Smacof at 50: A Manual
Part 8: Homogeneity Analysis

Jan de Leeuw - University of California Los Angeles

Started April 13 2024, Version of July 10, 2024

Abstract

smacofHO

Contents

Note: This is a working manuscript which will be expanded/updated frequently. All suggestions for improvement are welcome. All Rmd, tex, html, pdf, R, and C files are in the public domain. Attribution will be appreciated, but is not required. The files can be found at https://github.com/deleeuw in the repositories smacofCode, smacofManual, and smacofExamples.

1 Introduction: Categorical Data

In this chapter we shall analyze categorical data structures, with the following components.

- There are m *variables*.
- Variable j has $k_j > 1$ *categories*.
- There are n *objects*.
- Each object defines a *partial order* over the categories of each variable. Make this the simple tree.

Introduce indicators

Discuss: separation and compactness. categories as points or regions. circles, stars, voronoi regions

Start with MSA and De Leeuw (1969), De Leeuw (2004), De Leeuw (2003)

Thus the actual data we collect are the $n \times m$ partial orders $\lesssim_{ij}$.

We study minimization of the stress *loss function*

$$\sigma(X, Y_1, \cdots, Y_m) := \sum_{j=1}^{m}\sum_{i=1}^{n}\min_{\hat{d}_i^j\in\Delta_i^j}\sum_{l=1}^{k_j} w_{il}^j(\hat{d}_{il}^j - d_{il}(X, Y_j))^2 \tag{1}$$

over the $n \times p$ matrix of *object scores* X, the $k_j \times p$ matrices of *category scores* Y_j, and the $n \times k_j$ *transformations* (or *optimal scalings*) Δ_j. We write $d_{il}(X, Y_j)$ for the distance between object i and category l of variable j. Note that for each variable j there are different matrices of category scores Y_j, but there is only a single matrix of object scores X.

The w_{il}^j in definition (1) are non-negative *weights*. Formulas and derivations simplify if the data are *row-weighted*, by which we mean that $w_{il}^j = w_i^j$. They simplify even more if weights are *constant*, i.e. if all non-zero weights are equal to one.

The transformations in (1) are *row-conditional*, in the sense that for each i a vector $\hat{d}_i^j$ of length k_j is selected by the technique from a cone of admissible transformations Δ_i^j. Each row has its own cone.

We need a few words to discuss the meaning of the word "model" in this context, since it is used frequently in data analysis. The model corresponding with a loss function σ is the set of parameter values for which σ attains its global minimum (usually zero). Thus a model is a system of equations and/or inequalities. In the case of loss function (1) the model is that the $d_{il}(X, Y_j)$ with $w_{il}^j > 0$ satisfy the partial order $\lesssim_{ij}$. Define

$$\epsilon_{il\nu}^j := \begin{cases}1 & \text{if we require } \hat{d}_{il}^j \lesssim \hat{d}_{i\nu}^j,\\ 0 & \text{otherwise}\end{cases} \tag{2}$$

Then the model is the system of inequalities

$$w_{il}^j w_{i\nu}^j \epsilon_{il\nu}^j (d_{il}(X, Y_j) - d_{i\nu}(X, Y_j)) \geq 0 \tag{3}$$

If the cones Δ_i^j contain the zero vector, then the global minimum of loss function (1) is clearly equal to zero. Collapsing all x_i and all y_l^j into a single point makes all distances zero, and thus makes stress zero. There is also zero stress if we collapse all x_i into one point and all y_l^j into another point. These solutions are *trivial* in the sense that, although they satisfy the model, they are independent of the data and consequently not informative. There are also more subtle trivial solutions. Suppose the cones Δ_i^j contain the set of all non-negative constant vectors. Collapse all x_i into a single point, and place all y_l^j for variable j on a sphere around the point x_i. There can be different radii for different variables. This makes all $d(x_i, y_l^j)$ equal to the radius of the sphere and thus makes stress zero.

In the context of non-metric unfolding there has been much work on avoiding trivial and degenerate solutions. This started as soon as Kruskal-Guttman-type iterative MDS techniques using data transformation became available. Early contributions were Roskam (1968) and Kruskal and Carroll (1969). For valuable summaries of more recent work, mostly by Willem Heiser and his students, we refer to the dissertations of Van Deun (2005) and Busing (2010).

It follows from the existence of these trivial solutions that we cannot define the purpose of our algorithms as finding the minimum of (1) over all $\widehat{D}_j$, X and Y_j. Some constraints on the optimization problems are needed to prevent these trivial or degenerate solutions.

2 Homogeneity Analysis

The *Gifi System* (Gifi (1990), Michailidis and De Leeuw (1998), De Leeuw and Mair (2009)) implements non-linear or non-metric versions of the classical linear multivariate analysis techniques (regression, analysis of variance, canonical analysis, discriminant analysis, principal component analysis). The non-linear versions are introduced as special cases of *Homogeneity Analysis*, which is better known as *Multiple Correspondence Analysis*.

In this section we present homogeneity analysis as a technique for minimizing the loss function (1) when the data are $n \times k_j$ *indicator matrices* G_j, with $j = 1, \cdots, m$. This is a non-standard presentation, because usually homogeneity analysis is related to principal component analysis, and not to multidimensional scaling (see, for example, De Leeuw (2014) or De Leeuw (1923)). Hoffman and De Leeuw (1992)

In homogeneity analysis the data are (or are coded as) m *indicator matrices* G_j, where G_j is $n \times k_j$. Indicator matrices are binary matrices, with rows that add up to one or to zero. Thus each row has either a single element equal to one and the rest zeroes, or it has all elements equal to zero. Indicator matrices are used to code our categorical variables. Rows corresponds with objects (or individuals), columns with the categories (or levels) of a variable. Element g_{il}^j is one if object i is in category l of variable j, and all other elements in row i are zero. If an object is *missing* on variable j then the whole row is zero.

Homogeneity analysis makes joint maps in p dimensions of objects and categories, both represented as points. A joint map for variable j has n object points x_i and k_j category points y_{il}^j. In a homogeneous solution the object points are close to the points of the categories that the objects score in, i.e, to those y_{il}^j for which $g_{il}^j = 1$. If there is only one variable then it is trivial to make a perfectly homogeneous map. We just make sure the object points coincide with their category points. But there are $j > 1$ indicator matrices, corresponding with m categorical variables, and there is only a single set of object scores. The solution is a compromise trying to achieve as much homogeneity as possible for all variables simultaneously.

In loss function (1) applied to homogeneity analysis the sets Δ_i^j are defined in such a way that $\widehat{d}_{il}^j$ is zero if i is in category l of variable j. There are no constraints on the other $\widehat{d}$'s in row i of variable j. Thus for zero loss we want an object to coincide with all m categories it is in. With this definition of the Δ_i^j we have

$$\min_{\widehat{d}_i^j \in \Delta_i^j} \sum_{l=1}^{k_j} w_{il}^j (\widehat{d}_{il}^j - d(x_i, y_l^j))^2 = f_{ij} d_{ij}^2(X, Y), \tag{4}$$

where

$$d_{ij}(X, Y) := \sum_{l=1}^{k_j} g_{il}^j d(x_i, y_l^j), \tag{5a}$$

$$f_{ij} := \sum_{l=1}^{k_j} w_{il}^j w_{il}^j. \tag{5b}$$

Note that the w_{il}^j for which $g_{il}^j = 0$ play no role in homogeneity analysis. In the usual implementations of homogeneity analysis and multiple correspondence analysis f_{ij} is either zero or one, depending on whether observation i on variable j is missing or non-missing.

Using indicator matrices we can write loss function (4) as

$$\sigma(X, Y_1, \cdots, Y_m) = \sum_{j=1}^{m} \text{tr } (X - G_jY_j)'F_j(X - G_jY_j), \tag{6}$$

The F_j are diagonal matrices with the f_{ij} from (5b) on the diagonal.

In homogeneity analysis we minimize (6) using the explicit normalization $X'F_\star X = I$, where $F_\star$ is the sum of the F_j. The solution is given by the singular value equations

$$X\Lambda = F_\star^{-1} \sum_{j=1}^{m} F_jG_jY_j, \tag{7a}$$

$$Y_j = (G_j'F_jG_j)^{-1}G_j'F_jX, \tag{7b}$$

where Λ is a symmetric matrix of Lagrange multipliers.

In homals (Gifi (1980), De Leeuw and Mair (2009)) alternating least squares is used to solve the equations (7a) and (7b). We start with some initial X, then compute the corresponding Y_j using (7b), then for these new Y_j we compute a new corresponding X from (7a), and so on. Computations are efficient, because only diagonal matrices need to be inverted and matrix multiplication with an indicator matrix is not really multiplication but simply selection of a particular row or column. Alternating least squares thus becomes *reciprocal averaging*. Equation (7b) says that the optimal category point is the weighted averages of the objects points in the category, and (7a) says that, except for rescaling with the Lagrange multipliers, the optimal object point is the weighted average of the category points that the object scores in.

Alternative methods of computation (and interpretation) are possible if we substitute (7b) in (7a) to eliminate the Y_j and obtain an equation in X only. This gives

$$F_\star X\Lambda = \sum_{j=1}^{m} F_jG_j(G_j'F_jG_j)^{-1}G_j'F_jX, \tag{8}$$

which is a generalized eigenvalue equation for X. If we substitute (7a) in (7b) we obtain generalized eigenvalue equations for Y.

$$(G_j'F_jG_j)Y_j\Lambda = \sum_{h=1}^{m} G_j'F_jW_\star^{-1}F_hG_hY_h. \tag{9}$$

If $k_\star$, the sum of the k_j, is not too large then finding the p largest non-trivial eigenvalues with corresponding eigenvectors from (9) may be computationally efficient. The largest "trivial" eigenvalue is always equal to one, no matter what the G_j and W_j are, and we can safely ignore it. The trivial solution with all distances equal to zero mentioned in section 1 corresponds with this largest eigenvalue.

