SymScal: Symbolic Multidimensional Scaling of Interval Dissimilarities

P.J.F. Groenen* S. Winsberg† O. Rodríguez‡
E. Diday§

March 30, 2005

Econometric Institute Report EI 2005-15

Abstract

Multidimensional scaling aims at reconstructing dissimilarities between pairs of objects by distances in a low dimensional space. However, in some cases the dissimilarity itself is unknown, but the range of the dissimilarity is given. Such fuzzy data fall in the wider class of symbolic data (Bock & Diday, 2000). Denœux and Masson (2002) have proposed to model an interval dissimilarity by a range of the distance defined as the minimum and maximum distance between two rectangles representing the objects. In this paper, we provide a new algorithm called SymScal that is based on iterative majorization. The advantage is that each iteration is guaranteed to improve the solution until no improvement is possible. In a simulation study, we investigate the quality of this algorithm. We discuss the use of SymScal on empirical dissimilarity intervals of sounds.

Keywords: Multidimensional scaling, Symbolic data analysis, Iterative majorization, Distance smoothing, SymScal.

*Econometric Institute, Erasmus University Rotterdam, P.O. Box 1738, 3000 DR Rotterdam, The Netherlands (e-mail: groenen@few.eur.nl)

†IRCAM, 1 Place Igor Stravinsky, Paris 75004, France (e-mail: winsberg@ircam.fr)

‡CIMPA-PIMAD, Escuela de Matemática, Universidad de Costa Rica, 2060 San José, Costa Rica (e-mail: oldemar.rodriguez@predisoft.com)

§CEREMADE, Université Paris Dauphine, F-75775 Paris Cédex 16, France (e-mail: diday@ceremade.dauphine.fr)

1 Introduction

Classical multidimensional scaling, MDS, represents the dissimilarities among a set of objects as distances between points in a low dimensional space. The aim of these MDS methods is to reveal relationships among the objects and uncover the dimensions giving rise to the space. Many of the applications of MDS revolve around the analysis of proximity data collected in studies related to the social sciences or to fields like product marketing and development (e.g., the objects studied may be acoustical sounds, consumer products, etc.). The goal in these studies is to visualize the objects and the distances among them and to discover the dimensions underlying the dissimilarity ratings.

Sometimes the proximity data are collected for n objects yielding a single dissimilarity matrix with the entry for the i-th row and the j-th column being the dissimilarity between the i-th and j-th object (with $i = 1, \ldots, n$ and $j = 1, \ldots, n$). Techniques for analyzing this form of data (two way one mode) have been developed by Kruskal (1964a, 1964b), see also Winsberg and Carroll (1989) or Borg and Groenen (1997). Sometimes the proximity data are collected from K sources such as a panel of K judges or under K different conditions, yielding three way two mode data and an $n \times n \times K$ array. Techniques have been developed to deal with this form of data permitting the study of individual or group differences underlying the dissimilarity ratings (see, for example, Carroll, 1972; Carroll & Winsberg, 1995; Winsberg & DeSoete, 1993; Winsberg & De Soete, 1997).

All of these MDS techniques require that each entry of the dissimilarity matrix be a single numerical value. However, it may be interesting to collect dissimilarity data where the dissimilarity between object i and object j is fuzzy then it might be represented by an interval of values rather than a single value; consequently the ij-th entry of the $n \times n$ dissimilarity matrix is an interval of values, $[\delta_{ij}^{(L)}, \delta_{ij}^{(U)}]$, where $\delta_{ij}^{(U)}$ in the upper bound of the interval, and $\delta_{ij}^{(L)}$ is the lower bound. For example, it may be that a judge prefers to indicate a range of values of dissimilarity to express the difference between object i and object j. The extent of the range may vary from object pair to object pair, because he or she wishes to capture the relative precision of his individual judgment. Note that this feature cannot be reflected by a single value of dissimilarity. Or it may be that the objects in the set under consideration are of such a complex nature that the dissimilarity between each pair of them is better represented by a range or an interval of values, rather than a single value. In this latter case, the dissimilarity between object i and object j is intrinsically unrepresentable by a single value so it

must be represented by an interval of values. In addition, if the number of objects under study becomes very large, it may be unreasonable to collect pairwise dissimilarities from each judge and one may wish to aggregate the ratings from many judges where each judge has rated the dissimilarities from a subset of all the pairs. In such cases, rather than using an average value of dissimilarity for each object pair the researcher may wish to retain the information contained in the range of dissimilarities obtained for each pair of objects. In all of these circumstances, the resulting entry of the $n \times n$ dissimilarity matrix would be an interval of values $[a, b]$ corresponding to $[\delta_{ij}^{(L)}, \delta_{ij}^{(U)}]$ rather than a single value. Of course, if a given entry of the matrix was single-valued, it could be represented in interval form as $[a, a]$.

Denœux and Masson (2002) have developed an MDS technique that treats dissimilarity matrices composed of interval data. This technique yields a representation in which each object is represented by a hyperbox (hypercube) in a low dimensional space. They have used a hypersphere representation as well (see also Masson & Denœux, 2002). In this paper, we develop a technique which yields a representation of the objects as hyperboxes in a low-dimensional Euclidean space rather than hyperspheres because the hyperbox representation is reflected as a conjunction of p properties where p is the dimensionality of the space.

This representation as a conjunction is appealing for two reasons. The first is linguistic. In everyday language, if we have objects consisting of repeated sound bursts differing with respect to loudness and the number of bursts per second, a given sound or a group of sounds might be referred to as having a loudness lying between 2 and 3 dbSPR and a repetition rate corresponding to between 300 and 400 milliseconds between bursts, that is, as a conjunction of two properties. We would not refer to a sound as a hypersphere with a loudness and repetition rate centered at 2.5 dbSPR and 350 msec and a radius of a given value to be expressed in just what units. We note that perceptually a sound might not have a precise loudness or repetition rate to a listener. Second, since one of the principal aims of MDS is to reveal relationships among the objects in terms of the underlying dimensions, it is most useful for this type of data to express the location of each object in terms of a range of each of these underlying attributes or dimensions.

We are able to represent the results of our MDS analyses in two ways. The first is a plot for each pair of dimensions displaying each object as a rectangle. The second is a graph for each underlying dimension displaying the location and range for each object on that dimension.

The representation of an object as a conjunction of properties fits within a data analytic framework concerning symbolic data and symbolic objects,

which has proven useful in dealing with large databases. More and more we are required to conduct a statistical analysis on a huge data set. In fact, these data sets may be so large that it is necessary to preprocess the data by classifying or reorganizing it into classifications or classes where the number of classes is much smaller than the number of individuals in the original data set. Then, the resulting data set, after the preprocessing will most likely contain symbolic data rather than classical data values. We refer to symbolic data when instead of having a specific, or single value for an observed variable, an observed value, for a given variable, say y_j, may be multivalued (for example, $y_j = \{16, 21, 35, 40\}$ or $y_j = \{\text{yellow, white, pink}\}$), it may be interval-valued (e.g., $y_j = [10, 20]$), or it may be modal-valued (e.g., $y_j = \{1$ with probability 0.1, 0 with probability $0.9\}$). For example, if one is dealing with fuzzy data where the observed variable(s) are represented by an interval of values, then this data would be symbolic data. For any of such symbolic data, it may be inappropriate to use existing data analytic techniques developed for single-valued data.

An inherent part of symbolic data analysis is the definition of a symbolic object. A symbolic object is a model for an entity, which could be either a concept or it might be an individual from the real world. This symbolic object is equipped with the means to compare its description to that of an individual observation. It is defined by (i) a description D, say loudness $[2, 3]$ dbSPR and repetition rate between bursts of $[300, 400]$ msec (ii) a binary relation R, allowing the comparison of the description with the entire set of descriptions, e.g., R may be one of the relations in $\{=, \equiv, \leq, \subseteq\}$, and (iii) a function, or mapping, a, providing a means of evaluating the result of the comparison, (using R), of the description of an individual in the set of all individuals Ω, to the description D. The extent of a symbolic object, a, is the set of individuals who fit the description. A symbolic object is defined by the triple $s = (a, R, D)$, where a depends on the relation R and the description D. Since interval data are a type of symbolic data, we could have a symbolic object s be defined as $a(\omega) = [\text{loudness}(\omega) \in [2 \text{ dbSPR}, 3 \text{ dbSPR}]] \wedge [\text{repetition rate}(\omega) \in [500 \text{ msec}, 600 \text{ msec}]]$.

In this paper, we will focus above all on interval-type symbolic data; that is on fuzzy dissimilarity data represented by an interval of values. For a detailed description of symbolic objects and symbolic data we refer to Bock and Diday (2000). The remainder of this paper is organized as follows. In the next section, we propose the SymScal algorithm for MDS of interval dissimilarities based on iterative majorization. Then, we investigate the quality of this algorithm by looking at the local minimum problem and a small simulation study. We also discuss an empirical data set on interval dissimilarities of sounds. We end the paper with a discussion of the results and conclusions.

