0 \leq \text{tr } \mathbf{Z}'\mathbf{Z} \leq 1,$$

and therefore can be used as a measure of total fit. Lingoes and Borg pointed to the fact that, if the $\mathbf{X}_j$'s have been unit normalized, a measure of configuration fit could be derived from this total fit by calculating

$$s_j^2 \text{ tr } \mathbf{Z}'\mathbf{Z}$$

for each j. They also noted that in that case there exists the following relation between the measure of fit based on the squared correlation and the one based on the sum of squares of $\mathbf{Z}$:

$$r^2(\tilde{\mathbf{X}}_j,\mathbf{Z}) = s_j^2 \text{ tr } \mathbf{Z}'\mathbf{Z}.$$

In this section we investigate the relations between the two measures mentioned above, and generalize them to the case of configurations containing missing data. We will show that if the configurations are incomplete the two measures are no longer identical, and argue that in this case the second measure of fit (based on the sum of squares of the centroid configuration **Z**) is more appropriate than the use of squared correlations. At the same time we will generalize Gower's (1975) analysis of variance framework to the case of missing data.

We start by generalizing Lingoes and Borg's second measure of fit to missing data, and first note that the linear GPA model underlying loss function (2.15) according to the centroid approach may be written as

$$s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j = \mathbf{C}_j\mathbf{Z} + \mathbf{C}_j\mathbf{E}_j \qquad \text{for } j = 1, ..., n, \tag{2.45}$$

where $\mathbf{E}_j = (s_j\mathbf{X}_j\mathbf{R}_j - \mathbf{Z})$ is a (p x m) matrix of residuals. It follows from (2.45) that

$$\sum_{j=1}^{n} s_j^2 \text{ tr } \mathbf{R}_j'\mathbf{X}_j'\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j = \sum_{j=1}^{n} \text{tr } (\mathbf{Z} + \mathbf{E}_j)'\mathbf{C}_j(\mathbf{Z} + \mathbf{E}_j),$$

and therefore that

$$\sum_{j=1}^{n} s_j^2 \text{ tr } \mathbf{X}_j'\mathbf{C}_j\mathbf{X}_j = \sum_{j=1}^{n} \text{tr } \mathbf{Z}'\mathbf{C}_j\mathbf{Z} + \sum_{j=1}^{n} \text{tr } \mathbf{E}_j'\mathbf{C}_j\mathbf{E}_j + 2\sum_{j=1}^{n} \text{tr } \mathbf{Z}'\mathbf{C}_j\mathbf{E}_j. \tag{2.46}$$

Due to the orthogonality of the predictor and residual space in regression it is true that Σ tr $\mathbf{Z}'\mathbf{C}_j\mathbf{E}_j = 0$. This means that (2.46) may be written as

$$\sum_{j=1}^{n} s_j^2 \text{ tr } \mathbf{X}_j'\mathbf{C}_j\mathbf{X}_j = \sum_{j=1}^{n} \text{tr } \mathbf{Z}'\mathbf{C}_j\mathbf{Z} + \sum_{j=1}^{n} \text{tr } \mathbf{E}_j'\mathbf{C}_j\mathbf{E}_j,$$

and therefore as

$$n = \text{tr } \mathbf{Z}'\mathbf{C}\mathbf{Z} + f(\mathbf{s},\mathbf{R},\mathbf{Z}), \tag{2.47}$$

with $f(\mathbf{s},\mathbf{R},\mathbf{Z})$ as in (2.15). Equation (2.47) expresses that the total sum of squares of the n configurations about the origin can be partitioned into two parts. The first part, represented by tr $\mathbf{Z}'\mathbf{C}\mathbf{Z}$, is the proportion of the total sum of squares that is accounted for by the GPA model, while the second part, represented by loss function (2.15), is a residual sum of squares. Dividing (2.47) by n gives

$$1 = (1/n) \text{ tr } \mathbf{Z}'\mathbf{C}\mathbf{Z} + (1/n)\, f(\mathbf{s},\mathbf{R},\mathbf{Z}),$$

which proves that the sum of squares of the centroid configuration satisfies the following conditions

$$0 \leq (1/n)\ \mathrm{tr}\ \mathbf{Z}'\mathbf{C}\mathbf{Z} \leq 1,$$

and therefore can be used as a measure of total fit in GPA.

For the direct approach these relations are even more straightforward. The corresponding loss function

$$f(\mathbf{s},\mathbf{R}) = n - \mathrm{tr}\ (\sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j)'\mathbf{C}^{-}(\sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j) \tag{2.22}$$

may be written as

$$f(\mathbf{s},\mathbf{R}) = n - \mathrm{tr}\ \mathbf{Z}'\mathbf{C}\mathbf{Z},$$

due to (2.18). This shows that the minimization of (2.22) with respect to unknown scaling factors and orthonormal transformation matrices implicitly maximizes the sum of the squared averages of corresponding points. For the direct approach this result is implicit as the averages of corresponding points (i.e., $\mathbf{Z}$) are only calculated *after* convergence of the algorithm (see section 2.4).

Another interesting relation is that, since $(1/n)\ \mathrm{tr}\ \mathbf{Z}'\mathbf{C}\mathbf{Z} = (1/n)\ \mathbf{s}'\mathbf{Y}\mathbf{s} = \phi_1$, where $\mathbf{Y}$ and ϕ_1 are defined in section 2.3.5, GPA in fact maximizes the first eigenvalue of $\mathbf{W}^{-1/2}\mathbf{Y}\mathbf{W}^{-1/2}$ (see formula (2.34)).

Hence, this first eigenvalue, or equivalently the sum of squares of $\mathbf{Z}$, plays much the same role as the squared Pearson product moment correlation coefficient in regression analysis. In contrast with the latter technique, however, until now in GPA no statistical sampling theory has been developed. Thus, as yet no statistical tests exist to determine whether the proportion of the total sum of squares of the data accounted for by the GPA model (i.e., the sum of squares of $\mathbf{Z}$) is significant or not.

We now turn our attention to the relation between the sum of squares of $\mathbf{Z}$ and the other measure of fit that Lingoes and Borg consequently used in their models for the matching of n configurations: the squared correlation between the elements of the optimally transformed $\mathbf{X}_j$'s and $\mathbf{Z}$. We start by giving the relation between these two measures of fit for the general case of incomplete configurations, and then derive the relation between the two measures without missing data as a special case.