Homogeneity analysis can be most convincingly introduced using the concept of a *star plot*. For variable j we plot k_j category points and n object points in a single joint plot. We then draw a line from each category point to the points of the objects in that category. This creates k_j groups of lines and points in $\mathbb{R}^p$, and each of these groups is called a *star*. The sum of squares of the line lengths of a star is the loss of homogeneity for category l of variable j, and the total sum of squares of all line lengths in the k_j stars is the loss (6) for variable j. Homogeneity analysis chooses X and the Y_j such that X is normalized by $X'F_\star X = I$ and the stars are as small or as compact as possible, measured by the squared line lengths. For given X the stars are as small as possible by choosing the category points Y_j as the centroids of the object points in the category, as in equation (7b). That explains the use of the word “star”, because now the stars really look like stars. In graph theory a star is a tree with one internal node (the category point) and k leaves (the object points). Thus, given the optimum choice of the Y_j as centroids, we can also say that homogeneity analysis quantifies the n objects in such a way that the resulting stars are as small as possible.

3 Multidimensional Structuple Analysis MSA

The Guttman-Lingoes series of programs (Lingoes (1973)) discusses, among many others, several techniques for analyzing a number of indicator matrices. They have the acronyms MSA-I, MSA-II, MSA-III, and MSA-IV, where MSA is short for either Multidimensional Scalogram Analysis or Multidimensional Structuple Analysis. Unfortunately the techniques are rather poorly documented in the mainstream literature. I rely on Lingoes (1968b), Lingoes (1968a), Lingoes (1972), Lingoes (1979), and the various short program descriptions Lingoes published in Behavioural Science. Unfortunately I currently have no access to Lingoes (1973).

All MSA programs start their iterations with MAC-II. MAC stands for Multivariate Analysis of Contingencies, and the technique implements the equations from Guttman (1941). In other words, MAC is homogeneity analysis or multiple correspondence analysis. Thus the MSA programs have the same starting configuration as our smacof programs for categorical data.

The publications on MSA do not pay much attention to the existence of trivial solutions and to the speed of convergence of the iterations.

3.1 MSA-I

The most interesting member of the MSA sequence is MSA-I.

> The logic of MSA-I was worked out by Guttman as a creative reaction to a number of objections to other proposed solutions for multidimensional scalogram analysis raised by members of his course on multidimensional analysis during his visit to The University of Michigan (1964-1965). Some of the computational details and the programming of the technique were done by the author. (Lingoes (1968a), p. 76)

The most complete description of MSA-I is probably Zvulun (1978). There are also some computational details in Lingoes (1968a). So what is this MSA-I model ?

Partition the object points corresponding to any category A into inner and outer points. Take any point not in A and find the closest point in A to that point. Such a closest point is called an *outer point* of category A. Go through all points not in A to find all outer points of A. The points of A that are not outer points of A are *inner points* of A. Category A is *contiguous* if each inner point of A is closer to an outer point of A than to any other outer point. Since the closest point in B to an inner point of A is by definition an outer point of B we have also contiguity if and only if each inner point of A is closer to some outer point of A than to any point outside A.

In MSA-I there are no category points, only object points. This makes comparison with the partitioning by Voronoi regions complicated. In the same way there is no obvious connection with the convex hulls of the object points in a category. Separations and partitions can be quite irregular and in the various small examples I have seen are mostly done after the fact by hand.

The algorithm to optimize contiguity is described in Lingoes (1968a). I will try to reconstruct it.

3.2 MSA-II

Unlike MSA-I, MSA-II, which seems to be mostly due to Lingoes, is pretty straightforward. The model, as taken from Lingoes (1968a), is that there is a $\rho > 0$ such that $g_{il}^j = 1$ implies $d(x_i, y_l^j) \leq \rho$ and $g_{il}^j = 1$ implies $d(x_i, y_l^j) \geq \rho$. Geometrically:

- circles with center x_i and radius ρ contain all categories object i scores in, all other category points are outside the circle
- circles with center y_l^j and radius ρ contain all objects that score in category l of variable j, all other object points are outside the circle.

Computionally we interpret the $n \times k_\star$ binary supermatrix $(G_1 \mid \cdots \mid G_m)$ as a matrix of similarities and apply a non-metric MDS technique. The data consist of two tie-blocks, the ones and the zeroes, and we use the primary approach to ties. Observe there is no row-conditionality here and there is only a single radius ρ.

The loss function for MSA-II is simply Kruskal's stress formula one, implicitely normalized by the sum of all $n \times k_\star$ distances, with monotone regression replaced by rank images.

This use of rank images, by the way, is somewhat problematic. There are nm smallest distances, corresponding with the elements of G equal to one, and $n(k_\star - m)$ largest distances. But how do we define the rank images within the two tie blocks ? Lingoes ranks the distances within the tie blocks from small to large, which seems rather arbitrary.

3.3 MSA-III

MSA-III is closer to our smacofHO method.

4 The smacofHO Loss Function

The smacofHO technique solves the closely related problem in which we do not require, as in homogeneity analysis, that

$$\sum_{l=1}^{k_j} g_{il}^j \hat{d}_{il}^j = 0 \tag{10a}$$

for all i and j, but we impose the weaker condition that for all i and j

$$\sum_{l=1}^{k_j} g_{il}^j \hat{d}_{il}^j \leq \hat{d}_{i\nu}^j \tag{10b}$$

for all $\nu = 1, \cdots, k_j$. In homogeneity analysis the geometric interpretation of loss is that we want objects to coincide with all categories they score in. The geometric interpretation of loss function (1) is that we want objects to be closer to the categories they score in than to the categories they do not score in.

This can be formalized using the notion of *Voronoi regions*. The Voronoi region of category l of variable j is the polyhedral convex set of all points of $\mathbb{R}^p$ closer to category l than to any other category of variable j. The plot of the the k_j categories of variable j defines k_j Voronoi regions that partition $\mathbb{R}^p$. For a wealth of information about Voronoi regions we refer to

Loss function (1) with Δ defined by constraint(10b) vanishes if for each variable all x_i are in the Voronoi regions of the categories they score in. This condition implies, by the way, that the interiors of the k_j convex hulls of the x_i in a given category are disjoint, and the point clouds can consequently be weakly separated by hyperplanes. Since the category points themselves are in their own Voronoi region the convex hulls of the stars are also disjoint.

The initial configuration for the iterations is computed using homogeneity analysis. In each iteration configuration updates are alternated with updates of the $\widehat{D}_j$. The general majorization theory for MDS with restrictions (De Leeuw and Heiser (1980)) calls for configuration updates in two steps. In the first step we compute the Guttman transform of the current configuration, and in the second step we project the Guttman transform on the set of constrained configurations.

Minimizing loss (1) over the $\hat{d}_i^j$ is a monotone regression problem for a simple tree order. This is easily solved by using Kruskal's primary approach to ties (Kruskal (1964a), Kruskal (1964b), De Leeuw (1977)).

4.1 The Guttman Transform

The smacof iterations, or Guttman transforms, more or less ignore the fact that we are dealing with a rectangular matrix and use the weights to transform the problem into a symmetric one (as in Heiser and De Leeuw (1979)).

The loss function is

$$\sigma(Z_1, \cdots, Z_m) = \sum_{j=1}^{m} \sum_{i=1}^{n_j} \sum_{k=1}^{n_j} w_{ik}^j (\hat{d}_{ik}^j - d_{ik}(Z_j))^2, \tag{11}$$

with $n_j := n + k_j$ and with Z_j the $n_j \times p$ matrices that stack X on top of Y_j. The w_{ik}^j are zero for the diagonal $n \times n$ and the diagonal $k_j \times k_j$ block.

To compute the Guttman transform of Z_j we have to solve the partitioned system

$$\begin{bmatrix} R_W & -W \\ -W' & C_W \end{bmatrix} \begin{bmatrix} \tilde{X} \\ \tilde{Y} \end{bmatrix} = \begin{bmatrix} R_B & -B \\ -B' & C_B \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} \tag{12}$$

Since we have to solve this system for each variable separately we forget about the index j here. In (12) R_W and C_W are the diagonal matrices with row and column sums of $-W$, while R_B and C_B are diagonal matrices with the row and columns sums of the $n \times k_j$ matrix B, which has elements

$$b_{il} = w_{il} \frac{\hat{d}_{il}}{d_{il}(X, Y_l)}. \tag{13}$$

Matrices X and Y are the two parts of the current Z that we are updating, while we solve for $\tilde{X}$ and $\tilde{Y}$, the two parts of the Guttman transform.

Define

$$\begin{bmatrix} P \\ Q \end{bmatrix} := \begin{bmatrix} R_B & -B \\ -B' & C_B \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} \tag{14}$$

Now $R_W\tilde{X} - W\tilde{Y} = P$ or $\tilde{X} = R_W^{-1}(P + W\tilde{Y})$. Substitute this in $C_W\tilde{Y} - W'\tilde{X} = Q$ to get $C_W\tilde{Y} - W'R_W^{-1}(P + W\tilde{Y}) = Q$ or

$$(C_W - W'R_W^{-1}W)\tilde{Y} = Q + W'R_W^{-1}P \tag{15}$$

We solve equation (15) for $\tilde{Y}$ and then use $\tilde{X} = R_W^{-1}(P + W\tilde{Y})$. Note that $C_W - W'R_W^{-1}W$ is doubly-centered. As in homogeneity analysis we hope that $k_\star$ is not to big, and we avoid generalized inverses of very large and very sparse matrices.