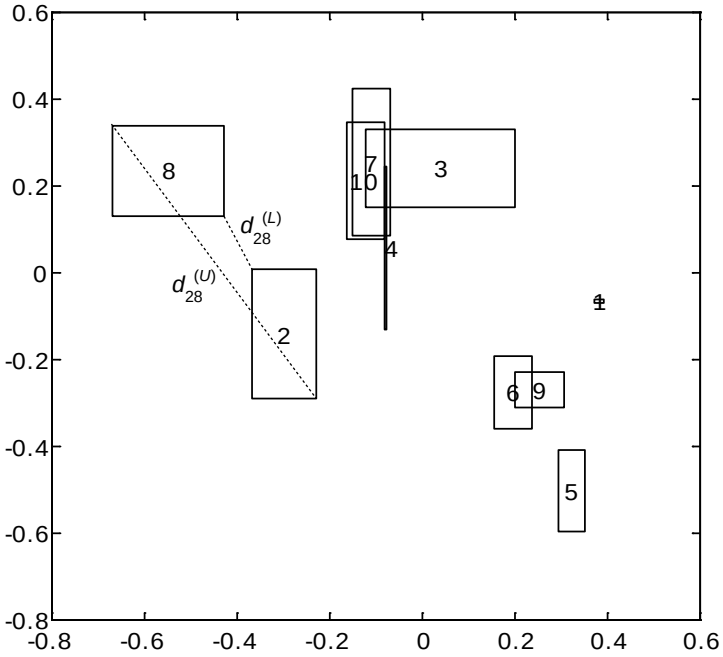

Figure 1: Example of distances in MDS for interval dissimilarities where the objects are represented by rectangles.

2 MDS of Interval Dissimilarities

To develop MDS for interval dissimilarities, the ranges of dissimilarities must be represented by ranges of distances. Here, we choose to represent the objects by rectangles and approximate the upper bound of the dissimilarity by the maximum distance between the rectangles and the lower bound by the minimum distance between the rectangles. Figure 1 shows an example of rectangle representation and how the minimum and maximum distance between two rectangles is defined.

Not only the distances are represented by ranges, the coordinates themselves are also ranges. Let the rows of the $n \times p$ matrix $\mathbf{X}$ contain the coordinates of the center of the rectangles, where n is the number of objects and p the dimensionality. The distance from the center of rectangle i along axis s, denoted the spread, is represented by r_{is}. Note that $r_{is} \geq 0$. The maximum Euclidean distance between rectangles i and j is given by

$$d_{ij}^{(U)}(\mathbf{X}, \mathbf{R}) = \left(\sum_{s=1}^{p} [|x_{is} - x_{js}| + (r_{is} + r_{js})]^2 \right)^{1/2} \tag{1}$$

and the minimum Euclidean distance by

$$d_{ij}^{(L)}(\mathbf{X},\mathbf{R}) = \left(\sum_{s=1}^{p} \max[0, |x_{is}-x_{js}| - (r_{is}+r_{js})]^2\right)^{1/2}. \tag{2}$$

Even though Euclidean distances are used between the hyperboxes, the lower and upper distances change when the solution is rotated. The reason is that the hyperboxes are are defined with respect to the axes. For a dimensional interpretation, the property rotational uniqueness can be seen as an advantage of symbolic MDS. Of course, if $\mathbf{R}$ all hyperboxes shrink to points and then symbolic MDS simplifies into ordinary MDS, which can be freely rotated.

The objective of symbolic MDS for interval dissimilarities is to represent the lower and upper bounds of the dissimilarities by minimum and maximum distances between rectangles as well as possible in least-squares sense. The Stress-Sym loss function that models this objective and needs to be minimized over $\mathbf{X}$ and $\mathbf{R}$ is given by

$$\begin{aligned}\sigma_{\text{sym}}^2(\mathbf{X},\mathbf{R}) &= \sum_{i<j}^{n} w_{ij}\left[\delta_{ij}^{(U)} - d_{ij}^{(U)}(\mathbf{X},\mathbf{R})\right]^2 \\ &\quad + \sum_{i<j}^{n} w_{ij}\left[\delta_{ij}^{(L)} - d_{ij}^{(L)}(\mathbf{X},\mathbf{R})\right]^2,\end{aligned} \tag{3}$$

where $\delta_{ij}^{(U)}$ is the upper bound of the dissimilarity of objects i and j, $\delta_{ij}^{(L)}$ is the lower bound , and w_{ij} is a given nonnegative weight.

Below, we derive a majorization algorithm called SymScal to minimize Stress-Sym for two reasons. First, iterative majorization is guaranteed to reduce Stress-Sym in each iteration from any starting configuration until a stationary point is obtained that, in practice, almost always coincides with a local minimum. Second, as in each iteration the algorithm operates on a quadratic function in $\mathbf{X}$ and $\mathbf{R}$ it is easy to impose constraints that have well known solutions for quadratic functions. This property can be useful for extensions of SymScal that require constraints.

2.1 A majorization algorithm

In this section, we develop a majorization algorithm to minimize (3) over $\mathbf{X}$ and $\mathbf{R}$. The basic idea of iterative majorization is that the original loss function is replaced in each iteration by an auxiliary function that is easier to handle. The auxiliary function, the so called majorizing function, needs to satisfy two requirements: (i) the majorizing function is equal to the original function at the current estimate, and (ii) the majorizing function is always

larger than or equal to the original function. Usually, the majorizing function is linear or quadratic so that the minimum of the majorizing function can be calculated easily. From the requirements it can be derived that (a) the loss of the majorizing function and the original loss function is equal at the current estimate, (b) at the update the majorizing function is smaller than at the current estimate, so that (c) the original loss function is smaller at the update since the original loss function is never larger than the majorizing function. This reasoning proves that if the conditions (i) and (ii) are satisfied, the iterative majorization algorithm yields a series of nonincreasing function values. For more details on iterative majorization, we refer to De Leeuw (1994), Heiser (1995), Kiers (2002) or, for an introduction, to Borg and Groenen (1997, Chapter 8) and Hunter and Lange (2004).

In what follows, we shall derive inequalities to find a majorizing function of $\sigma^2_{\text{sym}}(\mathbf{X}, \mathbf{R})$. To do this, we expand (3) as

$$\begin{aligned}\sigma^2_{\text{sym}}(\mathbf{X}, \mathbf{R}) = \\ \sum_{i<j}^{n} w_{ij}[\delta_{ij}^{(U)}]^2 + \sum_{i<j}^{n} w_{ij}[d_{ij}^{(U)}(\mathbf{X}, \mathbf{R})]^2 - 2\sum_{i<j}^{n} w_{ij}\delta_{ij}^{(U)} d_{ij}^{(U)}(\mathbf{X}, \mathbf{R}) \\ \sum_{i<j}^{n} w_{ij}[\delta_{ij}^{(L)}]^2 + \sum_{i<j}^{n} w_{ij}[d_{ij}^{(L)}(\mathbf{X}, \mathbf{R})]^2 - 2\sum_{i<j}^{n} w_{ij}\delta_{ij}^{(L)} d_{ij}^{(L)}(\mathbf{X}, \mathbf{R}). \qquad (4)\end{aligned}$$

Because $w_{ij}, \delta_{ij}^{(U)}$, and $\delta_{ij}^{(L)}$ are nonnegative, (4) can be considered as a weighted sum of $[d_{ij}^{(U)}(\mathbf{X}, \mathbf{R})]^2$, $-d_{ij}^{(U)}(\mathbf{X}, \mathbf{R})$, $[d_{ij}^{(L)}(\mathbf{X}, \mathbf{R})]^2$, and $-d_{ij}^{(L)}(\mathbf{X}, \mathbf{R})$. Thus, to find a majorizing function for (4) it suffices to find a majorizing function for each of the distance terms. In Appendix A, we derive for each of these terms a quadratic or linear majorizing function in the parameters. Once we have obtained the four majorizing functions, these majorizing functions are substituted in (4) which gives the overall majorizing function of $\sigma^2_{\text{sym}}(\mathbf{X}, \mathbf{R})$ that is quadratic in the parameters $\mathbf{X}$ and $\mathbf{R}$. We shall derive majorizing functions of the form $\alpha(x_{is}-x_{js})^2 - 2\beta(x_{is}-x_{js})(y_{is}-y_{js}) + \gamma$ for the center coordinates $\mathbf{X}$ (where $\mathbf{Y}$ is a known current estimate of $\mathbf{X}$). For the width parameters $\mathbf{R}$ we look for majorizing functions of the form $\alpha r_{is}^2 - 2\beta r_{is} + \gamma$. We shall prove that the majorization algorithm automatically finds values of $\mathbf{R}$, (where $\mathbf{Q}$ is the current estimate of $\mathbf{R}$) that are nonnegative.

There are four terms in (4) that need to be majorized. The majorization of each of these terms is discussed in Appendix A.