Let $r^2(s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j,\mathbf{C}_j\mathbf{Z})$ denote the squared correlation between the elements of $s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j$ and the elements of $\mathbf{C}_j\mathbf{Z}$, then it is easily verified that $r^2(s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j,\mathbf{C}_j\mathbf{Z}) = r^2(\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j,\mathbf{C}_j\mathbf{Z})$. Given an optimal $\mathbf{Z}$ and optimal scaling constants s_j, the general relation between the two measures of fit is given by

$$r^2(\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j,\mathbf{C}_j\mathbf{Z}) = \frac{(\text{tr } \mathbf{Z}'\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j)^2}{(\text{tr } \mathbf{Z}'\mathbf{C}_j\mathbf{Z})(\text{tr } \mathbf{X}_j'\mathbf{C}_j\mathbf{X}_j)} = \frac{s_j^2\ (\text{tr } \mathbf{X}_j'\mathbf{C}_j\mathbf{X}_j)[(1/n)\ \text{tr } \mathbf{Z}'\mathbf{C}\mathbf{Z}]^2}{\text{tr } \mathbf{Z}'\mathbf{C}_j\mathbf{Z}}. \quad (2.48)$$

Proof. Assuming that (2.48) indeed holds, multiplying both terms with $(\text{tr } \mathbf{Z}'\mathbf{C}_j\mathbf{Z})(\text{tr } \mathbf{X}_j'\mathbf{C}_j\mathbf{X}_j)$ results in

$$(\text{tr } \mathbf{Z}'\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j)^2 = s_j^2\ (\text{tr } \mathbf{X}_j'\mathbf{C}_j\mathbf{X}_j)^2\ [(1/n)\ \text{tr } \mathbf{Z}'\mathbf{C}\mathbf{Z}]^2, \quad \text{for } j = 1, ..., n, \quad (2.49)$$

and taking the square root of (2.49) gives

$$\text{tr } \mathbf{Z}'\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j = s_j\ (\text{tr } \mathbf{X}_j'\mathbf{C}_j\mathbf{X}_j)\ [(1/n)\ \text{tr } \mathbf{Z}'\mathbf{C}\mathbf{Z}], \quad \text{for } j = 1, ..., n. \quad (2.50)$$

Hence, to prove that (2.48) holds, it is sufficient to prove that (2.50) holds. Letting

$$\mathbf{W}^{-1/2}\mathbf{Y}\mathbf{W}^{-1/2} = \mathbf{P}\mathbf{\Phi}\mathbf{P}' \quad (2.34)$$

be the eigenvalue-eigenvector decomposition of $\mathbf{W}^{-1/2}\mathbf{Y}\mathbf{W}^{-1/2}$, where $\mathbf{Y}$ and $\mathbf{W}$ are defined as in section 2.3.5, it is always true that

$$\mathbf{P}\mathbf{\Phi}\mathbf{P}'\mathbf{p}_1 = \phi_1\mathbf{p}_1, \quad (2.51)$$

where ϕ_1 is the largest eigenvalue, and $\mathbf{p}_1$ is the corresponding first eigenvector. Substitution of (2.34) in the left side of (2.51), and premultiplying the result with $(\sqrt{n})\mathbf{W}^{1/2}$ yields

$$(\sqrt{n})\mathbf{Y}\mathbf{W}^{-1/2}\mathbf{p}_1 = \phi_1(\sqrt{n})\mathbf{W}^{1/2}\mathbf{p}_1 = \phi_1(\sqrt{n})\mathbf{W}\mathbf{W}^{-1/2}\mathbf{p}_1,$$

and therefore

$$\mathbf{Y}\mathbf{s} = \phi_1\mathbf{W}\mathbf{s}, \quad (2.52)$$

due to (2.35) (see section 2.3.5). Since, however, according to (2.36) and (2.41) it is true that $(n\ \phi_1) = \mathbf{s}'\mathbf{Y}\mathbf{s}$ given optimal scaling constants, (2.52) may be written as

$$\mathbf{Ys} = [(1/n)\ \mathbf{s'Ys}]\mathbf{Ws}. \tag{2.53}$$

This completes the proof because, collecting the n terms on the left side of (2.50) in a vector of order (n x 1), and doing the same with the n elements on the right side of (2.50), and substituting (2.19) in the elements of both vectors, yields identity (2.53).

We briefly discuss the special cases of identity (2.48). When there are no missing data identity (2.48) becomes

$$r^2(\mathbf{X}_j\mathbf{R}_j,\mathbf{Z}) = s_j^2\ (\text{tr}\ \mathbf{X}_j'\mathbf{X}_j)\ (\text{tr}\ \mathbf{Z'Z}),$$

and if the configurations have been unit normalized (i.e., tr $\mathbf{X}_j'\mathbf{X}_j = 1$ for every j) the identity simplifies into

$$r^2(\mathbf{X}_j\mathbf{R}_j,\mathbf{Z}) = s_j^2\ \text{tr}\ \mathbf{Z'Z},$$

as Lingoes and Borg noted. Due to (2.48), we also have the following simple identity in the case without missing data

$$r^2(\mathbf{X}_j\mathbf{R}_j,\mathbf{Z}) = s_j\ \text{tr}\ \mathbf{Z'X}_j\mathbf{R}_j. \tag{2.54}$$

If the total fit is defined as the mean of the configuration fits, then the relations between the two measures for the total fit are as follows. In the case that the configurations contain no missing data the total fit equals

$$(1/n) \sum_{j=1}^{n} r^2(\mathbf{X}_j\mathbf{R}_j,\mathbf{Z}) = (1/n) \sum_{j=1}^{n} s_j^2\ (\text{tr}\ \mathbf{X}_j'\mathbf{X}_j)\ (\text{tr}\ \mathbf{Z'Z})$$

$$= (1/n)\ (\text{tr}\ \mathbf{Z'Z}) \sum_{j=1}^{n} s_j^2\ \text{tr}\ \mathbf{X}_j'\mathbf{X}_j = \text{tr}\ \mathbf{Z'Z}.$$

Without missing data, therefore, the two measures of fit are identical, the mean of the squared correlations then being equal to the sum of squares of $\mathbf{Z}$.

If there are missing data, however, the total fit is

$$(1/n) \sum_{j=1}^{n} r^2(\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j,\mathbf{C}_j\mathbf{Z}) = (1/n) \sum_{j=1}^{n} \frac{s_j^2\ (\text{tr}\ \mathbf{X}_j'\mathbf{C}_j\mathbf{X}_j)[(1/n)\ \text{tr}\ \mathbf{Z'CZ}]^2}{\text{tr}\ \mathbf{Z'C}_j\mathbf{Z}}$$

$$= (1/n)[(1/n)\ \mathrm{tr}\ \mathbf{Z}'\mathbf{C}\mathbf{Z}]^2 \sum_{j=1}^{n} \frac{s_j^2\ (\mathrm{tr}\ \mathbf{X}_j'\mathbf{C}_j\mathbf{X}_j)}{\mathrm{tr}\ \mathbf{Z}'\mathbf{C}_j\mathbf{Z}},$$

and the latter expression is not necessarily equal to (1/n) tr $\mathbf{Z}'\mathbf{C}\mathbf{Z}$. Hence, if the configurations contain missing data the mean of the squared correlations is, in general, no longer identical to the 'size' of the centroid configuration.