4.2 The smacof Projection

After computing the Guttman transforms $\tilde{X}_j$ and $\tilde{Y}_j$ we have to project them on the set of constrained configurations.

First suppose the only constraint is $X_j = X$. We will discuss some additional (optional) constraints in a while. To project we must minimize

$$\begin{split}\sum_{j=1}^m \text{tr } (X - \tilde{X}_j)' R_j (X - \tilde{X}_j) - 2\sum_{j=1}^m \text{tr } (X - \tilde{X}_j)' W_j (Y_j - \tilde{Y}_j) + \\ \sum_{j=1}^m \text{tr } (Y_j - \tilde{Y}_j)' C_j (Y_j - \tilde{Y}_j)\end{split} \tag{16}$$

where R_j and C_j are now the diagonal matrices of row and column sums of the W_j.

The stationary equations are

$$Y_j = \tilde{Y}_j - C_j^{-1}W_j'(X - \tilde{X}_j), \tag{17}$$

$$X = \{R_\star\}^{-1}\sum_{j=1}^{m}\left\{R_j\tilde{X}_j - W_j(Y_j - \tilde{Y}_j)\right\}. \tag{18}$$

We solve these equations iteratively using alternating least squares. This means using (17) to compute a new Y for given X and (18) to compute a new X for given Y. We alternate these two updates until convergence.

Thus we have an iterative “inner” ALS process within the iterative “outer” ALS process of alternating the Guttman transform/projection and the monotone regressions. More precisely the inner iterations are in the projection phase of the Guttman update.

If there are further constraints on X, besides $X_j = X$, and if there are constraints on Y_j the updates in the projection phase must be modified.

4.2.1 Rank Constraints for Y

If ww choose to do we can require the Y_j to have rank $r_j \leq \min(k_j, p)$, i.e. $Y_j = Q_jA_j'$ with Q_j a $k_j \times r_j$ matrix and A_j a $p \times r_j$ matrix. The rank-constraint on Y_j is taken from the Gifi system, where it serves to connect homogeneity analysis with forms of non-linear principal component analysis.

If $r = 1$ then geometrically having all y_l^j on a line through the origin implies that all Voronoi boundaries are hyperplanes perpendicular to that line, and consequently all Voronoi regions are bounded by two parallel hyperplanes (parallel lines if $p = 2$). All objects scores must orthogonally project on the line in the interval corresponding with the category theyscore in. Note that the intervals on the line are actually the one-dimensional Voronoi regions of the line with the category points.

If $r = 2$ and $p = 3$, another case that may be practically relevant, then category points are in a hyperplane through the origin. The Voronoi regions in three-dimensional space are bounded by lines perpendicular to that plane, intersecting the plane at the two-dimensional Voronoi points for that plane. The object points must be in the correct polyhedral cylinder.

For each of the m variables we can independently choose the ranks r_j of the Y_j and combine it with one of the three options for X, creating a large number of different analyses (in a given dimensionality p).

If there are rank constraints on one of more of the Y_j then for those j we have to minimize

$$2\text{tr}\, A_j'\{Y_j'C_j - (X - \tilde{X}_j)'W_j\}Q_j + \text{tr}\, A_j'Q_j'C_jQ_jA_j$$

The stationary equations are

$$\{Y_j'C_j - (X - \tilde{X}_j)'W_j\}Q_j = Q_j'C_jQ_jA_j,$$

and

$$\{Y_j'C_j - (X - \tilde{X}_j)'W_j\}'A_j = C_jQ_jA_jA_j'$$

4.2.2 Centroid Constraints on Y

If we require that $Y_j = (G_j'R_jG_j)^{-1}G_j'R_jX$ then this effectively eliminates the Y_j as variables from the optimization problem and we only have to optimize over X. We must minimize

$$\sum_{j=1}^m \text{tr }(X-\tilde{X}_j)'R_j(X-\tilde{X}_j) - \text{tr }(X-\tilde{X}_j)'W_j(H_jX-\tilde{Y}_j)+ \\ \text{tr }(H_jX-\tilde{Y}_j)'C_j(H_jX-\tilde{Y}_j) \quad (19)$$

Expanding

$$2\ X'R_\star X - 2\text{ tr }X'\sum_{j=1}^m R_j\tilde{X}_j - 2\text{ tr }X'\{\sum_{j=1}^m W_jH_j\}X + 2\text{tr }X'\sum_{j=1}^m W_j\tilde{Y}_j+ \\ \text{tr }X'\{\sum_{j=1}^m H_j'C_jH_j\}X - 2\text{tr }X'\{\sum_{j=1}^m H_j'C_j\tilde{Y}_j\} \quad (20)$$

Substituting $Y_j = H_jX$ with $H_j := (G_j'R_jG_j)^{-1}G_j'R_j$ in .. and simplifying gives the stationary equations $P_\star X = Q_\star$ with

$$P_\star := \sum_{j=1}^m\{R_j - H_j'W_j' - W_jH_j + H_j'C_jH_j\}, \quad (21a)$$

$$Q_\star := \sum_{j=1}^m\{(R_j - H_j'W_j')\tilde{X}_j - (W_j - H_j'C_j)\tilde{Y}_j\}. \quad (21b)$$

Thus the unnormalized solution for the object scores is $X = P_\star^+Q_\star$.

We want to avoid inversion of the matrix $P_\star$, which has order n. In fact we do not want to compute and store $P_\star$ at all. huppel

We avoid the inversion by using majorization. Suppose μ is such that $P_\star \lesssim \mu R_\star$ in the Loewner sense. We would typically take μ as the largest eigenvalue of $R_\star^{-1}P_\star$.

Define

$$\omega(X) := \text{tr }X'P_\star X - 2\text{ tr }X'Q_\star$$

Then, reculer pour mieux sauter, writing $\overline{X}$ for the current best X,

$$\omega(X) = \text{tr }(\overline{X}+(X-\overline{X}))'P_\star(\overline{X}+(X-\overline{X})) - 2\text{ tr }(\overline{X}+(X-\overline{X}))'Q_\star$$

Now

$$\omega(X) \leq \omega(\overline{X}) + \mu\text{tr }(X-\overline{X})'R_\star(X-\overline{X}) - 2\text{ tr }(X-\overline{X})'(Q_\star - P_\star\overline{X}) \quad (22)$$

The stationary equations in the unnormalized case have solution

$$X = \overline{X} + \mu^{-1}R_\star^{-1}(Q_\star - P_\star\overline{X})$$

If we require the normalization $X'R_\star X = I$ then we must solve the Procrustus problem

$$(\mu R_\star - P_\star)\overline{X} - Q_\star = R_\star X\Lambda$$

with Λ a symmetric matrix of Lagrange multipliers.

4.2.3 Normalization of X

Besides $X_j = X$ we can also optionally require the normalization constraint $X' R_\star X = I$. In both cases the stationary equation (17) remains the same, while (17) becomes

$$R_\star X \Lambda = P_\star,$$

with

$$P_\star := \sum_{j=1}^{m} \left\{ R_j \tilde{X}_j - W_j (Y_j - \tilde{Y}_j) \right\}$$

and with Λ a symmetric matrix of Lagrange multipliers. Using the symmetric square root, $\Lambda = (P' R_\star^{-1} P)^{\frac{1}{2}}$ and thus

$$X = R_\star^{-1} P (P' R_\star^{-1} P)^{-\frac{1}{2}}.$$

Thus requiring normalized object scores only needs small modifications in the X update step of the unnormalized update $R_\star^{-1} P$.

5 Convergence and Degeneracy

6 Utilities

6.1 Object Plot Function

6.2 Category Plots Function

6.3 Joint Plot Function

6.4 Prediction Table

In the solution (X, Y) we say that pair (i, j) is a *hit* if

$$d_{il}^{j}(X, Y) = \overset{k_j}{\underset{\nu=1}{\min}}\, d_{i\nu}^{j}(X, Y)$$

or, in words, if object point x_i is in the Voronoi region of the category point corresponding to the category the object scores in.

7 Examples

Since there are so many different analyses that can be done (choosing the rank and normalization constraints), and since each analysis leads to a large number of plots, presentation of results is a problem. We encourage readers to repeat the analyses and study the output in more detail.

7.1 Small

We start we a small artificial example, earlier used for illustrative purposes in Gifi (1990), chapter 2. The data have $n = 10$ objects and $m = 3$ variables with 3, 3, 2 categories.

```
##    first second third
## 01     a      p     u
## 02     b      q     v
## 03     a      r     v
## 04     a      p     u
## 05     b      p     v
## 06     c      p     v
## 07     a      p     u
## 08     a      p     v
## 09     c      p     v
## 10     a      p     v
```

7.1.1 Homogeneity

We first give the Voronoi plus star joint plots for a homogeneity analysis of the data, using the function `smacofHomogeneityHO()`.

INSERT FIGURE 1 ABOUT HERE

The solution is Voronoi homogeneous for variables one and three. For variable two the star for category p has objects in the Voronoi region of category r, and is consequently not perfectly homogeneous. We also see this in the prediction table from this analysis.

```
##       [,1] [,2] [,3]
##  [1,]    1    0    1
##  [2,]    1    1    1
##  [3,]    1    1    1
##  [4,]    1    0    1
##  [5,]    1    0    1
##  [6,]    1    1    1
##  [7,]    1    0    1
##  [8,]    1    1    1
##  [9,]    1    1    1
## [10,]    1    1    1
```

Note that variable p is atypical, because eight of the ten objects are in category p, while categories q and r only have a single object in them.

7.1.2 Voronoi

We next use the Homogeneity Analysis solution as intial estimate for a smacofHO analysis without normalization or rank constraints. Stress is 1.1313536×10^{-9} after 169 iterations.