2.2 Combining the majorization results

In the first appendix to this paper, we have derived majorizing functions for each of the terms in (4). To obtain an overall majorizing function, we combine the majorizing results from Appendix A with (4). Substituting each of the terms in (4) by (23), (28), (32), and (36) gives the overall majorizing inequality

$$\begin{aligned}
\sigma^2_{\text{sym}}(\mathbf{X},\mathbf{R}) \;\leq\; & \sum_{s=1}^{p}\sum_{i<j}(\alpha^{(1)}_{ijs}+\alpha^{(3)}_{ij})(x_{is}-x_{js})^2 \\
& -2\sum_{s=1}^{p}\sum_{i<j}(\beta^{(1)}_{ijs}+\beta^{(3)}_{ijs}+\beta^{(5)}_{ijs})(x_{is}-x_{js})(y_{is}-y_{js}) \\
& +\sum_{s=1}^{p}\sum_{i<j}(\alpha^{(2)}_{ijs}+\alpha^{(4)}_{ijs}+\alpha^{(5)}_{ijs})r^2_{is} \\
& +\sum_{s=1}^{p}\sum_{i<j}(\alpha^{(2)}_{jis}+\alpha^{(4)}_{jis}+\alpha^{(5)}_{jis})r^2_{js} \\
& -2\sum_{s=1}^{p}\sum_{i<j}(\beta^{(2)}_{ijs}+\beta^{(4)}_{ijs})(r_{is}+r_{js}) \\
& +\sum_{s=1}^{p}\sum_{i<j}(\gamma^{(1)}_{ijs}+\gamma^{(2)}_{ijs}). \qquad (5)
\end{aligned}$$

We first consider the terms that are linear and quadratic in $\mathbf{X}$. Let $\mathbf{A}^{(1)}_s$ be a matrix with elements

$$\begin{aligned}
a^{(1)}_{ijs} &= -(\alpha^{(1)}_{ijs}+\alpha^{(3)}_{ij}) \text{ if } i\neq j \qquad (6)\\
a^{(1)}_{iis} &= -\sum_{j\neq i} a^{(1)}_{ijs}. \qquad (7)
\end{aligned}$$

Note that $\alpha^{(1)}_{ijs}$ and $\alpha^{(3)}_{ij}$ are only dependent on the known current estimates $\mathbf{Y}$ and $\mathbf{Q}$. It may be verified that

$$\sum_{s=1}^{p}\sum_{i<j}(\alpha^{(1)}_{ijs}+\alpha^{(3)}_{ij})(x_{is}-x_{js})^2=\sum_{s=1}^{p}\mathbf{x}'_s\mathbf{A}^{(1)}_s\mathbf{x}_s.$$

In a similar way, the linear term can be easily written in matrix algebra by defining matrix $\mathbf{B}^{(1)}_s$ with elements

$$\begin{aligned}
b^{(1)}_{ijs} &= -(\beta^{(1)}_{ijs}+\beta^{(3)}_{ijs}+\beta^{(5)}_{ijs}) \text{ if } i\neq j \qquad (8)\\
b^{(1)}_{iis} &= -\sum_{j\neq i} b_{ijs} \qquad (9)
\end{aligned}$$

so that

$$\sum_{s=1}^{p}\sum_{i<j}(\beta_{ijs}^{(1)}+\beta_{ijs}^{(3)}+\beta_{ijs}^{(5)})(x_{is}-x_{js})(y_{is}-y_{js})=\sum_{s=1}^{p}\mathbf{x}_s'\mathbf{B}_s^{(1)}\mathbf{y}_s.$$

We now turn to a compact expression for the terms with r_{is}^2 and r_{is}. The first thing to notice is that

$$\sum_{s=1}^{p}\sum_{i<j}(\alpha_{ijs}^{(2)}+\alpha_{ijs}^{(4)}+\alpha_{ijs}^{(5)})r_{is}^2+\sum_{s=1}^{p}\sum_{i<j}(\alpha_{jis}^{(2)}+\alpha_{jis}^{(4)}+\alpha_{jis}^{(5)})r_{js}^2=$$
$$\sum_{s=1}^{p}\sum_{i=1}^{n}r_{is}^2\sum_{j\neq i}(\alpha_{ijs}^{(2)}+\alpha_{ijs}^{(4)}+\alpha_{ijs}^{(5)}).$$

Let $\mathbf{A}_s^{(2)}$ be a diagonal matrix with diagonal elements $a_{iis}^{(2)}=\sum_{j\neq i}(\alpha_{ijs}^{(2)}+\alpha_{ijs}^{(4)}+\alpha_{ijs}^{(5)})$, so that

$$\sum_{s=1}^{p}\sum_{i=1}^{n}r_{is}^2\sum_{j\neq i}(\alpha_{ijs}^{(2)}+\alpha_{ijs}^{(4)}+\alpha_{ijs}^{(5)})=\sum_{s=1}^{p}\mathbf{r}_s'\mathbf{A}_s^{(2)}\mathbf{r}_s$$

where $\mathbf{r}_s$ is column s of $\mathbf{R}$. For the terms linear in r_{is}, we note that

$$\sum_{s=1}^{p}\sum_{i<j}(\beta_{ijs}^{(2)}+\beta_{ijs}^{(4)})(r_{is}+r_{js})=\sum_{s=1}^{p}\sum_{i=1}^{s}r_{is}\sum_{j\neq i}(\beta_{ijs}^{(2)}+\beta_{ijs}^{(4)}).$$

Define $\mathbf{b}_s^{(2)}$ as a vector with elements $\sum_{j\neq i}(\beta_{ijs}^{(2)}+\beta_{ijs}^{(4)})$ so that we may write

$$\sum_{s=1}^{p}\sum_{i=1}^{n}r_{is}\sum_{j\neq i}(\beta_{ijs}^{(2)}+\beta_{ijs}^{(4)})=\sum_{s=1}^{p}\mathbf{r}_s'\mathbf{b}_s^{(2)}.$$

The reformulation above allows us to write the right hand side of (5) as

$$\begin{aligned}\sigma_{\text{sym}}^2(\mathbf{X},\mathbf{R}) \;\leq\; & \sum_{s=1}^{p}(\mathbf{x}_s'\mathbf{A}_s^{(1)}\mathbf{x}_s-2\mathbf{x}_s'\mathbf{B}_s^{(1)}\mathbf{y}_s)\\ & +\sum_{s=1}^{p}(\mathbf{r}_s'\mathbf{A}_s^{(2)}\mathbf{r}_s-2\mathbf{r}_s'\mathbf{b}_s^{(2)})+\sum_{s=1}^{p}\sum_{i<j}(\gamma_{ijs}^{(1)}+\gamma_{ijs}^{(2)}). \quad (10)\end{aligned}$$

From (10), a dimensionwise update for $\mathbf{X}$ and $\mathbf{R}$ can be derived. Since the right hand side of (10) is quadratic in $\mathbf{X}$, a minimum is obtained by equating the gradient to zero, i.e.,

$$2\mathbf{A}_s^{(1)}\mathbf{x}_s-2\mathbf{B}_s^{(1)}\mathbf{y}_s=\mathbf{0},\ \text{or, equivalently}\ \mathbf{A}_s^{(1)}\mathbf{x}_s=\mathbf{B}_s^{(1)}\mathbf{y}_s. \quad (11)$$

To solve the linear system (11) we need a generalized inverse of $\mathbf{A}_s^{(1)}$, since it is not of full rank. We use the Moore-Penrose inverse $\mathbf{A}_s^{(1)^+}$ which equals here

$\mathbf{A}_s^{(1)^+} = (\mathbf{A}_s^{(1)} + n^{-1}\mathbf{1}\mathbf{1}')^{-1} - n^{-1}\mathbf{1}\mathbf{1}'$ with $\mathbf{1}$ a vector of ones of appropriate length. The update for $\mathbf{x}_s$ is defined by

$$\mathbf{x}_s = \mathbf{A}_s^{(1)^+}\mathbf{B}_s^{(1)}\mathbf{y}_s. \tag{12}$$

The update for $\mathbf{r}_s$ is found in a similar fashion as

$$\mathbf{r}_s = \mathbf{A}_s^{(2)^{-1}}\mathbf{b}_s^{(2)}, \text{ or, equivalently } r_{is} = \frac{b_{is}^{(2)}}{a_{iis}^{(2)}}. \tag{13}$$

The restriction on r_{is} is that it is nonnegative. It may be verified that all terms that make up $a_{iis}^{(2)}$ and $b_{is}^{(2)}$ are nonnegative, so that (13) automatically yields an update that is also nonnegative thereby satisfying the restrictions.

2.3 The SymScal Algorithm

The SymScal algorithm based on iterative majorization can be described as follows.

```
 1  Set X_0 to some initial matrix for the coordinate centers and
    set R_0 to some matrix of nonnegative values for the width.
    Set iteration counter k := 0. Set X_{-1} := X_0 and R_{-1} := R_0.
    Set the convergence criterion ε to a small positive value, for
    example, 10^{-6}.
 2  While σ²_sym(X_{k-1}, R_{k-1}) − σ²_sym(X_{k-1}, R_{k-1}) ≤ ε or k = 0
 3      k := k + 1
 4      Set Y := X_{k-1} and Q := R_{k-1}.
 5      For s = 1 to p
 6          Compute A_s^{(1)} by (7) and B_s^{(1)} by (9).
 7          Compute the update of x_s by (12).
 8          Compute A_s^{(2)} and b_s^{(2)}.
 9          Compute the update of r_s by (13).
10      End for
11      Set X_k := X and R_k := R.
12  End
```

3 Investigating the Quality of the Majorization Algorithm

To validate the SymScal algorithm, we have analyzed several artificial data sets. First, we investigated the local minimum problem. The algorithm

permits considering both the rational start described in the Appendix B and many random starts and then chooses the best global solution. We generated artificial data with random values from the uniform distribution for x from 0 to 1 for each coordinate, and random values from the uniform distribution for r from 0 to 0.2 for each coordinate. Upper and lower values of the distance were calculated from (1) and (2) and these values were used as the upper bound of the dissimilarities $\delta_{ij}^{(U)}$ and the lower bound $\delta_{ij}^{(L)}$. If there are no local minima and only a single global minimum, then the majorization algorithm should be able to find zero σ^2_{sym} to recover these data perfectly. In this study, we chose $n = 20$ and $p = 2$. To see how bad the local minimum problem is, we performed 1000 random starts on these perfect data and additionally the rational start from Appendix B.