Thus, with missing data a choice must be made. For several reasons, we have chosen to use the sum of squares of $\mathbf{Z}$ as a measure of total fit. The first reason for our choice is that this measure is nicely embedded in the GPA model, as expression (2.47) demonstrates, while the squared correlation is not. In the second place, if the configurations contain missing data (tr $\mathbf{Z}'\mathbf{C}\mathbf{Z}$) is completely equivalent to the MAXBET match criterion of Van de Geer (1984), as the next section will show. This equivalence does not hold for the mean of the squared correlations. Last but not least, this measure of fit enables us to generalize Gower's analysis of variance framework for GPA to the case of missing data. Again, this is not possible if squared correlations are used.

In the last part of this section we propose a further decomposition of the sums of squares in GPA. To this end the sum of squares of $\mathbf{Z}$ and the residual sum of squares are partitioned in three separate ways, allowing one to assess: a) the contribution of each individual stimulus to the total fit of the solution, and b) the relative importance of each individual configuration to the proportion of the sum of squares explained by the GPA model, and c) the relative contribution of each dimension to the total GPA solution. These decompositions are a generalization of Gower's analysis of variance. Because the expression 'analysis of variance' suggests the possibility of performing statistical tests, however, we prefer the more neutral expression 'analysis of variation'.

The first decomposition of the total sum of squares with respect to each stimulus separately is given in Table 2.3. Whether we are dealing with complete or incomplete configurations, it can be proved that a_{ii} and b_{ii} in the table are independent components of d_{ii} for each stimulus i. Since the diagonal of the (p x p) matrix $\sum \mathbf{M}_j\mathbf{Z}(\mathbf{M}_j\mathbf{E}_j)'$, where $\mathbf{E}_j = s_j\mathbf{M}_j(\mathbf{X}_j - \mathbf{1}\mathbf{u}_j')\mathbf{R}_j$, consists of the covariations between the p elements a_{ii} and b_{ii}, this diagonal contains zeroes only. The decomposition in

Table 2.3 is based on the following, easily verified, relation between n points $\mathbf{x}_j$ and their centroid $\mathbf{z} = (1/n) \sum \mathbf{x}_j$:

$$\sum_{j=1}^{n} \mathbf{x}_j'\mathbf{x}_j = n\, \mathbf{z}'\mathbf{z} + \sum_{j=1}^{n} (\mathbf{x}_j - \mathbf{z})'(\mathbf{x}_j - \mathbf{z}). \tag{2.55}$$

In words, (2.55) expresses that the sum of squared distances between n points $\mathbf{x}_j$ and the origin always equals n times the squared distance between their centroid and the origin *plus* the sum of squared distances between the n points and their centroid. Notice that we use the GPA model

$$s_j\mathbf{M}_j(\mathbf{X}_j - \mathbf{1}\mathbf{u}_j')\mathbf{R}_j = \mathbf{M}_j\mathbf{Z} + \mathbf{M}_j\mathbf{E}_j, \quad \text{for } j = 1, \ldots, n, \tag{2.56}$$

instead of (2.45) in the decomposition in Table 2.3. The reason is that (2.55) can not be applied in (2.45) because, when dealing with incomplete configurations, the stimulus points of the centroid configuration $\mathbf{Z}$ in (2.45) are idiosyncratically translated for each j (cf. section 2.3.1). In (2.56), on the other hand, $\mathbf{Z}$ remains at the fixed centroid of the individual configurations. The use of (2.56) allows for a

Table 2.3 Contribution of individual stimuli to GPA solution including missing data.

stimulus i	SS_{fit}	$SS_{residual}$	SS_{total}
1	a_{11}*	b_{11}**	d_{11}***
2	a_{22}*	b_{22}**	d_{22}***
⋮	⋮	⋮	⋮
p	a_{pp}*	b_{pp}**	d_{pp}***
$\sum_{i=1}^{p}$	n tr $\mathbf{Z}'\mathbf{Z}$°	$\sum_{j=1}^{n}$ tr $\mathbf{E}_j'\mathbf{C}_j\mathbf{E}_j$	n°

* a_{ii} is diagonal element ii of $\sum_{j=1}^{n} \mathbf{M}_j\mathbf{Z}(\mathbf{M}_j\mathbf{Z})'$

** b_{ii} is diagonal element ii of $\sum_{j=1}^{n} (\mathbf{C}_j\mathbf{E}_j)(\mathbf{C}_j\mathbf{E}_j)'$

*** d_{ii} is diagonal element ii of $\sum_{j=1}^{n} [s_j\mathbf{M}_j(\mathbf{X}_j - \mathbf{1}\mathbf{u}_j')\mathbf{R}_j][s_j\mathbf{M}_j(\mathbf{X}_j - \mathbf{1}\mathbf{u}_j')\mathbf{R}_j]'$

° *if and only if* all configurations are complete

decomposition in independent components both for complete and incomplete configurations, but at a cost: SS_{fit} and SS_{total} in Table 2.3 do not add up to the sum of squares of $\mathbf{Z}$ and to n, respectively, in the case of missing data.

If we attempt to identify the relative contribution of each separate configuration to the total sum of squares in GPA, more serious problems arise. Even without missing data, Gower's decomposition scheme for each configuration sometimes yields a residual sum of squares for a configuration that is larger than its total sum of squares. Therefore, in Table 2.4 a slightly different decomposition is presented which at least can not lead to such a situation. This scheme has the additional advantage that, if we are dealing with complete configurations, the sum of squares accounted for by a configuration is identical to the squared correlation between the elements of the optimally transformed $\mathbf{X}_j$ and $\mathbf{Z}$. But we emphasize that, whether we have complete or incomplete configurations, the components of the decomposition are not independent. In the absence of missing data there is no problem, since, according to (2.54), (s_j tr $\mathbf{Z}'\mathbf{X}_j\mathbf{R}_j$) is then equal to the squared correlation, allowing one to directly identify the relative contribution of a configuration to the total solution. With missing data, however, one must keep in mind that the covariation terms tr $\mathbf{Z}'\mathbf{C}_j\mathbf{E}_j$ are unequal to zero. Therefore, we substract these covariation terms from the total sum of squares associated with each configuration in Table 2.4. The reason that we substract them from the total sum of squares is as follows. If we would add the covariation terms to (s_j tr $\mathbf{Z}'\mathbf{X}_j\mathbf{R}_j$), the property would be destroyed that SS_{fit} is equal to the squared correlation without missing data. If we would add them to tr $\mathbf{E}_j'\mathbf{C}_j\mathbf{E}_j$, on the other

Table 2.4 Contribution of individual configurations to GPA solution including missing data.