INSERT FIGURE 2 ABOUT HERE

As expected, variables one and three, which already have perfect fit, do not change. There is some change in variable two, in the right direction, but it is not enough to improve the number of correct predictions.

```
##       [,1] [,2] [,3]
##  [1,]    1    0    1
##  [2,]    1    1    1
##  [3,]    1    1    1
##  [4,]    1    0    1
##  [5,]    1    0    1
##  [6,]    1    1    1
##  [7,]    1    0    1
##  [8,]    1    1    1
##  [9,]    1    1    1
## [10,]    1    1    1
```

7.1.3 Rank one

Bext, for this example, we constrain the Y_j to be of rank one and leave X unnormalized. Stress is $4.3071361 \times 10^{-10}$ after 73 iterations.

INSERT FIGURE 3 ABOUT HERE

Variable three, which is binary, does not change. The plot for variable two changes for the better. To improve the fit the algorithm moves the category points for categories p and r very close together. There is still one prediction violation in variable two, but if the category points of p and r coincide they have the same Voronoi region and the prediction violation disappears. This may happen if we continue iterating. The same is true for the prediction violation in variable one, where object five in category b is only marginally on the wrong side of the boundary between a and b.

```
##       [,1] [,2] [,3]
##  [1,]    1    1    1
##  [2,]    1    1    1
##  [3,]    1    1    1
##  [4,]    1    1    1
##  [5,]    0    0    1
##  [6,]    1    1    1
##  [7,]    1    1    1
##  [8,]    1    1    1
##  [9,]    1    1    1
## [10,]    1    1    1
```

We do note that the constrained version does better than the unconstrained version. But this merely means that the constrained version finds a better local minimum – both analyses do not find the global minimum, which we know is equal to zero.

7.1.4 Centroids

The next analysis has unnormalized X and centroid restrictions on the Y_j.

Stress is 1.4410786×10^{-9} after 288 iterations.

INSERT FIGURE 4 ABOUT HERE

```
##       [,1] [,2] [,3]
##  [1,]    1    1    1
##  [2,]    1    1    1
##  [3,]    1    1    1
##  [4,]    1    1    1
##  [5,]    0    0    1
##  [6,]    1    1    1
##  [7,]    1    1    1
##  [8,]    1    1    1
##  [9,]    1    1    1
## [10,]    1    1    1
```

7.1.5 Circular

The final analysis for the small example has unnormalized X and circular restrictions on the Y_j.

Stress is 9.1284764×10^{-8} after 203 iterations.

INSERT FIGURE 5 ABOUT HERE

7.2 Cetacea

Our first real example has $m = 15$ variables and $n = 36$ objects. The objects are genera of whales, dolphins, and porpoises . The variables are morphological, osteological, and behavioral descriptors, all categorical with a small number of categories. They are (with the number of categories)

```
##                                     [,1]
## NECK                                  2
## FORM OF THE HEAD                      6
## SIZE OF THE HEAD                      2
## BEAK                                  4
## DORSAL FIN                            4
## FLIPPERS                              4
## SET OF TEETH                          5
## FEEDING                               4
## BLOW HOLE                             4
## COLOR                                 5
## CERVICAL VERTEBRAE                    2
## LACRYMAL AND JUGAL BONES              3
## HABITAT                               5
## LONGITUDINAL FURROWS ON THE THROAT    3
## HEAD BONES                            5
```

In order to be able to interpret the plots, we also give the 36 genera.

```
##       [,1]
##  [1,] "Bowhead whales"
##  [2,] "Rorquals"
##  [3,] "Blue whale"
##  [4,] "Giant bottle-nosed whales"
##  [5,] "Commerson's Dolphins"
##  [6,] "White whales"
##  [7,] "Common dolphins"
##  [8,] "Grey whales"
##  [9,] "Right whales"
## [10,] "Pilot whales"
## [11,] "Risso's dolphins"
## [12,] "Bottle-nosed whales"
## [13,] "Amazon dolphins"
## [14,] "Pygmy sperm whales"
## [15,] "White-sided dolphins"
## [16,] "Chinese river dolphins"
## [17,] "Right whale dolphins"
## [18,] "Humpback whales"
## [19,] "Sowerby's whales"
## [20,] "Narwhals"
## [21,] "Pygmy right whales"
## [22,] "Finless black porpoises"
## [23,] "Irawady dolphins"
## [24,] "Killer whales"
## [25,] "Common porpoises"
## [26,] "Sperm whales"
## [27,] "Gangetic dolphins"
## [28,] "False killer whales"
## [29,] "Guyanian river dolphins"
## [30,] "Cameroun's dolphins"
## [31,] "Spotted dolphins"
## [32,] "Rough toothed dolphins"
## [33,] "La Plata's dolphins"
## [34,] "Shepherd's beaked whales"
## [35,] "Bottle-nosed dolphins"
## [36,] "Goosebeak whales"
```

The data matrix has been constructed by Vescia (1985). Chapter 1 of the book edited by Marcotorchino, Proth, and Janssen (1985) has the data, and a number of sub-chapters in which different data analysts apply various techniques to these data and discuss the results. Among the contenders were MSA-I (Guttman (1985)) and homals (Van der Burg (1985)).

7.2.1 Homogeneity

We start with a homogeneity analysis.

INSERT FIGURE 6 ABOUT HERE

The number of correct predictions per variable, out of a possible 36, is

```
hcethompre <- smacofPredictionTable(hcethom)
print(colSums(hcethompre, na.rm = TRUE))
```

```
##  [1] 31 17 34 25 28 20 21 17 26  8 31 34 16 18 25
```

Thus we predict correctly in 65 percent of the cases.

7.2.2 Voronoi

Stress is 5.9054624×10^{-8} after 3417 iterations.

INSERT FIGURE 7 ABOUT HERE

The number of correct predictions per variable is

```
##  [1] 33 15 35 33 28 25 28 25 23 36 29 35 16 22 28
```

Thus we predict correctly in 76 percent of the cases.

7.2.3 Voronoi, Normed

Stress is 6.3457203×10^{-8} after 2584 iterations.

INSERT FIGURE 8 ABOUT HERE

```
hcetnopre <- smacofPredictionTable(hcetno)
print(sum(hcetnopre, na.rm = TRUE))
```

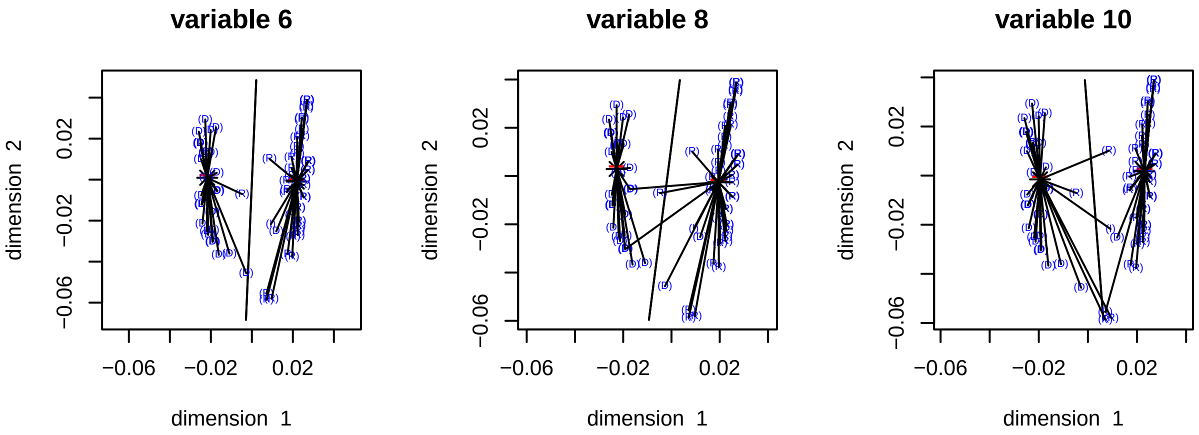
There is an alternative analysis available, based on the same model, which uses the smacofPC program. The indicator matrix can be converted into $n(k_\star - m)$ tetrads, for the cetacea example that is 1548 tetrads. These tetrads are used in an MDS problem of order 94, using a matrix of weights in which only the 36 by 58 off-diagonal matrix is non-zero.

7.3 Senate

7.3.1 Homogeneity

INSERT FIGURE 9 ABOUT HERE

```
par(mfrow = c(1,3))
smacofJointPlotsHO(hhom, jvar = c(6, 8, 10), objects = TRUE, voronoi = TRUE, stars = TRU
```



```
##  [1]  93  95  97  88  89 100  99  97 100  96  98  91  96  85  89  87  89  86  86
## [20]  85
```

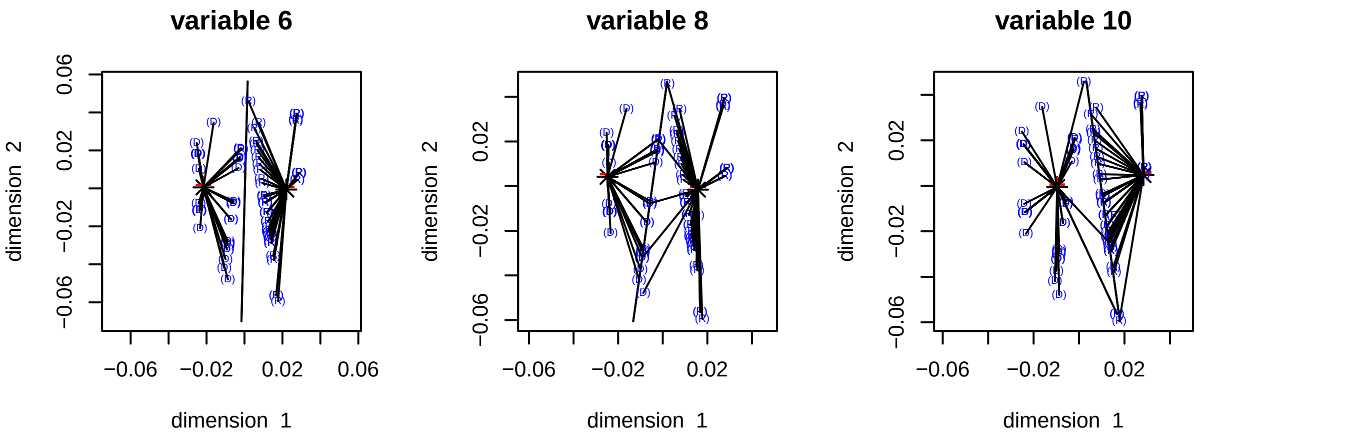
Thus we predict correctly in 92.3 percent of the cases.