The results of the study are shown in the histograms of Figure 2, where the left panel shows the entire distribution of σ^2_{sym} values and the right panel zooms in on those values smaller than .15. From this figure, we see that a lot of different local minima were found. Apparently, the loss function of σ^2_{sym} has a severe local minimum problem. However, the majority of the σ^2_{sym} values are close to zero indicating that the majorization algorithm combined with the random start strategy is indeed able to reconstruct perfect data. The rational start yielded Stress-Sym values close to the global minimum of zero. In the sequel, we shall apply the majorization algorithm always in conjunction with the multistart strategy (that is the rational start combined with random starts) to avoid bad fitting local minima. Preliminary experimentation suggested that about 50 random starts is usually enough to circumvent local minima problems, when using the multistart strategy.

In the second study, we investigate how well our algorithm performs for a variety of artificial data sets. These data sets were generated artificially for two values of n ($n = 10, 20$), for three error levels (no error, error variance 5% of the total variance, and error variance 15% of the total variance), and for two values of p ($p = 2, 3$). All of these factors were crossed in our study and 10 data sets were generated for each combination using the following procedure: random values of $\mathbf{X}$ and $\mathbf{R}$ were generated as described above, upper and lower values for the distance were calculated using (1) and (2), and then normally distributed error was added to these distance values to obtain the dissimilarity matrix. To ensure that the lower and upper bounds of the dissimilarities were nonnegative and that the lower bound is not larger than the upper bounds, we enforced the restrictions $0 \leq \delta_{ij}^{(L)} \leq \delta_{ij}^{(L)}$ for all ij.

To measure the ability of our algorithm to recover the true object configuration, we computed the root mean square deviation between the true and

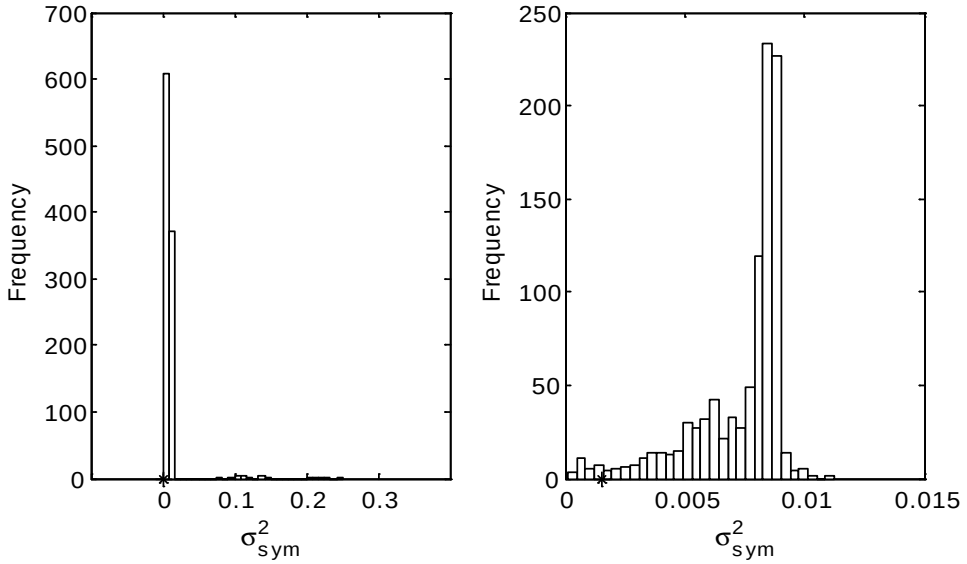

Figure 2: Distribution of σ^2_{sym} obtained by 1000 random starts followed by the majorization algorithm for symbolic MDS. The '*' indicates the position of σ^2_{sym} obtained by the rational start discussed in the appendix followed by the majorization algorithm.

recovered coordinates. This measure called DEL, is defined for the centers of the objects, by

$$\text{DELX} = [(np)^{-1} \sum_{i=1}^{n} \sum_{s=1}^{p} (\hat{x}_{is} - x_{is})^2]^{1/2}, \tag{14}$$

where $\hat{x}_{is}$ is the true value of the center for object i for the s-th coordinate, and x_{is} is the recovered value and p is the number of dimensions. We also define a measure DEL for the spreads of the objects by,

$$\text{DELR} = [(np)^{-1} \sum_{i=1}^{n} \sum_{s=1}^{p} (\hat{r}_{is} - r_{is})^2]^{1/2} \tag{15}$$

Finally we define a measure DEL for the recovery of size of the objects by

$$\text{DELSIZE} = \left[n^{-1} \sum_{i=1}^{n} \left[\prod_{s=1}^{p} (2\hat{r}_{is}) - \prod_{s=1}^{p} (2r_{is}) \right]^2 \right]^{1/2} \tag{16}$$

A DEL value of 0 is clearly the ideal and indicates perfect recovery.

To measure how well the method is able to reconstruct the true upper and lower bounds of the dissimilarities ($\hat{d}_{ij}^{(U)}$ and $\hat{d}_{ij}^{(L)}$), we calculate Tucker's

Table 1: Recovery values for the simulations. Each cell has 10 replications. For these replications, the mean and the standard deviation (within parentheses) are reported.

error	n	p	$\lambda^{(L)}$		$\lambda^{(U)}$		DELX		DELR		DELSIZE $\times 10^{-7}$	
.00	10	2	.9987	(.0020)	.9998	(.0002)	.0062	(.0086)	.0013	(.0016)	101.2	(92.9)
.00	10	3	.9988	(.0008)	.9998	(.0001)	.0183	(.0141)	.0034	(.0020)	8.2	(9.9)
.00	20	2	.9998	(.0001)	.9999	(.0000)	.0003	(.0003)	.0001	(.0001)	7.6	(13.8)
.00	20	3	.9985	(.0015)	.9998	(.0002)	.0101	(.0104)	.0013	(.0013)	3.2	(3.4)
.05	10	2	.9949	(.0034)	.9989	(.0005)	.0047	(.0050)	.0019	(.0015)	319.1	(325.5)
.05	10	3	.9956	(.0022)	.9987	(.0002)	.0194	(.0121)	.0050	(.0018)	9.4	(7.2)
.05	20	2	.9987	(.0003)	.9996	(.0001)	.0003	(.0001)	.0003	(.0001)	49.3	(45.7)
.05	20	3	.9972	(.0012)	.9993	(.0002)	.0085	(.0075)	.0015	(.0007)	3.4	(3.4)
.15	10	2	.9903	(.0046)	.9964	(.0009)	.0118	(.0092)	.0037	(.0018)	554.0	(491.5)
.15	10	3	.9911	(.0038)	.9963	(.0009)	.0272	(.0173)	.0093	(.0034)	16.0	(9.6)
.15	20	2	.9967	(.0008)	.9987	(.0003)	.0009	(.0006)	.0007	(.0002)	103.0	(43.8)
.15	20	3	.9944	(.0019)	.9983	(.0004)	.0107	(.0069)	.0032	(.0010)	7.2	(4.2)

coefficient of congruence, $\lambda^{(U)}$ and $\lambda^{(L)}$ of the true and reconstructed vectors of upper and lower bounds. For the upper bounds, Tucker's coefficient of congruence, $\lambda^{(U)}$ is defined by

$$\lambda^{(U)} = \frac{\sum_{i<j} \hat{d}_{ij}^{(U)} d_{ij}^{(U)}}{[\sum_{i<j} (\hat{d}_{ij}^{(U)})^2 \sum_{i<j} (d_{ij}^{(U)})^2]^{1/2}}. \tag{17}$$

Note that Tucker's coefficient of congruence can be viewed as a correlation coefficient without subtracting the mean. Therefore, this measure is always between -1 and 1, but since distances are positive by definition, $\lambda^{(U)}$ and $\lambda^{(L)}$ will be between 0 and 1.

The results of the simulation study are presented in Table 1. Each value in the table is the average of 10 simulations for that case, followed by in parentheses the standard deviation obtained. We note that for DELSIZE one cannot compare values obtained for two dimensions with those obtained for three dimensions without realizing that errors in the r_{is}'s are multiplied when determining the size of an object.

The values in the table indicate excellent recovery of the true values using our algorithm. We note that for example Tucker's coefficient of congruence decreases slightly when error is increased as would be expected. Also the lower bounds appear to be recovered better than the upper bounds. DELX and DELR and consequently DELSIZE also increase slightly with increasing error. All measures of fit improve with increasing n. DELX and DELR are

both recovered better for the 2 dimensional case than for the 3 dimensional case.