configuration j	SS_{fit}	$SS_{residual}$	SS_{total}
1	$s_1$tr $\mathbf{Z}'\mathbf{C}_1\mathbf{X}_1\mathbf{R}_1$	tr $\mathbf{E}_1'\mathbf{C}_1\mathbf{E}_1$	$\{s_1^2$ tr $\mathbf{X}_1'\mathbf{C}_1\mathbf{X}_1$ – tr $\mathbf{Z}'\mathbf{C}_1\mathbf{E}_1\}$
2	$s_2$tr $\mathbf{Z}'\mathbf{C}_2\mathbf{X}_2\mathbf{R}_2$	tr $\mathbf{E}_2'\mathbf{C}_2\mathbf{E}_2$	$\{s_2^2$ tr $\mathbf{X}_2'\mathbf{C}_2\mathbf{X}_2$ – tr $\mathbf{Z}'\mathbf{C}_2\mathbf{E}_2\}$
⋮	⋮	⋮	⋮
n	$s_n$tr $\mathbf{Z}'\mathbf{C}_n\mathbf{X}_n\mathbf{R}_n$	tr $\mathbf{E}_n'\mathbf{C}_n\mathbf{E}_n$	$\{s_n^2$ tr $\mathbf{X}_n'\mathbf{C}_n\mathbf{X}_n$ – tr $\mathbf{Z}'\mathbf{C}_n\mathbf{E}_n\}$
$\sum_{j=1}^{n}$	tr $\mathbf{Z}'\mathbf{C}\mathbf{Z}$	$\sum_{j=1}^{n}$ tr $\mathbf{E}_j'\mathbf{C}_j\mathbf{E}_j$	n

Table 2.5 Contribution of individual dimensions to GPA solution including missing data.

dimension k	SS_{fit}	$SS_{residual}$	SS_{total}
1	e_{11}*	f_{11}**	g_{11}***
2	e_{22}*	f_{22}**	g_{22}***
⋮	⋮	⋮	⋮
m	e_{mm}*	f_{mm}**	g_{mm}***
$\sum_{k=1}^{m}$	tr **Z'CZ**	$\sum_{j=1}^{n}$ tr $\mathbf{E}_j'\mathbf{C}_j\mathbf{E}_j$	n

* e_{kk} is diagonal element kk of $\mathbf{\Lambda}$

** f_{kk} is diagonal element kk of $\sum_{j=1}^{n} (s_j\mathbf{X}_j\mathbf{R}_j\mathbf{K} - \mathbf{ZK})'\mathbf{C}_j(s_j\mathbf{X}_j\mathbf{R}_j\mathbf{K} - \mathbf{ZK})$

*** g_{kk} is diagonal element kk of $\sum_{j=1}^{n} (s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j\mathbf{K})'(s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j\mathbf{K})$

hand, the contribution of each configuration to the total sum of squares would always be equal to (1/n) tr **Z'CZ**, which follows directly from (2.46).

The third and last way in which the total sums of squares can be partitioned makes it possible to identify the relative contribution of each dimension to the total GPA solution, and is given in Table 2.5. The matrices **K** and $\mathbf{\Lambda}$ in the table are the eigenvectors and eigenvalues of $\mathbf{Z'CZ} = \mathbf{K\Lambda K'}$, which are determined in order to rotate the whole GPA solution to the principal components of **Z** (see also section 2.4). Because the whole solution is rotated to the principal components of **Z**, the proportion of sum of squares accounted for by the first dimension will always be the largest, the proportion accounted for by the second dimension will always be second largest, etc. The sums of squares in Table 2.5 can be determined independently both with and without missing data.

The MATCHALS program computes and prints all three decompositions discussed in this section.

2.6 GPA with missing data and MAXNEAR, MAXBET, and MAXDIFF

In this section the relations between the GPA match criterion and Van de Geer's (1984) match criteria MAXNEAR, MAXBET and MAXDIFF are briefly discussed. To be able to express the GPA match criterion in Van de Geer's notation we first define two new matrices. Let supermatrix $\mathbf{X}$ of order (p x nm) contain the n matrices $s_jC_jX_j$, that is let

$$\mathbf{X} = \left[\; s_1C_1X_1 \;\vdots\; s_2C_2X_2 \;\vdots\; \ldots \;\vdots\; s_nC_nX_n \; \right] .$$

Also define matrix $\mathbf{D}$ of order (nm x nm) as the block-diagonal matrix of $\mathbf{X'X}$, that is: diagonal blocks of $\mathbf{D}$ are identical to the diagonal blocks $\{(s_jC_jX_j)'s_jC_jX_j\}$ of $\mathbf{X'X}$, while off-diagonal blocks are zero matrices.

Van de Geer (1984, p. 81) defines the MAXNEAR match criterion for the linear relation among n sets of variables as

> The best MAXNEAR solution requires that projections of corresponding points are as much as possible close together: the sum of the squared distances between projections of corresponding points must be minimized.

In his notation the function that must be minimized according to the MAXNEAR match criterion is

$$f(\mathbf{s},\mathbf{R}) = \text{tr } \mathbf{R}'(\mathbf{D} - \mathbf{X}'\mathbf{C}^{-}\mathbf{X})\mathbf{R} = \text{tr } \mathbf{R}'\mathbf{D}\mathbf{R} - \text{tr } \mathbf{R}'\mathbf{X}'\mathbf{C}^{-}\mathbf{X}\mathbf{R}. \qquad (2.57)$$

The relation between Van de Geer's MAXNEAR criterion (2.57) and the match criterion in GPA in our notation is

$$\begin{aligned} f(\mathbf{s},\mathbf{R}) &= \text{tr } \mathbf{R}'\mathbf{D}\mathbf{R} - \text{tr } \mathbf{R}'\mathbf{X}'\mathbf{C}^{-}\mathbf{X}\mathbf{R} \\ &= \sum_{j=1}^{n} s_j^2 \text{ tr } \mathbf{R}_j'\mathbf{X}_j'\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j - \text{tr } (\sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j)'\mathbf{C}^{-}(\sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j) \\ &= n - \text{tr } (\sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j)'\mathbf{C}^{-}(\sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j), \end{aligned}$$

and the match criterion in GPA therefore is completely identical to MAXNEAR.

The second match criterion discussed by Van de Geer is MAXBET, which he defines as (p. 81):

The best solution maximizes the sum of the squared projections of averages of corresponding points.

Because the matrix **Z** contains these 'averages of corresponding points', this is exactly what happens in GPA, and as MAXBET in Van de Geer's notation is

$$h(\mathbf{s},\mathbf{R}) = \text{tr } \mathbf{R}'\mathbf{X}'\mathbf{C}^{-}\mathbf{X}\mathbf{R},$$

the relation with the match criterion in GPA in our notation is:

$$\begin{aligned} h(\mathbf{s},\mathbf{R}) &= \text{tr } \mathbf{R}'\mathbf{X}'\mathbf{C}^{-}\mathbf{X}\mathbf{R} \\ &= \text{tr } (\sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j)'\mathbf{C}^{-}(\sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j) = \text{tr } \mathbf{Z}'\mathbf{C}\mathbf{Z}. \end{aligned} \tag{2.58}$$

Hence, in GPA the minimization of the MAXNEAR match criterion at the same time results in the maximization of the MAXBET match criterion.