7.3.2 Voronoi

Stress is 2.0710178×10^{-7} after 7449 iterations.

INSERT FIGURE 10 ABOUT HERE

```
par(mfrow = c(1,3))
smacofJointPlotsHO(hho, jvar = c(6, 8, 10), objects = TRUE, voronoi = TRUE, stars = TRU
```



```
##  [1]  87  94  91  86  79 100  99  92 100  95  95  90  96  87  80  98  84  79 100
## [20]  79
```

Thus we predict correctly in 90.55 percent of the cases.

7.3.3 Centroid

Stress is 3.8200404×10^{-5} after 10^4 iterations.

INSERT FIGURE 10 ABOUT HERE

```
##  [1] 78 84 90 85 84 95 97 94 95 95 95 79 79 73 81 82 89 82 99 85
```

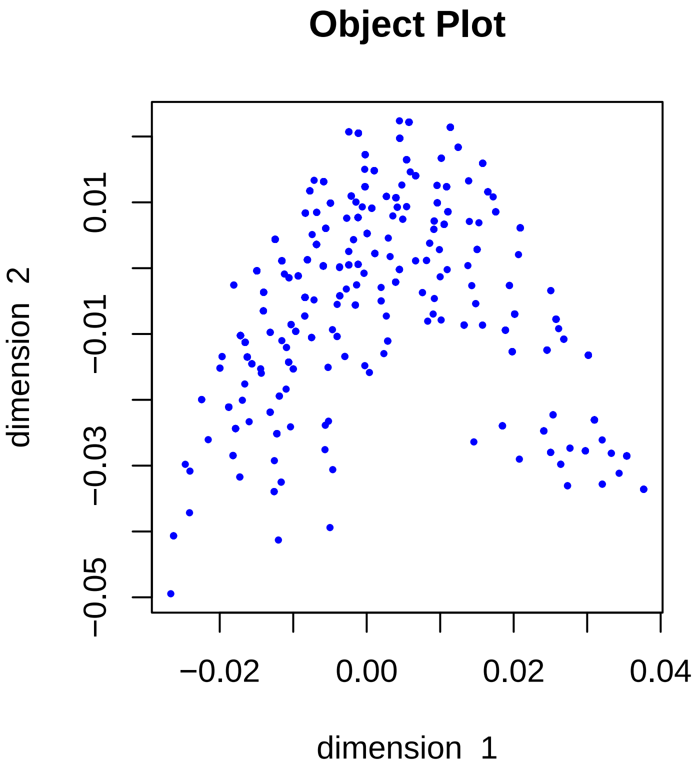
Thus we predict correctly in 87.05 percent of the cases.

Stress is 3.8250142×10^{-5} after 10^4 iterations.

7.4 GALO

```
hgalohom <- smacofHomogeneityHO(galo[, 1:4])
```

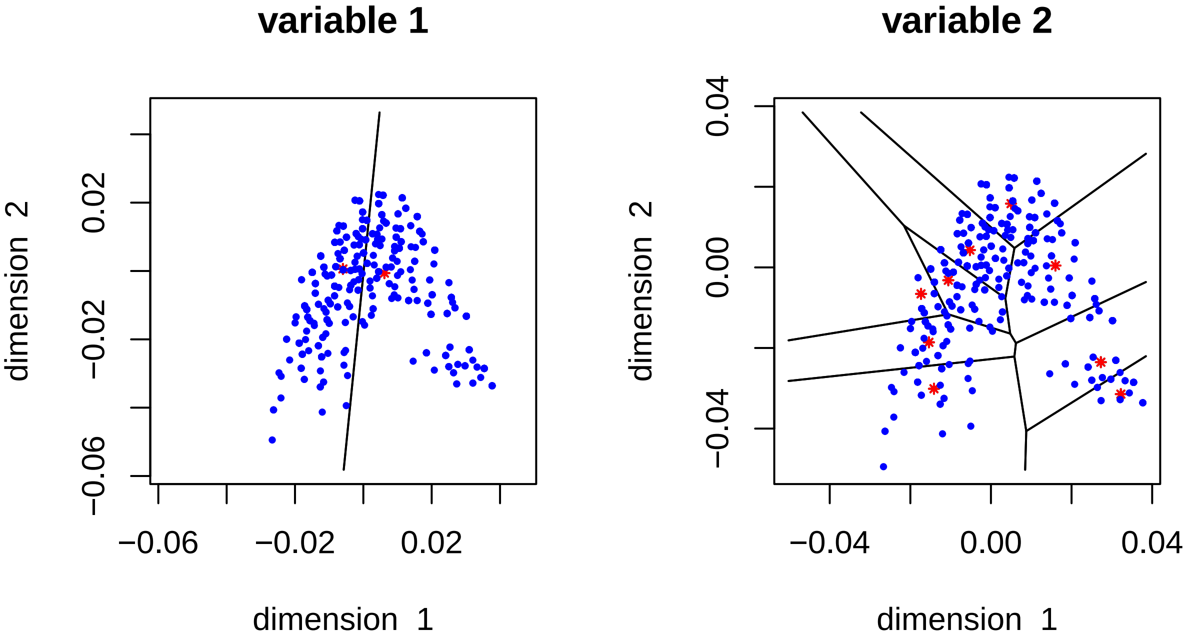
```
smacofObjectsPlotHO(hgalohom, cex = .5)
```



```r
hgalohompre <- smacofPredictionTable(hgalohom)
print(colSums(hgalohompre, na.rm = TRUE))
```

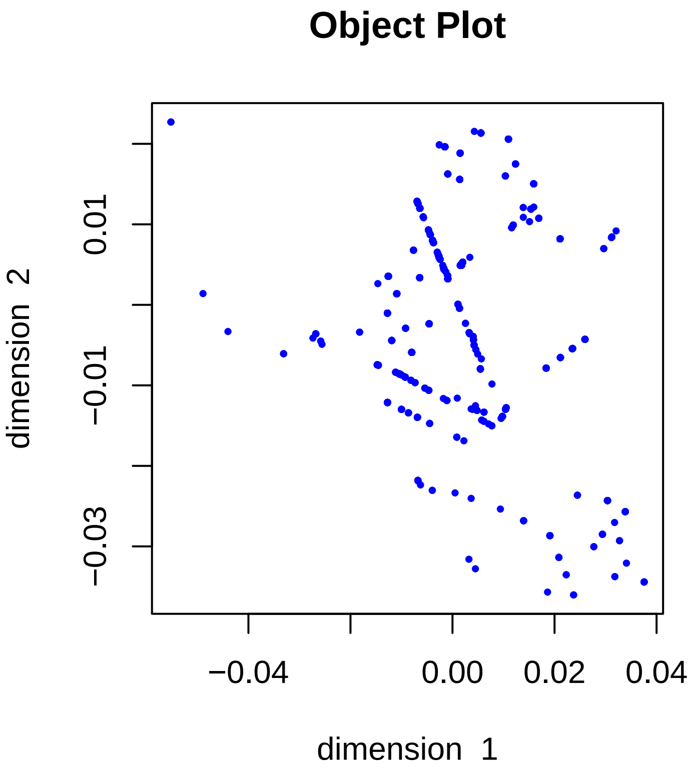
```
## [1] 863 828 799 525
```

```r
par(mfrow = c(1, 2))
smacofJointPlotsHO(hgalohom, jvar = 1:2, objects = TRUE, voronoi = TRUE, xcex = .5, ycex
```



```r
hgaloho <- smacofHO(galo[, 1:4], verbose = FALSE, itmax = 10000)
```