4 Symbolic MDS for Empirical Data

We now consider two real data sets where the entries in the dissimilarity matrix are an interval of values. The objects in the study are ten sounds differing with respect to only two physical parameters: the spectral center of gravity and log attack time. Previous studies of musical timbre have shown that these two physical parameters are highly correlated with the perceptual axes found when dissimilarity judgments are collected for sounds from different musical instruments playing the same note at the same loudness for the same duration. Until about 35 years ago timbre was considered to be a perceptual parameter of sound that was complex and multidimensional, defined primarily by what it was not, that is, what distinguishes two sounds presented in a similar manner equal in pitch, subjective duration, and loudness (see Plomp, 1970). MDS studies have shown that these two attributes of sound, namely spectral center of gravity and log attack time explain the factors we use to distinguish, say, middle C on the piano from middle C on some other instrument (see, e.g., McAdams, Winsberg, Donnadieu, De Soete, & Krimphoff, 1995; McAdams & Winsberg, 1999). So when middle C is sounded on the piano the sound has some unidimensional attributes such as pitch, corresponding to the frequency of the fundamental, loudness, and duration. In addition, it is characterized by its timbre, that is, it is a note from a piano not some other instrument. This last attribute is perceptually multidimensional with two important underlying dimensions related to spectral center of gravity and log attack time. The spectral center of gravity is the weighted average of the harmonics generated when the note is sounded averaged over the duration of the tone with a running time window of, say, 12ms and is higher say for the harpsichord than it is for the piano. The log attack time is the logarithm of the rise time measured from the time the amplitude envelope reaches a threshold of 2% of the maximum amplitude to the time it takes to reach the maximum amplitude and is longer for a wind instrument like the trumpet than it is for a string instrument like the harp. The sounds in this study were artificially generated to represent the range of values found in natural instruments according to the design in Figure 3. The data represents dissimilarity judgments from the same expert listener taken on two occasions, see Table 2. On each occasion the expert listened to each pair of sounds and indicated a range of dissimilarity for each pair on a slider scale going from very similar to very different.

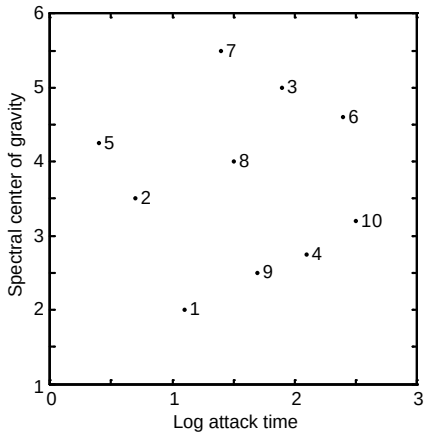

Figure 3: Design of the ten sounds according to spectral center of gravity (vertical axis) and log attack time (horizontal axes).

Table 2: Interval dissimilarities of ten sounds judged by an expert at occasions 1 (lower triangle) and 2 (upper triangle).

Tone	1	2	3	4	5	6	7	8	9	10
1	[–, –]	[62,81]	[82,95]	[6,22]	[62,87]	[58,87]	[67,81]	[64,77]	[0,13]	[48,61]
2	[73, 88]	[–, –]	[39,59]	[53,68]	[3,16]	[68,92]	[17,41]	[45,57]	[40,66]	[92,98]
3	[93,100]	[6,21]	[–, –]	[48,74]	[51,78]	[0, 8]	[0,11]	[23,56]	[46,69]	[33,61]
4	[7, 25]	[46,66]	[60,72]	[–, –]	[51,68]	[17,41]	[72,92]	[44,55]	[0,20]	[31,42]
5	[95,100]	[4,36]	[38,58]	[63,74]	[–, –]	[34,54]	[0, 5]	[3,23]	[68,79]	[45,70]
6	[73, 90]	[37,63]	[16,22]	[33,46]	[1, 8]	[–, –]	[9,26]	[8,37]	[42,54]	[22,54]
7	[90,100]	[49,71]	[4,13]	[87,98]	[10,21]	[28,46]	[–, –]	[21,42]	[47,77]	[77,91]
8	[64, 79]	[7,36]	[10,26]	[36,54]	[26,45]	[28,50]	[32,60]	[–, –]	[54,79]	[18,33]
9	[0, 8]	[37,63]	[58,78]	[8,19]	[66,81]	[71,84]	[76,90]	[29,46]	[–, –]	[3,18]
10	[35, 44]	[78,88]	[49,75]	[0, 7]	[69,82]	[65,81]	[75,91]	[75,88]	[20,53]	[–, –]

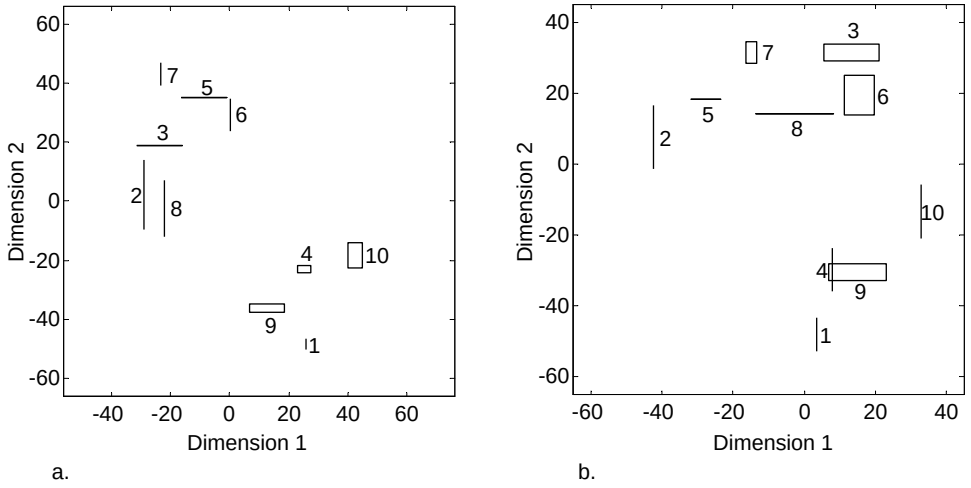

Figure 4: SymScal solutions for the sound data in Table 2. Panel (a) gives the results for Occasion 1 with Stress-sym .02861128 and Panel (b) for Occasion 2 with Stress-sym .04893295.

The data in Table 2 where analyzed by SymScal for both occasions separately using 1000 random starts. The resulting solutions are given in Figure 4. Visual examination of the two solutions reveals the following. The horizontal axis represents log attack time and the vertical axes the spectral center of gravity. Without imposing any restrictions, SymScal seems to be able to reconstruct the physical space. The results for the second occasion in Figure 4b reflect the physical space quite well. Notice the groupings 10, 9, 4, 1 and 2, 5, 7 and 3, 6, 8 reflect well how these stimuli are grouped in the physical space. Moreover the relation of these groups to one another approximates their disposition in the physical space reasonably well. However, the solution from the first occasion shows some deviations from the physical space: 8, 3, 6 are too far to the left, 3 is too low, 7 is too far to the left, and 1 is too far to the right. It is interesting to note that these differences from one occasion to another are greater than the range of uncertainty reflected in the solutions. The improved results on the second occasion indicate that the task is better performed with some practice and with greater familiarity with the group of sounds. It also appears from the figures that sounds with long attack times are more difficult to localize. This type of data with the SymScal solution might possibly be used to establish norms which could then be used to detect people with specific types of hearing impairment.

5 Discussion and Conclusions

We have presented an MDS technique for symbolic data that deals with fuzzy dissimilarities consisting of an interval of values observed for each pair of objects. In this technique each object is represented as a hyperbox in a p dimensional space. By representing the objects as hypercubes, we are able to convey information contained when the dissimilarity between the objects or for any object pair needs to be expressed as a range of values not a single value. It may be so, moreover, that the precision inherent in the dissimilarities is such that the precision in one recovered dimension is worse than that for the other dimensions. Our technique is able to tease out and highlight this kind of information.

We proposed the SymScal algorithm for doing symbolic MDS of interval dissimilarities. This algorithm is based on iterative majorization. The advantage is that each iteration yields better Stress-sym until no improvement is possible. Simulation studies have shown that SymScal combined with multiple random start and a rational start yields good quality solutions.

Denœux and Masson (2002) discuss an extension that allows the upper and lower bounds to be transformed. Although it is technically feasible to do so in our case, we do not believe that transformations are useful for symbolic MDS with interval or histogram data. The reason is that by having the available information of a given interval for each dissimilarity, it seems unnatural to destroy this information. Therefore, we recommend to apply symbolic MDS without any transformation and perform it directly on the upper and lower bounds.

Moreover, we are easily able to extend our method to deal with the case in which the dissimilarity between object i and object j is an empirical distribution of values or, equivalently, a histogram. For example, we may have enough detailed information so that we can represent the empirical distribution of the dissimilarity as a histogram, for example by $.10[0,1], .30[1,2], .40[2,3], .20[3,4]$, where the first number indicates the relative frequency and values between the brackets define the bin. Then, each object is represented in the MDS plane by a series of embedded rectangles, one for each bin.

As the SymScal algorithm is based on iterative majorization, each majorizing function is quadratic in the parameters. Therefore, restrictions as symbolic MDS for histogram data or the extension of symbolic MDS to three-way data (by, for example, the weighted Euclidean model) can be easily derived combined with the SymScal algorithm. We intend to pursue these extensions in future publications.