The relation between Van de Geer's MAXDIFF match criterion, finally, and the same criterion in our notation is

$$\begin{aligned} v(\mathbf{s},\mathbf{R}) &= \text{tr } \mathbf{R}'(n\,\mathbf{X}'\mathbf{C}^{-}\mathbf{X} - \mathbf{D})\mathbf{R} = n \text{ tr } \mathbf{R}'\mathbf{X}'\mathbf{C}^{-}\mathbf{X}\mathbf{R} - \text{tr } \mathbf{R}'\mathbf{D}\mathbf{R} \\ &= n \text{ tr } (\sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j)'\mathbf{C}^{-}(\sum_{j=1}^{n} s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j) - \sum_{j=1}^{n} s_j^2 \text{ tr } \mathbf{X}_j'\mathbf{C}_j\mathbf{X}_j \\ &= n \text{ tr } \mathbf{Z}'\mathbf{C}\mathbf{Z} - n, \end{aligned} \tag{2.59}$$

which must be maximized for unknown **s** and **R**. But since the maximization of (2.59) is equivalent to the maximization of (2.58), in generalized Procrustes analysis MAXBET and MAXDIFF are equivalent match criteria.

If the configurations have unequal numbers of columns, in GPA the three match criteria MAXNEAR, MAXBET and MAXDIFF no longer coincide (cf., Ten Berge, 1988). As this section demonstrates, on the other hand, the three criteria of Van de Geer remain equivalent if the configurations have unequal numbers of rows.

2.7 Illustrations

In this section two examples are presented of the generalized Procrustes analysis of configurations containing missing data. The first example consists of a constructed data set, while the data in the second example are empirical. In both examples the convergence criterion in the algorithm presented in section 2.4 was set on 1E-7. The algorithm itself has been programmed in APL.

2.7.1 Regular solids

We constructed a data set consisting of three of the five completely regular three-dimensional solids. These five solids (also called the 'Platonic solids' as Plato, the famous Greek philosopher, used them to explain the structure of the universe) are: the tetrahedron or four-face, the hexahedron or six-face or cube, the octahedron or eight-face, the dodecahedron or twelve-face and the icosahedron or twenty-face. The coordinates of the vertices of the tetrahedron, the hexahedron and the dodecahedron were used as input to the GPA algorithm described in section 2.4.

We chose to fit these particular three of the five solids with GPA because the tetra-, the hexa- and the dodecahedron perfectly fit into one another. That is, by connecting eight vertices of the dodecahedron a cube is created, and the connection of four vertices of either the dodecahedron or the cube yields a tetrahedron. We were interested in finding out whether our algorithm would converge to this perfect solution, even though we placed the three regular solids in different positions in space and scaled them differently. A second reason why these data are interesting to analyse is that they contain quite a number of missing data: the tetrahedron has only four vertices and the cube only eight vertices, while the dodecahedron is made up of twenty vertices.

The coordinates of the vertices of the three solids are given in Table 2.6, and a picture of the relative size and position in space of the solids before iterations is given in Figure 2.5, where the horizontal axis represents the first dimension, the vertical axis the third dimension and the remaining axis is the second dimension.

The optimal translation vectors, scaling constants, orthonormal transformation matrices and centroid configuration **Z** after generalized Procrustes analysis of the three

Table 2.6 Coordinates of vertices of tetrahedron ($\mathbf{X}_1$), hexahedron ($\mathbf{X}_2$) and dodecahedron ($\mathbf{X}_3$) before iterations.

$\mathbf{X}_1$	$\mathbf{X}_2$	$\mathbf{X}_3$
missing	1.0 0.0 -0.5	2.2705 -2.5000 4.9098
missing	missing	1.8197 -1.3197 4.1803
-1.5000 -2.0000 2.4330	1.0 1.0 -0.5	3.0000 -0.5902 3.7295
missing	missing	4.1803 -1.3197 4.1803
missing	missing	3.7295 -2.5000 4.9098
missing	missing	1.0902 -1.7705 3.0000
missing	1.0 1.0 -1.5	1.8197 -1.3197 1.8197
missing	missing	3.0000 -0.5902 2.2705
missing	missing	1.8197 -3.6803 4.1803
missing	missing	3.0000 -4.4098 3.7295
-2.0774 -3.0000 0.8000	0.0 0.0 -0.5	4.1803 -3.6803 4.1803
missing	missing	4.9098 -3.2295 3.0000
missing	0.0 1.0 -0.5	4.9098 -1.7705 3.0000
missing	missing	4.1803 -1.3197 1.8197
-2.0774 -1.0000 0.8000	0.0 1.0 -1.5	3.7295 -2.5000 1.0902
missing	missing	4.1803 -3.6803 1.8197
-0.3453 -2.0000 0.8000	1.0 0.0 -1.5	1.0902 -3.2295 3.0000
missing	missing	2.2705 -2.5000 1.0902
missing	0.0 0.0 -1.5	3.0000 -4.4098 2.2705
missing	missing	1.8197 -3.6803 1.8197

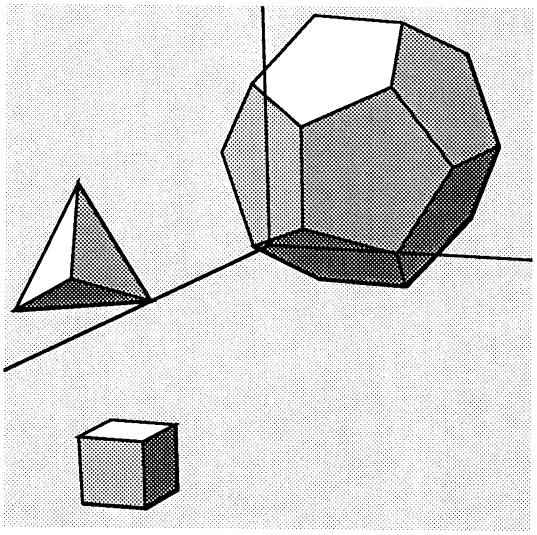

Figure 2.5 Relative size and position in space of the three configurations given in Table 2.6.

Table 2.7 Optimal translation vectors, scaling constants, rotation matrices and centroid configuration **Z** after GPA of data in Table 2.6.