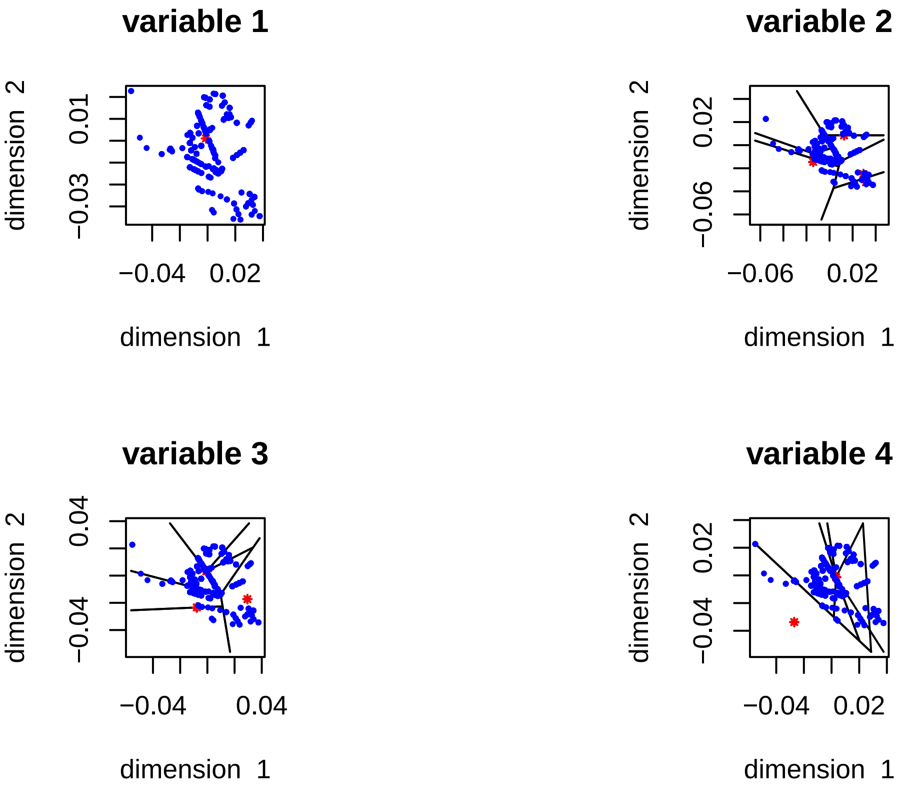
```r
smacofObjectsPlotHO(hgaloho, cex = .5)
```



```r
hgalohopre <- smacofPredictionTable(hgaloho)
print(colSums(hgalohopre, na.rm = TRUE))
```

```
## [1] 1290  397  921  385
```

```r
par(mfrow = c(2, 2))
smacofJointPlotsHO(hgaloho, objects = TRUE, voronoi = TRUE, xcex = .5, ycex = .5, clabel
```


variable 1
dimension 2
0.01
−0.03
−0.04
0.02
dimension 1
variable 2
dimension 2
0.02
−0.06
−0.06
0.02
dimension 1
variable 3
dimension 2
0.04
−0.04
−0.04
0.04
dimension 1
variable 4
dimension 2
0.02
−0.04
−0.04
0.02
dimension 1