A Majorizing the Terms in Stress-Sym

A.1 Majorizing $[d_{ij}^{(U)}(\mathbf{X},\mathbf{R})]^2$

The square of the upper bound of the distance $[d_{ij}^{(U)}(\mathbf{X},\mathbf{R})]^2$ can be written as

$$[d_{ij}^{(U)}(\mathbf{X},\mathbf{R})]^2 = \sum_{s=1}^{p}[(x_{is}-x_{js})^2 + (r_{is}+r_{js})^2 + 2|x_{is}-x_{js}|(r_{is}+r_{js})]. \quad (18)$$

The term $(x_{is}-x_{js})^2$ is standard in MDS and is quadratic in $\mathbf{X}$. The product $|x_{is}-x_{js}|(r_{is}+r_{js})$ can be seen as the product of two functions, i.e., a_1a_2. Consider the following inequality:

$$\left(\frac{a_1}{b_1}-\frac{a_2}{b_2}\right)^2 \geq 0 \quad (19)$$

with strict equality if $a_1 = b_1$ and $a_2 = b_2$. Expanding (19) gives

$$\begin{aligned} 0 &\leq \frac{a_1^2}{b_1^2}+\frac{a_2^2}{b_2^2}-2\frac{a_1a_2}{b_1b_2} \\ 2\frac{a_1a_2}{b_1b_2} &\leq \frac{a_1^2}{b_1^2}+\frac{a_2^2}{b_2^2} \\ 2a_1a_2 &\leq \frac{b_2}{b_1}a_1^2+\frac{b_1}{b_2}a_2^2. \end{aligned} \quad (20)$$

Note that (20) only holds when $b_1 > 0$ and $b_2 > 0$. If $b_1 = 0$ (or $b_2 = 0$) then we shall replace it by a small ε. Note that this adaptation violates restriction (i) of the requirements of a majorizing function, but by making ε small enough, it should not distort the convergence properties of the majorizing algorithm. In the sequel, we shall use this adaptation implicitly whenever necessary.

Let $\mathbf{Y}$ and $\mathbf{Q}$ be the current known estimates of $\mathbf{X}$ and $\mathbf{R}$. Then, substituting $|x_{is}-x_{js}|$ for a_1, $(r_{is}+r_{js})$ for a_2, $|y_{is}-y_{js}|$ for b_1, $(q_{is}+q_{js})$ for b_2 into (20) gives

$$\begin{aligned} 2|x_{is}-x_{js}|(r_{is}+r_{js}) &\leq \frac{q_{is}+q_{js}}{|y_{is}-y_{js}|}(x_{is}-x_{js})^2 \\ &\quad +\frac{|y_{is}-y_{js}|}{(q_{is}+q_{js})}(r_{is}+r_{js})^2. \end{aligned} \quad (21)$$

To get rid of the crossproduct term $2r_{is}r_{js}$ in $(r_{is}+r_{js})^2$, we apply (20) again:

$$2r_{is}r_{js} \leq \frac{q_{js}}{q_{is}}r_{is}^2+\frac{q_{is}}{q_{js}}r_{js}^2. \quad (22)$$

Combining (18), (21), and (22), and multiplying by w_{ij} gives the majorizing inequality

$$w_{ij}[d_{ij}^{(U)}(\mathbf{X},\mathbf{R})]^2 \leq \sum_{s=1}^{p}[\alpha_{ijs}^{(1)}(x_{is}-x_{js})^2 + \alpha_{ijs}^{(2)}r_{is}^2 + \alpha_{jis}^{(2)}r_{js}^2]. \tag{23}$$

with $\alpha_{ijs}^{(1)} = w_{ij}[1+(q_{is}+q_{js})/|y_{is}-y_{js}|]$ and $\alpha_{ijs}^{(2)} = w_{ij}[|y_{is}-y_{js}|+(q_{is}+q_{js})]/q_{is}$.

A.2 Majorizing $-d_{ij}^{(U)}(\mathbf{X},\mathbf{R})$

Minus the upper bound of the distance, $-d_{ij}^{(U)}(\mathbf{X},\mathbf{R})$, can be written as

$$-d_{ij}^{(U)}(\mathbf{X},\mathbf{R}) = -\left(\sum_{s=1}^{p}[|x_{is}-x_{js}| + (r_{is}+r_{js})]^2\right)^{1/2}. \tag{24}$$

This term is concave in the parameters $\mathbf{X}$ and $\mathbf{R}$ and hence can be majorized by a linear function in $\mathbf{X}$ and $\mathbf{R}$. Consider the Cauchy-Schwarz inequality $\|\mathbf{a}\|\|\mathbf{b}\| \geq \mathbf{a}'\mathbf{b}$. Dividing both sides of the inequality by $-\|\mathbf{b}\|$ (assuming that $\|\mathbf{b}\| > 0$) gives

$$-\|\mathbf{a}\| \leq \begin{cases} -\mathbf{a}'\mathbf{b}/\|\mathbf{b}\| & \text{if } \|\mathbf{b}\| > 0, \\ 0 & \text{if } \|\mathbf{b}\| = 0. \end{cases} \tag{25}$$

Note that the inequality $-\|\mathbf{a}\| \leq 0$ if $\|\mathbf{b}\| = 0$ does not violate any of the majorization requirements. Applying (25) to $-d_{ij}^{(U)}(\mathbf{X},\mathbf{R})$ gives

$$-d_{ij}^{(U)}(\mathbf{X},\mathbf{R}) \leq \begin{cases} -\frac{\sum_{s=1}^{p}[|x_{is}-x_{js}|+(r_{is}+r_{js})][|y_{is}-y_{js}|+(q_{is}+q_{js})]}{d_{ij}^{(U)}(\mathbf{Y},\mathbf{Q})} & \text{if } d_{ij}^{(U)}(\mathbf{Y},\mathbf{Q}) > 0, \\ 0 & \text{if } d_{ij}^{(U)}(\mathbf{Y},\mathbf{Q}) = 0. \end{cases} \tag{26}$$

Applying (25) to the single term $-|x_{is}-x_{js}|$ gives

$$-|x_{is}-x_{js}| \leq \begin{cases} -\frac{(x_{is}-x_{js})(y_{is}-y_{js})}{|y_{is}-y_{js}|} & \text{if } |y_{is}-y_{js}| > 0, \\ 0 & \text{if } |y_{is}-y_{js}| = 0 \end{cases} \tag{27}$$

(Kiers & Groenen, 1996). Combining the results of (26) and (27) and multiplying by $w_{ij}\delta_{ij}^{(U)}$ gives the majorizing inequality

$$-w_{ij}\delta_{ij}^{(U)}d_{ij}^{(U)}(\mathbf{X},\mathbf{R}) \leq -\sum_{s=1}^{p}[\beta_{ijs}^{(1)}(x_{is}-x_{js})(y_{is}-y_{js}) + \beta_{ijs}^{(2)}(r_{is}+r_{js})] \tag{28}$$

with

$$\beta_{ijs}^{(1)} = \begin{cases} \frac{w_{ij}\delta_{ij}^{(U)}[|y_{is}-y_{js}|+(q_{is}+q_{js})]}{|y_{is}-y_{js}|d_{ij}^{(U)}(\mathbf{Y},\mathbf{Q})} & \text{if } |y_{is}-y_{js}|>0 \text{ and } d_{ij}^{(U)}(\mathbf{Y},\mathbf{Q})>0, \\ 0 & \text{if } |y_{is}-y_{js}|=0 \text{ or } d_{ij}^{(U)}(\mathbf{Y},\mathbf{Q})=0, \end{cases}$$

$$\beta_{ijs}^{(2)} = \begin{cases} \frac{w_{ij}\delta_{ij}^{(U)}[|y_{is}-y_{js}|+(q_{is}+q_{js})]}{d_{ij}^{(U)}(\mathbf{Y},\mathbf{Q})} & \text{if } d_{ij}^{(U)}(\mathbf{Y},\mathbf{Q})>0, \\ 0 & \text{if } d_{ij}^{(U)}(\mathbf{Y},\mathbf{Q})=0. \end{cases}$$

A.3 Majorizing $[d_{ij}^{(L)}(\mathbf{X},\mathbf{R})]^2$

To majorize $[d_{ij}^{(L)}(\mathbf{X},\mathbf{R})]^2$, we start by considering a majorizing function for $\max[0,|x_{is}-x_{js}|-(r_{is}+r_{js})]^2$. Let $a_1=|x_{is}-x_{js}|, a_2=r_{is}+r_{js}, b_1=|y_{is}-y_{js}|$, and $b_2=q_{is}+q_{js}$. Then $\max[0,|x_{is}-x_{js}|-(r_{is}+r_{js})]^2=\max[0,a_1-a_2]^2$. To simplify notation even further, we use $a=a_1-a_2$ and $b=b_1-b_2$ so that $\max[0,a_1-a_2]^2=\max[0,a]^2$. To majorize $\max[0,a]^2$, we need to consider two cases, i.e., $b\geq 0$ and $b<0$. For $b\geq 0$ the function $\max[0,a]^2=a^2$, so that $\max[0,a]^2$ can be majorized by a^2. For $b<0$, the function $\min[0,a]^2$ can be majorized by the function $(a-b)^2$. Summarizing these majorization results and resubstituting gives,

$$\max[0,a_1-a_2]^2 \leq \begin{cases} (a_1-a_2)^2 & \text{if } b_1-b_2\geq 0, \\ [(a_1-a_2)-(b_1-b_2)]^2 & \text{if } b_1-b_2<0. \end{cases} \tag{29}$$