$\mathbf{u}'_1$	$\mathbf{u}'_2$	$\mathbf{u}'_3$
[-.2657 -.3543 .2140]	[.0886 .0886 -.1772]	[.5315 -.4429 .5315]

s_1	s_2	s_3
1.4112	1.9957	0.8454

$\mathbf{R}_1$	$\mathbf{R}_2$	$\mathbf{R}_3$
-.9928 -.0422 .1117	-.8068 .4935 .3248	.9999 -.0125 .0119
-.1192 .4025 -.9076	.3248 .8298 -.4537	.0119 .9986 .0518
.0066 .9144 .4047	.4935 .2606 .8298	-.0125 -.0516 .9986

$$\mathbf{Z} = \begin{bmatrix} -0.1128 & -0.0134 & 0.2843 \\ -0.1769 & 0.1696 & 0.1836 \\ 0.0020 & 0.2800 & 0.1239 \\ 0.1766 & 0.1652 & 0.1878 \\ 0.1057 & -0.0161 & 0.2869 \\ -0.2847 & 0.1127 & 0.0023 \\ -0.1724 & 0.1879 & -0.1695 \\ 0.0048 & 0.2913 & -0.0943 \\ -0.1811 & -0.1834 & 0.1653 \\ -0.0048 & -0.2913 & 0.0943 \\ 0.1724 & -0.1879 & 0.1695 \\ 0.2847 & -0.1127 & -0.0023 \\ 0.2873 & 0.1055 & 0.0091 \\ 0.1811 & 0.1834 & -0.1653 \\ 0.1128 & 0.0134 & -0.2843 \\ 0.1769 & -0.1696 & -0.1836 \\ -0.2873 & -0.1055 & -0.0091 \\ -0.1057 & 0.0161 & -0.2869 \\ -0.0020 & -0.2800 & -0.1239 \\ -0.1766 & -0.1652 & -0.1878 \end{bmatrix}$$

Table 2.8 History of iterations expressed in total loss for data in Table 2.6.

Iteration number	Rotation step	Scaling step
1	0.1795683482	0.0015849889
2	0.0000273636	0.0000261134
3	0.0000002719	0.0000002715
4	0.0000000029	0.0000000029
5	0.0000000001	0.0000000001

Table 2.9 Analysis of variation of GPA solution for data in Table 2.6.

stimulus	SS_{fit}	$SS_{residual}$	SS_{total}
1	0.1875	0.0000	0.1875
2	0.0937	0.0000	0.0937
3	0.2813	0.0000	0.2813
4	0.0937	0.0000	0.0937
5	0.0938	0.0000	0.0938
6	0.0938	0.0000	0.0938
7	0.1875	0.0000	0.1875
8	0.0938	0.0000	0.0938
9	0.0937	0.0000	0.0937
10	0.0938	0.0000	0.0938
11	0.2812	0.0000	0.2812
12	0.0938	0.0000	0.0938
13	0.1875	0.0000	0.1875
14	0.0937	0.0000	0.0937
15	0.2813	0.0000	0.2813
16	0.0937	0.0000	0.0937
17	0.2813	0.0000	0.2813
18	0.0938	0.0000	0.0938
19	0.1875	0.0000	0.1875
20	0.0937	0.0000	0.0937
Σ	3.0000	0.0000	3.0000
configuration			
1	0.3750	0.0000	0.3750
2	0.7500	0.0000	0.7500
3	1.8750	0.0000	1.8750
Σ	3.0000	0.0000	3.0000
dimension			
1	1.0000	0.0000	1.0000
2	1.0000	0.0000	1.0000
3	1.0000	0.0000	1.0000
Σ	3.0000	0.0000	3.0000

configurations are given in Table 2.7. The history of iterations of the algorithm for this example is shown in Table 2.8. As expected, the algorithm converges to a perfect solution. The reason that it does not converge to zero *exactly* is that the coordinates of the vertices given in Table 2.6 have been rounded to four digits. In this example we

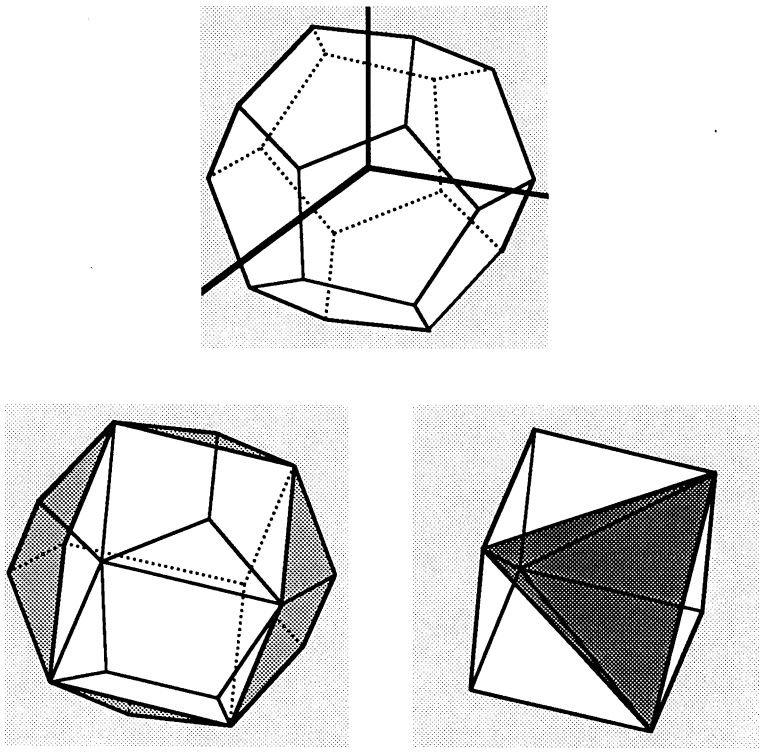

Figure 2.6 Optimally transformed configurations $\mathbf{X}_1$, $\mathbf{X}_2$ and $\mathbf{X}_3$ from Table 2.6 after convergence of the GPA algorithm.

have *not* rotated **Z** to its principal components, for the simple reason that the sum of the three regular solids is completely symmetric. This implies that an eigenvalue-eigenvector decomposition of **Z'CZ** yields three completely identical eigenvalues and thus that all dimensions are equally important.

In Table 2.9 the complete decomposition of the total sum of squares of the three configurations is given with respect to individual stimuli, configurations and dimensions (see section 2.5). Because the solution is perfect, the total residual sum of squares is equal to zero. Therefore, the three-way decomposition of this total residual sum of squares with respect to individual stimuli, configurations and dimensions must also result in zero components, as Table 2.9 points out. The table shows the equal

importance of the three dimensions for the solution. In section 2.5 we discussed that, in the case of missing data, the fitted sum of squares for the stimuli do not add up to the sum of squares of **Z**, and that the total sum of squares of the stimuli do not add up to n. Notice, however, that the fitted sum of squares for the stimuli in this example add up to (n tr **Z**'**Z**), and that the total sum of squares of the stimuli add up to n, notwithstanding the fact that there are two incomplete configurations. This shows that a perfect GPA solution is the one exception to this rule.