8 Generalizations

1. Fuzzy Indicators
2. Voronoi with clouds
3. Circles with varying radius
4. Disjoint circles

9 Figures

9.1 Small

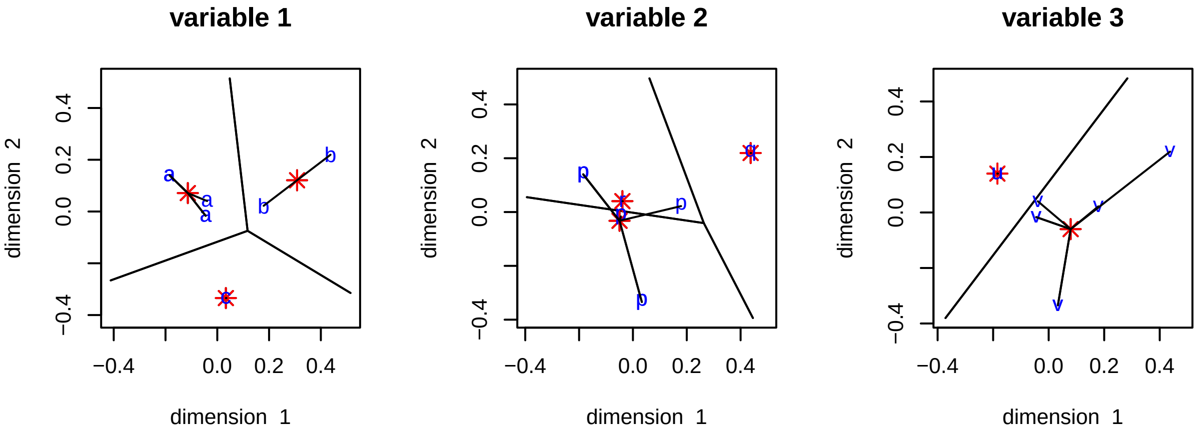


Figure 1: Small example, Homogeneity Analysis

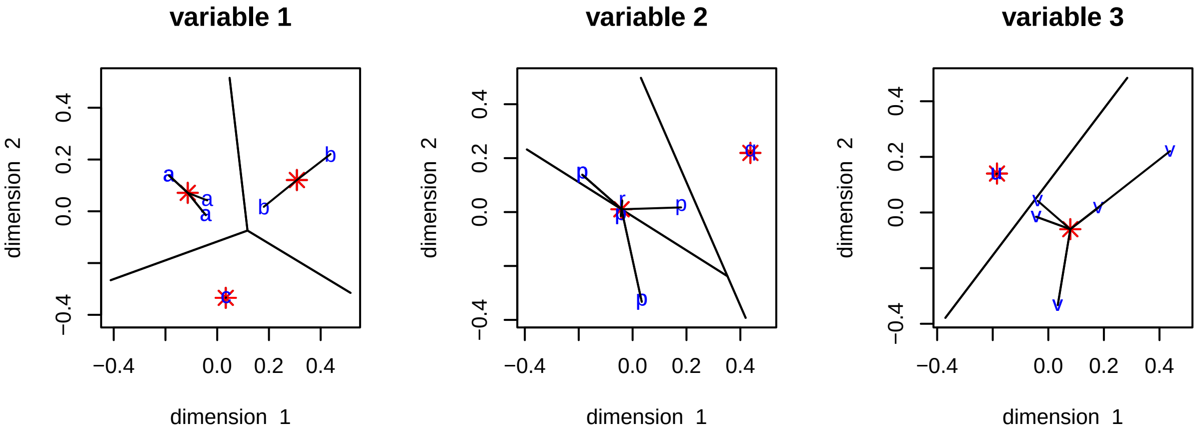


Figure 2: Small example, smacofHO unrestricted

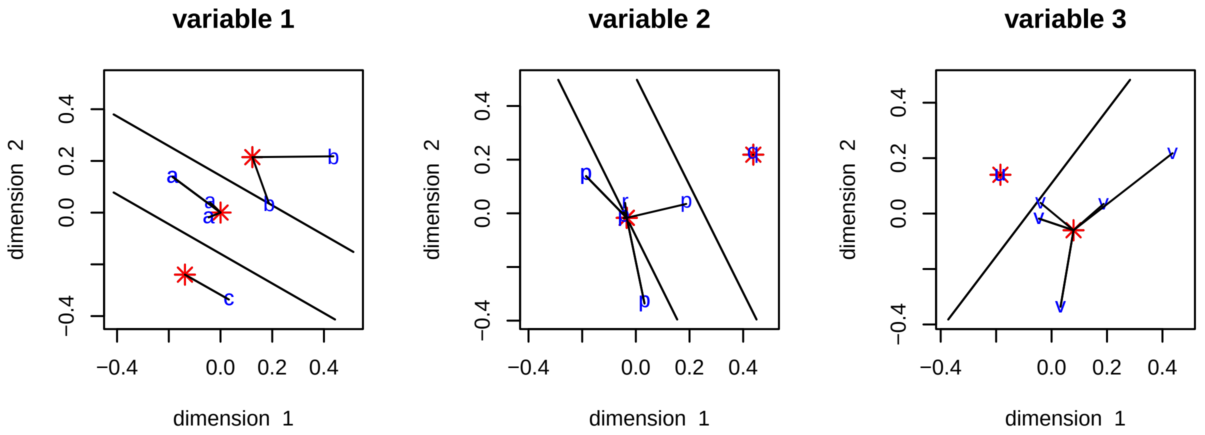


Figure 3: Small example, smacofHO rank one

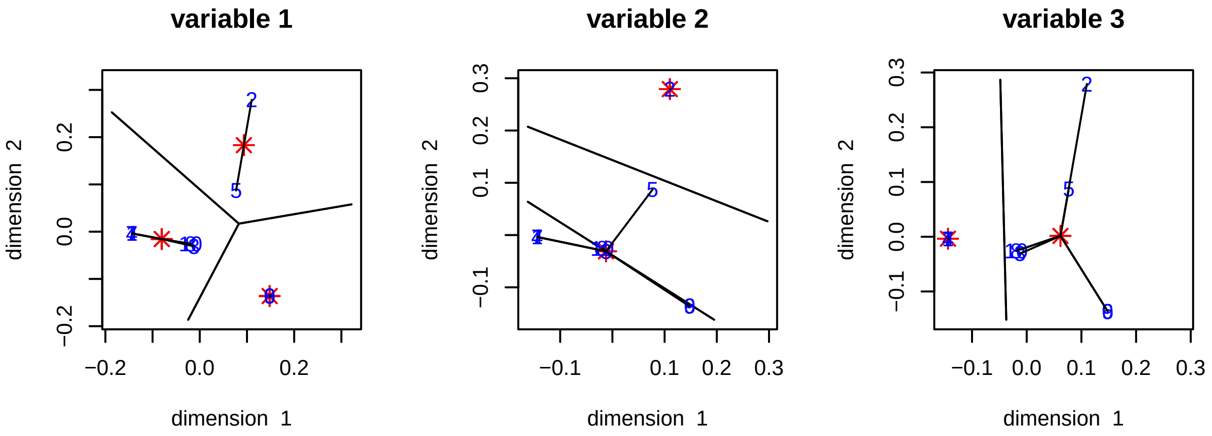


Figure 4: Small example, smacofHC centroid

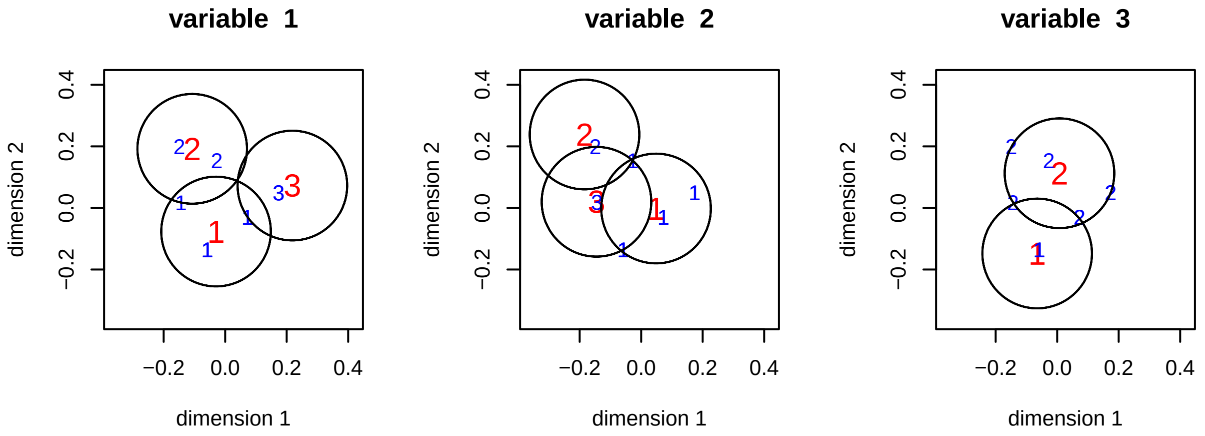


Figure 5: Small example, smacofHS circular

9.2 Cetacea

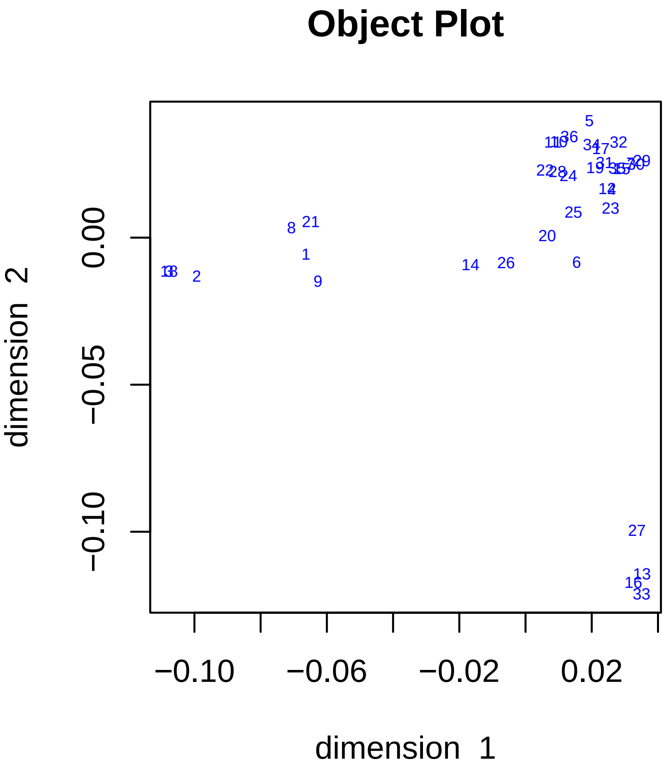


Figure 6: Cetacea Homogeneity Analysis

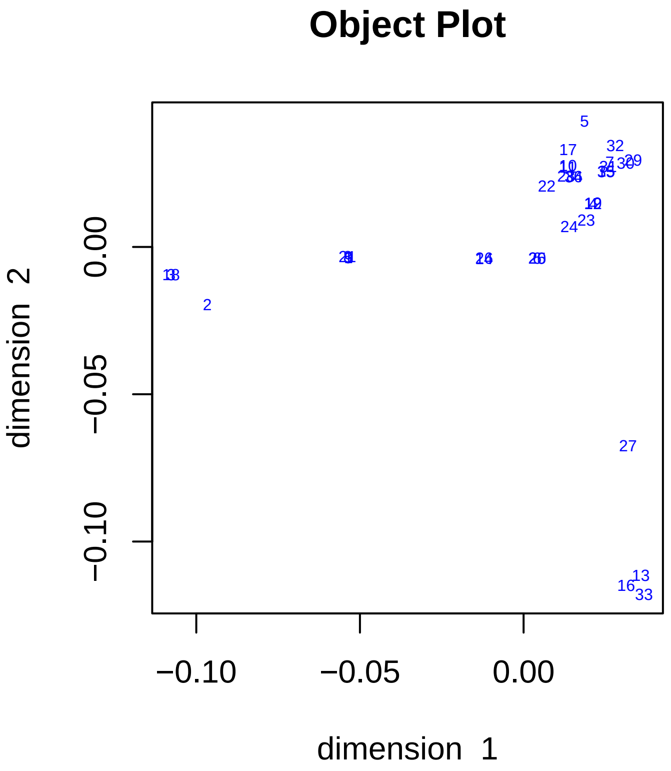


Figure 7: Cetacea smacofHO unrestricted

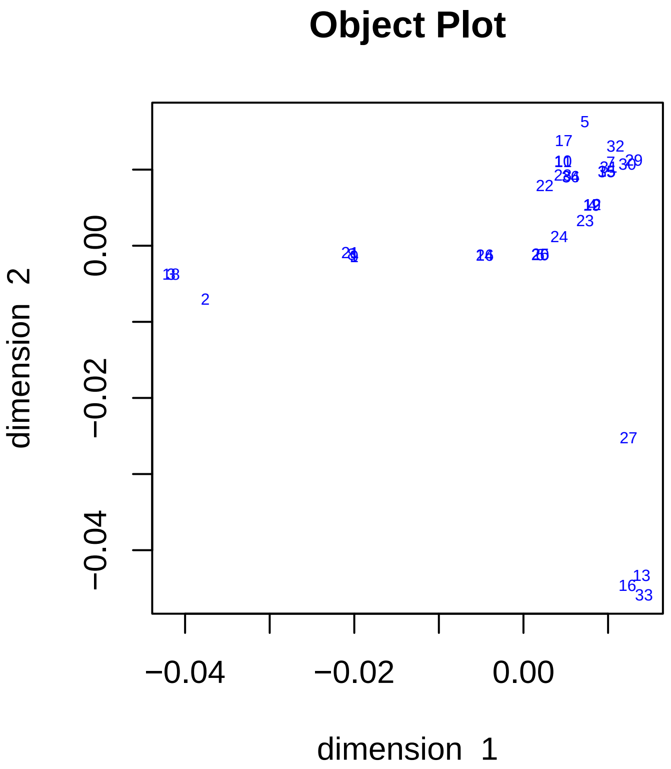


Figure 8: Cetacea smacofHO normed

9.3 Senate

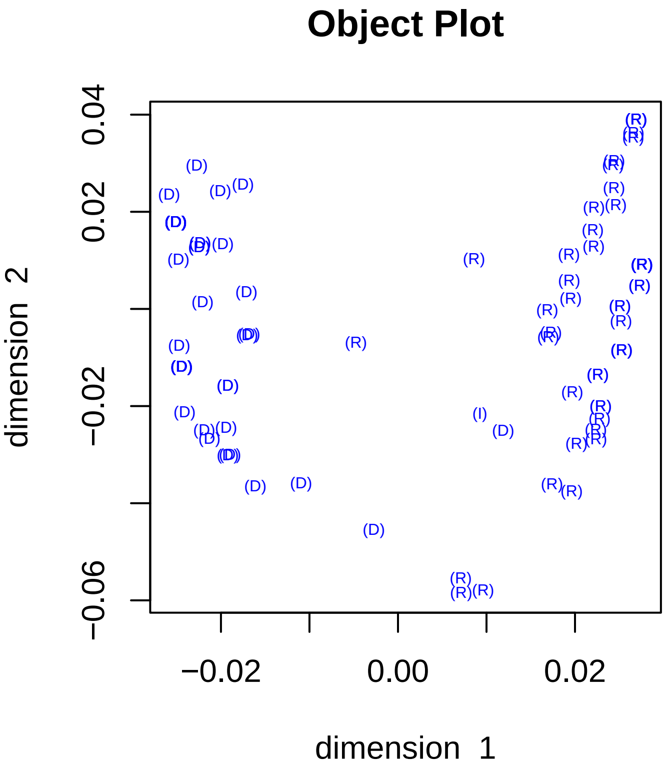


Figure 9: Senate Homogeneity Analysis

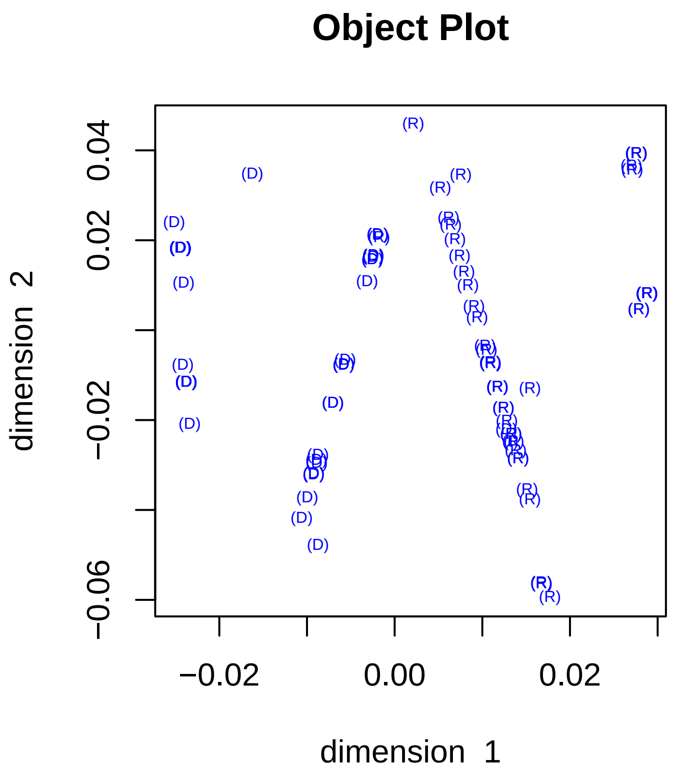


Figure 10: Senate smacofHO Unrestricted

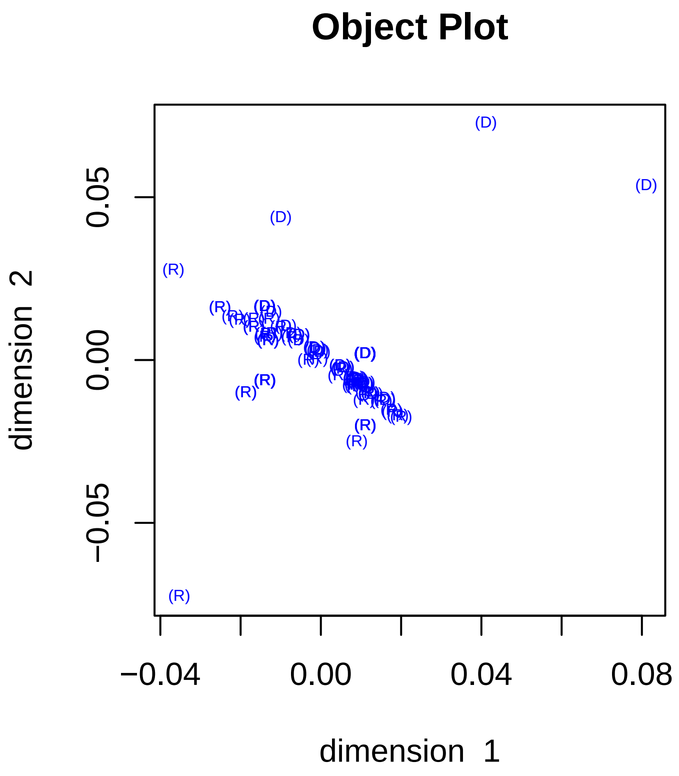


Figure 11: Senate smacofHC Centroid

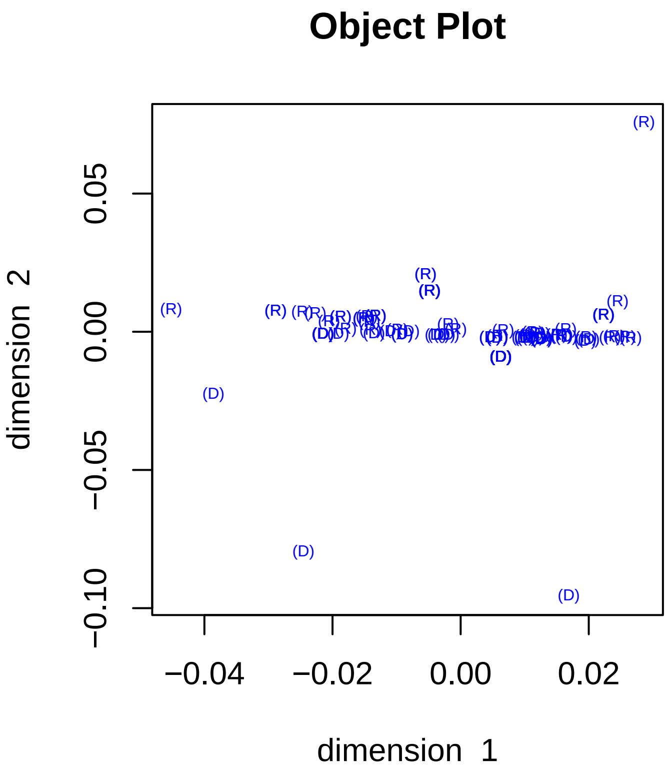


Figure 12: Senate smacofHC Centroid

References

Busing, Frank M. T. A. 2010. "Advances in Multidimensional Unfolding." PhD thesis, Leiden University.

De Leeuw, J. 1923. "Deconstructing Multiple Correspondence Analysis." In *Analysis of Categorical Data from Historical Perspectives. Essays in Honour of Shizuhiko Nishisato.*, edited by Eric J. Beh, Rosaria Lombardo, and Jose G. Clavel, 383–407. Springer.

———. 1969. "Some Contributions to the Analysis of Categorical Data." Research Note 004-69. Leiden, The Netherlands: Department of Data Theory FSW/RUL.

———. 1977. "Correctness of Kruskal's Algorithms for Monotone Regression with Ties." *Psychometrika* 42: 141–44.

———. 2003. "Homogeneity Analysis Using Euclidean Minimum Spanning Trees."

———. 2004. "Homogeneity Analysis of Pavings." Preprint Series 389. Los Angeles, CA: UCLA Department of Statistics.

———. 2014. "History of Nonlinear Principal Component Analysis." In *The Visualization and Verbalization of Data*, edited by J. Blasius and M. Greenacre. Chapman; Hall.

De Leeuw, J., and W. J. Heiser. 1980. "Multidimensional Scaling with Restrictions on the Configuration." In *Multivariate Analysis, Volume V*, edited by P. R. Krishnaiah, 501–22. Amsterdam, The Netherlands: North Holland Publishing Company.

De Leeuw, J., and P. Mair. 2009. "Homogeneity Analysis in R: the Package homals." *Journal of Statistical Software* 31 (4): 1–21. https://www.jstatsoft.org/v31/i04/.

Gifi, A. 1980. *Niet-Lineaire Multivariate Analyse [Nonlinear Multivariate Analysis]*. Leiden, The Netherlands: Department of Data Theory FSW/RUL.

———. 1990. *Nonlinear Multivariate Analysis*. New York, N.Y.: Wiley.

Guttman, L. 1941. "The Quantification of a Class of Attributes: A Theory and Method of Scale Construction." In *The Prediction of Personal Adjustment*, edited by P. Horst, 321–48. New York: Social Science Research Council.