It can be verified that (29) satisfies the requirements of a majorizing function. Working out both conditions of (29) yield terms a_1^2, a_2^2, a_1, a_2, and $-2a_1a_2$. The only complicating term is the product $-2a_1a_2$ for which we derive a majorizing function. Consider the following inequalities

$$\begin{aligned} 0 &\leq [(a_1+a_2)-(b_1+b_2)]^2 \\ 0 &\leq (a_1+a_2)^2+(b_1+b_2)^2-2(a_1+a_2)(b_1+b_2) \\ 0 &\leq a_1^2+a_2^2+2a_1a_2+(b_1+b_2)^2-2(a_1+a_2)(b_1+b_2) \\ -2a_1a_2 &\leq a_1^2+a_2^2+(b_1+b_2)^2-2(a_1+a_2)(b_1+b_2). \end{aligned} \tag{30}$$

Thus, (30) is the majorization inequality for majorizing minus the product of two functions. Combining (29) and (30) gives

$$\max[0,a_1-a_2]^2 \leq \begin{cases} 2(a_1^2+a_2^2)-2(a_1+a_2)(b_1+b_2)+(b_1+b_2)^2 & \text{if } b_1\geq b_2, \\ 2(a_1^2+a_2^2)-4a_1b_1-4a_2b_2+2(b_1^2+b_2^2) & \text{if } b_1<b_2. \end{cases} \tag{31}$$

Note that (31) has terms with $-a_1$ that after resubstitution yields a term with $-|x_{is}-x_{js}|$ that can be majorized by (27). Also, a_2^2 equals after resubstitution $r_{is}^2 + r_{js}^2 + 2r_{is}r_{js}$. The crossproduct term $2r_{is}r_{js}$ is majorized by (22). Combining all results and multiplying by w_{ij} gives the following majorization inequality:

$$\begin{aligned} & w_{ij}[d_{ij}^{(L)}(\mathbf{X},\mathbf{R})]^2 \leq \\ & \quad \sum_{s=1}^{p}[\alpha_{ij}^{(3)}(x_{is}-x_{js})^2 + \alpha_{ijs}^{(4)}r_{is}^2 + \alpha_{jis}^{(4)}r_{js}^2 \\ & \quad -2\beta_{ijs}^{(3)}(x_{is}-x_{js})(y_{is}-y_{js}) - 2\beta_{ijs}^{(4)}(r_{is}+r_{js}) + \gamma_{ijs}^{(1)}] \end{aligned} \tag{32}$$

with

$$\begin{aligned} \alpha_{ij}^{(3)} &= 2w_{ij} \\ \alpha_{ijs}^{(4)} &= 2w_{ij}(1+q_{js}/q_{is}) \\ \beta_{ijs}^{(3)} &= \begin{cases} \frac{w_{ij}[|y_{is}-y_{js}|+(q_{is}+q_{js})]}{|y_{is}-y_{js}|} & \text{if } |y_{is}-y_{js}| \geq q_{is}+q_{js} \text{ and } |y_{is}-y_{js}|>0, \\ 2w_{ij} & \text{if } |y_{is}-y_{js}| < q_{is}+q_{js} \text{ and } |y_{is}-y_{js}|>0, \\ 0 & \text{if } |y_{is}-y_{js}|=0, \end{cases} \\ \beta_{ijs}^{(4)} &= \begin{cases} w_{ij}[|y_{is}-y_{js}|+(q_{is}+q_{js})] & \text{if } |y_{is}-y_{js}| \geq q_{is}+q_{js}, \\ 2w_{ij}(q_{is}+q_{js}) & \text{if } |y_{is}-y_{js}| < q_{is}+q_{js}, \end{cases} \\ \gamma_{ijs}^{(1)} &= \begin{cases} w_{ij}[|y_{is}-y_{js}|+(q_{is}+q_{js})]^2 & \text{if } |y_{is}-y_{js}| \geq q_{is}+q_{js}, \\ 2w_{ij}[|y_{is}-y_{js}|^2+(q_{is}+q_{js})^2] & \text{if } |y_{is}-y_{js}| < q_{is}+q_{js}. \end{cases} \end{aligned}$$

A.4 Majorizing $-d_{ij}^{(L)}(\mathbf{X},\mathbf{R})$

To find a majorizing function for minus the lower bound of the distance, $-d_{ij}^{(L)}(\mathbf{X},\mathbf{R})$, we can make use of the majorization inequality (25) based on Cauchy-Schwarz. This allows us to write

$$\begin{aligned} & -d_{ij}^{(L)}(\mathbf{X},\mathbf{R}) \leq \\ & \quad \begin{cases} \sum_{s=1}^{p} -\frac{\max[0,|x_{is}-x_{js}|-(r_{is}+r_{js})]\max[0,|y_{is}-y_{js}|-(q_{is}+q_{js})]}{d_{ij}^{(L)}(\mathbf{Y},\mathbf{Q})} & \text{if } d_{ij}^{(L)}(\mathbf{Y},\mathbf{Q})>0, \\ 0 & \text{if } d_{ij}^{(L)}(\mathbf{Y},\mathbf{Q})=0. \end{cases} \end{aligned} \tag{33}$$

Thus, (33) yields a function of the sum of $-\max[0,|x_{is}-x_{js}|-(r_{is}+r_{js})]$. Using the same notational simplification as in the previous subsection, this can be rewritten as $-\max[0,a_1-a_2]$. This function can be majorized as follows:

$$-\max[0,a_1-a_2] \leq \begin{cases} -(a_1-a_2) & \text{if } b_1 \geq b_2, \\ 0 & \text{if } b_1 < b_2. \end{cases} \tag{34}$$

Again we majorize the terms $-a_1$ (which is equal to $-|x_{is}-x_{js}|$ after resubstitution) by (27). The term $+a_2$ leads after resubstitution to a term $(r_{is}+r_{js})$. For algorithmic reasons it is better to only have linear terms that are negative. For this reason, we apply another majorization step to $+r_{is}$ (and r_{js}), i.e.,

$$r_{is} \leq \tfrac{1}{2}\frac{r_{is}^2}{q_{is}} + \tfrac{1}{2}q_{is}. \tag{35}$$

Combining these results and multiplication by $w_{ij}\delta_{ij}^{(L)}$ yields the majorizing function

$$\begin{aligned} &-w_{ij}\delta_{ij}^{(L)}d_{ij}^{(L)}(\mathbf{X},\mathbf{R}) \leq \\ &\quad \sum_{s=1}^{p}[\tfrac{1}{2}\alpha_{ijs}^{(5)}r_{is}^2 + \tfrac{1}{2}\alpha_{jis}^{(5)}r_{js}^2 - \beta_{ijs}^{(5)}(x_{is}-x_{js})(y_{is}-y_{js}) + \tfrac{1}{2}\gamma_{ijs}^{(2)}] \end{aligned} \tag{36}$$

with

$$\alpha_{ijs}^{(5)} = \begin{cases} \frac{w_{ij}\delta_{ij}^{(L)}\max[0,|y_{is}-y_{js}|-(q_{is}+q_{js})]}{q_{is}d_{ij}^{(L)}(\mathbf{Y},\mathbf{Q})} & \text{if } |y_{is}-y_{js}| \geq q_{is}+q_{js}, \\ 0 & \text{if } |y_{is}-y_{js}| < q_{is}+q_{js}, \end{cases}$$

$$\beta_{ijs}^{(5)} = \begin{cases} \frac{w_{ij}\delta_{ij}^{(L)}\max[0,|y_{is}-y_{js}|-(q_{is}+q_{js})]}{|y_{is}-y_{js}|d_{ij}^{(L)}(\mathbf{Y},\mathbf{Q})} & \text{if } |y_{is}-y_{js}| \geq q_{is}+q_{js} \\ & \text{and } |y_{is}-y_{js}| > 0, \\ 0 & \text{if } |y_{is}-y_{js}| < q_{is}+q_{js} \\ & \text{or } |y_{is}-y_{js}| = 0, \end{cases}$$

$$\gamma_{ijs}^{(2)} = \begin{cases} \frac{w_{ij}\delta_{ij}^{(L)}(q_{is}+q_{js})\max[0,|y_{is}-y_{js}|-(q_{is}+q_{js})]}{d_{ij}^{(L)}(\mathbf{Y},\mathbf{Q})} & \text{if } |y_{is}-y_{js}| \geq q_{is}+q_{js}, \\ 0 & \text{if } |y_{is}-y_{js}| < q_{is}+q_{js}. \end{cases}$$

B Rational Start for MDS of Interval Data

It is well known that classical scaling (Torgerson, 1958; Gower, 1966) has a duality property with standard or classical principal components analysis, PCA, when the dissimilarities are Euclidean distances. To avoid local minimum problems we propose to use a rational start, using InterScal (Rodríguez, 2000), an algorithm for MDS of interval data that produces results similar to the vertices method for PCA of interval data, developed by Chouakria, Cazes, and Diday (2000, pages 200–212). The interval data for n objects on m variables can be described by two matrices: the $n \times m$ matrices $\mathbf{H}^{(U)}$ and $\mathbf{H}^{(L)}$ with the upper and lower bounds. The trick in the vertices method is

that each object can be thought of as being somewhere in a hyperbox defined by all the intervals on the m variables. Therefore, the vertices method for PCA represents each object by all its vertices in the $2^m \times m$ matrix $\mathbf{M}_i$. An example of $\mathbf{M}_i$ with $m = 3$ variables is given by

$$\mathbf{M}_i = \begin{bmatrix} h_{i1}^{(L)} & h_{i2}^{(L)} & h_{i3}^{(L)} \\ h_{i1}^{(L)} & h_{i2}^{(L)} & h_{i3}^{(U)} \\ h_{i1}^{(L)} & h_{i2}^{(U)} & h_{i3}^{(L)} \\ h_{i1}^{(L)} & h_{i2}^{(U)} & h_{i3}^{(U)} \\ h_{i1}^{(U)} & h_{i2}^{(L)} & h_{i3}^{(L)} \\ h_{i1}^{(U)} & h_{i2}^{(L)} & h_{i3}^{(U)} \\ h_{i1}^{(U)} & h_{i2}^{(U)} & h_{i3}^{(L)} \\ h_{i1}^{(U)} & h_{i2}^{(U)} & h_{i3}^{(U)} \end{bmatrix} \tag{37}$$