The geometry of the solution is illustrated in Figure 2.6. The location of the optimally transformed dodecahedron (i.e., $\mathbf{X}_3$) with respect to the three coordinate axes is shown at the top of the figure. This at the same time demonstrates where the centroid configuration **Z** is located with respect to the coordinate axes since **Z** obviously always is completely identical to the configuration(s) that do(es) not contain any missing data if the solution is perfect. In the bottom left corner of Figure 2.6 is illustrated how the vertices of the optimally transformed hexahedron or cube perfectly fit in the vertices of **Z** (or, which is equivalent in this case, in the vertices of $\mathbf{X}_3$). In the bottom right corner of the figure, finally, is illustrated how the vertices of the optimally transformed tetrahedron again coincide with those of the cube, and therefore with those of the dodecahedron.

In this example the algorithm presented in section 2.4 did exactly what we expected it to do: converge to the global minimum of zero loss. The example discussed here, however, is highly artificial. In the next and last section of this chapter on GPA of configurations containing missing data, a more realistic empirical data set is analyzed.

2.7.2 Personality traits

In this section we analyse a data set consisting of the results of five consecutive studies performed by Van der Kloot (1978) in the years 1969 through 1975. The main objective of these studies was to investigate the structure of the so-called implicit theory of personality, that is, to find out whether there is an underlying common structure in the way people perceive a number of personality traits.

Three of the five studies yielded similarity matrices that could be analyzed with multidimensional scaling techniques. The other two studies resulted in data sets that were analyzed by means of canonical discriminant analysis. All together these analyses again yielded five configurations. What is interesting about these five configurations in the context of this chapter is that different numbers of personality traits were used in the five studies. Our algorithm therefore seems especially appropriate to answer the question whether the five configurations basically have the same structure. In Table 2.10 the coordinates of the five (all of them two-dimensional) configurations are presented.

Table 2.10 Coordinates in two dimensions of five configurations of differing numbers of personality traits.

stimulus	$\mathbf{X}_1$	$\mathbf{X}_2$	$\mathbf{X}_3$	$\mathbf{X}_4$	$\mathbf{X}_5$
aggressive	-.17 .59	-.74 .32	-.39 .63	-.38 .60	-.22 .62
cooperative	.32 -.07	.75 -.07	.79 -.13	.75 -.06	missing
dominant	.22 .43	-.38 .71	.12 .78	.05 .82	missing
intelligent	missing	.67 .22	missing	missing	missing
passive	-.37 -.30	missing	-.20 -.72	-.02 -.91	-.07 -.90
pessimistic	-.32 .00	-.38 -.65	-.32 -.24	-.26 .12	-.04 -.17
submissive	-.03 -.56	-.19 -.84	.10 -.74	.04 -.70	-.19 -.73
unreliable	missing	missing	missing	missing	-.95 -.10

The results of GPA of this data set are given in the following tables. Table 2.11 contains the history of iterations of the GPA algorithm. In Table 2.12 the optimal translation vectors, scaling constants, and orthonormal transformation matrices are given, as well as the coordinates of the centroid configuration rotated to its principal components. Matrix **K** in Table 2.12 is the matrix of eigenvectors of $\mathbf{Z}'\mathbf{C}\mathbf{Z}$, and is used to rotate the five optimally transformed $\mathbf{X}_j$'s and **Z** to the principal components of **Z**. Table 2.13 contains the three-way decomposition of the total sum of squares of

Table 2.11 History of iterations of GPA of data in Table 2.10.

Iteration number	TOTAL LOSS	
	Rotation step	Scaling step
1	0.2749227788	0.1633848176
2	0.1612889220	0.1612885030
3	0.1612805281	0.1612805278
4	0.1612802468	0.1612802467
5	0.1612802426	0.1612802426

Table 2.12 Optimal scaling constants, orthonormal transformation matrices and centroid configuration **Z** after GPA of data in Table 2.10.

s_1	s_2	s_3	s_4	s_5
1.4336	0.8612	0.9772	0.9499	0.9930

$\mathbf{u}_1'$	$\mathbf{u}_2'$	$\mathbf{u}_3'$
[.0404 .0099]	[-.0070 -.1007]	[.1167 -.0085]
$\mathbf{u}_4'$	$\mathbf{u}_5'$	
[.1262 .0273]	[-.3425 -.0960]	

$\mathbf{R}_1$	$\mathbf{R}_2$	$\mathbf{R}_3$
$\begin{bmatrix} .9526 & .3043 \\ -.3043 & .9526 \end{bmatrix}$	$\begin{bmatrix} .9683 & -.2497 \\ .2497 & .9683 \end{bmatrix}$	$\begin{bmatrix} 1.0000 & .0039 \\ -.0039 & 1.0000 \end{bmatrix}$
$\mathbf{R}_4$	$\mathbf{R}_5$	$\mathbf{K}$
$\begin{bmatrix} .9991 & -.0423 \\ .0423 & .9991 \end{bmatrix}$	$\begin{bmatrix} -.9623 & -.2719 \\ -.2719 & .9623 \end{bmatrix}$	$\begin{bmatrix} -.0810 & .9967 \\ .9967 & .0810 \end{bmatrix}$

$$\mathbf{ZK} = \begin{bmatrix} .4039 & -.3018 \\ -.0765 & .3245 \\ .4575 & -.0055 \\ .0739 & .4081 \\ -.4516 & -.2453 \\ -.0749 & -.2973 \\ -.4124 & -.1125 \\ .0801 & .2298 \end{bmatrix}$$

Table 2.13 Analysis of variation of GPA solution for data of Table 2.10.

stimulus	SS_{fit}	$SS_{residual}$	SS_{total}
aggressive	1.2712	0.0062	1.2773
cooperative	0.4445	0.0183	0.4627
dominant	0.8372	0.0121	0.8493
intelligent	0.1720	0.0000	0.1720
passive	1.0563	0.0296	1.0859
pessimistic	0.4699	0.0392	0.5091
submissive	0.9138	0.0560	0.9698
unreliable	0.0592	0.0000	0.0592
Σ	5.2242	0.1613	5.3855
configuration			
1	1.0108	0.0666	1.0774
2	0.9920	0.0415	1.0334
3	1.0719	0.0075	1.0794
4	1.0400	0.0384	1.0784
5	0.7240	0.0073	0.7313
Σ	4.8387	0.1613	5.0000
dimension			
1	3.3061	0.0957	3.4017
2	1.5327	0.0656	1.5983
Σ	4.8387	0.1613	5.0000

the five configurations with respect to individual stimuli, configurations and dimensions.