———. 1985. "Multidimensional Structuple Analysis (MSA-i) for the Classification ofCetacea: Whales, Porpoises and Dolphins." In *Data Analysis in Real Life Environment: Ins and Outs of Solving Problems*, edited by J.-F. Marcotorchino, J. M. Proth, and J. Janssen. North-Holland.

Heiser, W. J., and J. De Leeuw. 1979. "Metric Multidimensional Unfolding." *Methoden En Data Nieuwsbrief SWS/VVS* 4: 26–50.

Hoffman, D. L., and J. De Leeuw. 1992. "Interpreting Multiple Correspondence Analysis as a Multidimensional Scaling Method." *Marketing Letters* 3: 259–72.

Kruskal, J. B. 1964a. "Multidimensional Scaling by Optimizing Goodness of Fit to a Nonmetric Hypothesis." *Psychometrika* 29: 1–27.

———. 1964b. "Nonmetric Multidimensional Scaling: a Numerical Method." *Psychometrika* 29: 115–29.

Kruskal, J. B., and J. D. Carroll. 1969. "Geometrical Models and Badness of Fit Functions." In *Multivariate Analysis, Volume II*, edited by P. R. Krishnaiah, 639–71. North Holland Publishing Company.

Lingoes, J. C. 1968a. "The Multivariate Analysis Of Qualitative Data." *Multivariate Behavioral Research* 3 (1): 61–94.

———. 1968b. "The Rationale of the Guttman-Lingoes Nonmetric Series: A Letter to Doctor

Philip Runkel." *Multivariate Behavioral Research* 3 (4): 495–507.

———. 1972. "A General Survey of the Guttman-Lingoes Nonmetric Program Series." In *Multidimensional Scaling. Theory and Applications in the Behavioral Siences, Volume i, Theory*, edited by R. N. Shepard, A. K. Romney, and S. B. Nerlove, 52–68. Seminar Press.

———. 1973. *The Guttman-Lingoes Nonmetric Program Series*. Mathesis Press.

———. 1979. "Additional Programs in the Guttman-Lingoes Nonmetric Program Series: An Update." In *Geometric Representations of Relational Data*, edited by J. C. Lingoes, Roskam E. E., and I. Borg, 283–87. Mathesis Press.

Marcotorchino, J.-F., J. M. Proth, and J. Janssen, eds. 1985. *Data Analysis in Real Life Environment: Ins and Outs of Solving Problems*. North-Holland.

Michailidis, G., and J. De Leeuw. 1998. "The Gifi System for Descriptive Multivariate Analysis." *Statistical Science* 13: 307–36.

Roskam, E. E. 1968. "Metric Analysis of Ordinal Data in Psychology." PhD thesis, University of Leiden.

Van der Burg, E. 1985. "HOMALS Classification of Whales, Porpoises and Dolphins." In *Data Analysis in Real Life Environment: Ins and Outs of Solving Problems*, edited by J.-F. Marcotorchino, J. M. Proth, and J. Janssen. North-Holland.

Van Deun, Katrijn. 2005. "Degeneracies in Multidimensional Scaling." PhD thesis, KU Leuven.

Vescia, G. 1985. "Descriptive Classification of Cetacea: Whales, Porpoises and Dolphins." In *Data Analysis in Real Life Environment: Ins and Outs of Solving Problems*, edited by J.-F. Marcotorchino, J. M. Proth, and J. Janssen, 7–13. North-Holland.

Zvulun, Eli. 1978. "Multidimensional Scalogram Analysis:the Method and Its Application." In *Theory Construction and Data Analysis in the Behavioral Sciences*, edited by Samuel Shye, 237–64. Jossey-Bass.