As can be deduced from Table 2.13, the total fit of the GPA solution for this data set is equal to 4.8387/5.0000 = 0.9677, meaning that the solution accounts for 97% of the sum of squares of the individual configurations. Thus the GPA model fits quite well, suggesting, at least as far as this data set is concerned, that there indeed is an underlying common structure in the way people perceive the eight personality traits investigated in the five studies of Van der Kloot.

The decomposition of the total sum of squares with respect to the individual stimuli shows that the stimuli 'aggressive', 'cooperative', 'dominant', 'intelligent', 'passive',

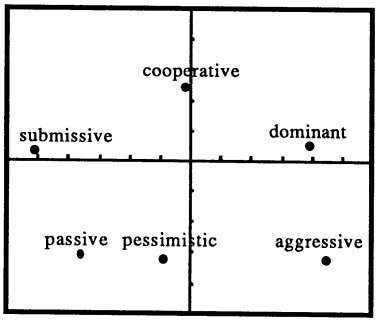

Figure 2.7a $s_1M_1(X_1 - 1u_1')R_1K$.

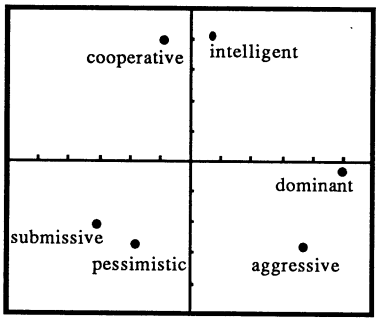

Figure 2.7b $s_2M_2(X_2 - 1u_2')R_2K$.

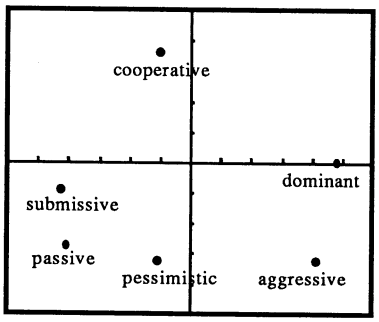

Figure 2.7c $s_3M_3(X_3 - 1u_3')R_3K$.

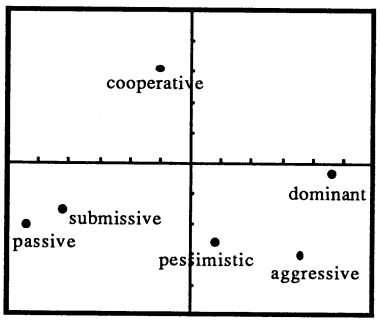

Figure 2.7d $s_4M_4(X_4 - 1u_4')R_4K$.

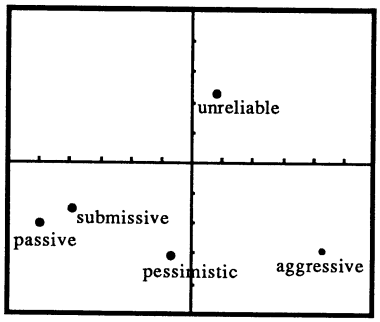

Figure 2.7e $s_5M_5(X_5 - 1u_5')R_5K$.

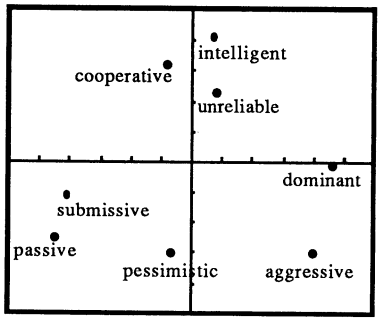

Figure 2.7f ZK.

'pessimistic', 'submissive' and 'unreliable' are accounted for 99.5%, 96.1%, 98.6%, 100%, 97.3%, 92.3%, 94.2% and 100%, respectively. Thus, stimulus 'pessimistic' has the lowest fit, while the stimuli 'intelligent' and 'unreliable' are perfectly represented in the solution. That the fit for the latter two stimuli is perfect is hardly surprising: they both occur only once in the five configurations (see Table 2.10). Therefore no compromises have to be made, and the coordinates of the centroids for these two stimulus points are equal to the coordinates of the stimulus points themselves. Notice that the fitted sums of squares for the eight stimuli do not add up to (tr $\mathbf{Z}'\mathbf{C}\mathbf{Z}$), and that the sum of the total sums of squares is unequal to five, due to the presence of missing data.

The second decomposition in Table 2.13 demonstrates that the five configurations are accounted for 94%, 96%, 99%, 96% and 99%, respectively. The analysis of variation with respect to the two dimensions shows that the first dimension is accounted for 97%, while the second dimension is accounted for 96%. The two dimensions are therefore almost equally important in the GPA solution.

In Figure 2.7 the geometry of the optimally transformed configurations (rotated to the principal components of $\mathbf{Z}$) is shown together with the optimal centroid configuration $\mathbf{Z}$ for this data set (rotated to its principal components). The points in the last plot of Figure 2.7 are the averages of the corresponding points in the first five plots. The two dimensions of all six plots in Figure 2.7 can be interpreted as representing a submission versus dominance dimension (the first dimension of the plots) and an evaluation dimension (the second dimension of the plots). This result is not new: the dimensions found in studies on the implicit personality theory can usually be interpreted in a similar fashion.

The location of the stimulus point for 'unreliable' in the centroid $\mathbf{Z}$ is strange (see Figure 2.7f). One would not expect this stimulus to be situated near 'cooperative' and 'intelligent', but rather in the neighboorhood of any one of the remaining stimuli. We already noted that the stimuli 'unreliable' and 'intelligent' both are perfectly represented in the solution because information about these two stimuli is only available in one configuration (about 'intelligent' in configuration two, and about 'unreliable' in configuration five). The reason that stimulus 'intelligent', though being only represented once, falls on a very understandable spot in the centroid configuration $\mathbf{Z}$, while stimulus 'unreliable' does not, may well be the following. Stimulus 'unreliable' is *in its own configuration* (Figure 2.7e) already an outlier and

being perceived by the subjects in the fifth study as completely different from the other four stimuli. About the location of this stimulus point, therefore, nothing more is known than that it is 'somewhere else' compared to the rest of the stimuli in this study. Stimulus 'intelligent', on the other hand, is situated far from 'submissive', 'pessimistic', 'aggressive' and 'dominant', *but close to* 'cooperative' (Figure 2.7b). Thus, the location of stimulus 'intelligent' in configuration two is more reliable than the location of 'unreliable' in configuration five, since not only information is available about what 'intelligent' is not, but also about what it is similar to (i.e., 'cooperative').

Concluding, this suggests that in GPA of configurations containing missing data the location of a stimulus that occurs only once in all n configurations, and whose position is not 'pinned down' locally by closely related stimuli in its own configuration, should not be given too much weight in the interpretation of the centroid configuration **Z** (notwithstanding the fact that such a stimulus is always perfectly represented in the solution).