The vertices method for PCA proceeds by applying standard PCA to the $2^m n \times m$ matrix $\mathbf{M}$ that has all matrices $\mathbf{M}_i$ stacked underneath each other, that is,

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_1 \\ \mathbf{M}_2 \\ \mathbf{M}_3 \\ \vdots \\ \mathbf{M}_n \end{bmatrix}.$$

It is well known that there exists a duality property between PCA and classical MDS (Gower, 1966). Let the singular value decomposition of $\mathbf{M}$ be given by $\mathbf{M} = \mathbf{P}\mathbf{\Phi}\mathbf{Q}'$ with $\mathbf{P}'\mathbf{P} = \mathbf{Q}'\mathbf{Q} = \mathbf{I}$ and $\mathbf{\Phi}$ a diagonal matrix with nonnegative singular values ordered from large to small. Then, the rank p approximation of $\mathbf{H}$ in PCA solution is given by $\mathbf{P}_p\mathbf{\Phi}_p\mathbf{Q}_p' = \mathbf{X}\mathbf{Q}_p'$, where the subscript p denotes that the first p columns are used for $\mathbf{P}$ and $\mathbf{Q}$ and the first p rows and columns of $\mathbf{\Phi}$.

To see the equivalence between classical MDS and PCA, we need a compact expression of the matrix of squared distances $\mathbf{D}$ of squared distances between the rows of $\mathbf{M}$, that is,

$$\mathbf{D} = \mathbf{a}\mathbf{1}' + \mathbf{1}\mathbf{a}' - 2\mathbf{M}\mathbf{M}', \tag{38}$$

where $\mathbf{a}$ is the vector with the diagonal elements of $\mathbf{M}\mathbf{M}'$. Let $\mathbf{J} = \mathbf{I} - n^{-1}\mathbf{1}\mathbf{1}'$ be the centering matrix so that $\mathbf{J}\mathbf{1} = \mathbf{0}$ and $\mathbf{1}'\mathbf{J} = \mathbf{0}'$. Pre- and post multiplying (38) by $\mathbf{J}$ and also multiplying by the factor $-\frac{1}{2}$ gives

$$\mathbf{M}\mathbf{M}' = -\tfrac{1}{2}\mathbf{J}\mathbf{D}\mathbf{J}. \tag{39}$$

Classical MDS minimizes $\|\mathbf{X}\mathbf{X}' - (-\frac{1}{2}\mathbf{J}\mathbf{D}\mathbf{J})\|^2$ by computing the eigendecomposition $-\frac{1}{2}\mathbf{J}\mathbf{D}\mathbf{J} = \mathbf{P}\mathbf{\Phi}^2\mathbf{P}'$ and choosing $\mathbf{X} = \mathbf{P}_p\mathbf{\Phi}_p$. The above proves that classical MDS on the matrix of squared Euclidean distances between the rows of $\mathbf{M}$ is the same as PCA on $\mathbf{M}$ directly. Classical MDS can also be used directly on a dissimilarity matrix $\mathbf{\Delta}$. In this case, the eigendecomposition of $-\frac{1}{2}\mathbf{J}\mathbf{\Delta}^{(\mathbf{2})}\mathbf{J}$ is taken, where $\mathbf{\Delta}^{(2)}$ denotes the matrix of squared dissimilarities.

To get a Symbolic MDS method that has a *duality property* with Vertices PCA, when dissimilarity is modelled by an Euclidean distance, we need as input the dissimilarities between all $2^m n$ rows of the matrix $\mathbf{M}$ defined above, because Vertices PCA effects a classical PCA of the matrix $\mathbf{M}$. Thus, we would need as input a matrix $\mathbf{\Delta}$ of size $2^m n \times 2^m n$ but it is clearly impossible to construct a matrix of this size, because we only have two dissimilarities, that is the maximum and the minimum, for each pair of objects. So it seems impossible to find a Symbolic MDS method that has a *duality property* with Vertices PCA. Therefore, we propose to find an approximate solution.

The idea, then, is to carry out an MDS of the distance matrix $\widetilde{\mathbf{\Delta}}$ defined below. For each hypercube (thus for each object i), the matrix $\widetilde{\mathbf{\Delta}}$ has two rows. In the first row, we use the minimum dissimilarity and the maximum dissimilarity among a hypercube and itself, whereas we use the dissimilarity minimum and the average dissimilarity among each different couple of hypercubes, that is to say, we use $2m$ dissimilarities. In the second row of the matrix $\widetilde{\mathbf{\Delta}}$, we use the maximum dissimilarity and the minimum dissimilarity among a hypercube and itself, and we use the average dissimilarity and the maximum dissimilarity among each different couple of hypercubes, in this row we also use $2n$ dissimilarities, but as the average dissimilarities were already employed we really use n dissimilarities, therefore for each hypercube we use $3n$ dissimilarities. Then, since $d(x, y) = d(y, x)$, in total we use $(3/2)n(n + 1) > n(n + 1)$ dissimilarities. Note that $\widetilde{\mathbf{\Delta}}$ is a symmetric matrix and its size is $2n \times 2n$. Since for each hyperbox we have two rows, we can compute a coordinate minimum and maximum, that is, coordinates

of interval type. The matrix $\widetilde{\mathbf{\Delta}}$ is given by

$$\widetilde{\mathbf{\Delta}} = \begin{bmatrix} 0 & 0 & \delta_{12}^{(L)} & \overline{\delta}_{12} & \delta_{13}^{(L)} & \overline{\delta}_{13} & \cdots & \delta_{1m}^{(L)} & \overline{\delta}_{1m} \\ 0 & 0 & \overline{\delta}_{12} & \delta_{12}^{(U)} & \overline{\delta}_{13} & \delta_{13}^{(U)} & \cdots & \overline{\delta}_{1m} & \delta_{1m}^{(U)} \\ \delta_{21}^{(L)} & \overline{\delta}_{21} & 0 & 0 & \delta_{23}^{(L)} & \overline{\delta}_{23} & \cdots & \delta_{2m}^{(L)} & \overline{\delta}_{2m} \\ \overline{\delta}_{21} & \delta_{21}^{(U)} & 0 & 0 & \overline{\delta}_{23} & \delta_{23}^{(U)} & \cdots & \overline{\delta}_{2m} & \delta_{2m}^{(U)} \\ \delta_{31}^{(L)} & \overline{\delta}_{31} & \delta_{32}^{(L)} & \overline{\delta}_{32} & 0 & 0 & \cdots & \delta_{3m}^{(L)} & \overline{\delta}_{3m} \\ \overline{\delta}_{31} & \delta_{31}^{(U)} & \overline{\delta}_{32} & \delta_{32}^{(U)} & 0 & 0 & \cdots & \overline{\delta}_{3m} & \delta_{3m}^{(U)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \delta_{m1}^{(L)} & \overline{\delta}_{m1} & \delta_{m2}^{(L)} & \overline{\delta}_{m2} & \delta_{m3}^{(L)} & \overline{\delta}_{m3} & \cdots & 0 & 0 \\ \overline{\delta}_{m1} & \delta_{m1}^{(U)} & \overline{\delta}_{m2} & \delta_{m2}^{(U)} & \overline{\delta}_{m3} & \delta_{m3}^{(U)} & \cdots & 0 & 0 \end{bmatrix}, \qquad (40)$$

where $\overline{\delta}_{ij} = (\delta_{ij}^{(L)} + \delta_{ij}^{(U)})/2$.

The InterScal Algorithm for rational start for MDS of interval-value dissimilarity data is performed as follows:

1 Obtain the dissimilarities $\delta_{ij}^{(L)}$ and $\delta_{ij}^{(U)}$ for all $i, j = 1, 2, \ldots, n$.

2 Compute the matrix $\widetilde{\mathbf{\Delta}}$ according to (40).

3 Find the matrix $\mathbf{B} = -\frac{1}{2}\mathbf{J}\widetilde{\mathbf{\Delta}}^{(2)}\mathbf{J}$ with $\mathbf{J}$ the centering matrix.

4 Find the eigenvalues $\mathbf{\Phi}^2$ and eigenvectors $\mathbf{P}$ of $\mathbf{B}$.

5 Compute the coordinates of the $2n$ points in p dimensions using the formula

$$y_{is} = p_{is}\phi_{ss} \text{ for } i = 1, 2, \ldots, 2n \text{ and } s = 1, 2, \ldots, p.$$

6 Construct the center coordinates $\mathbf{X}$ and the spread $\mathbf{R}$ of the hypercube for object i by for each dimension s

$$\begin{aligned} x_{is} &= (y_{2i,s} + y_{2i-1,s})/2, \\ r_{is} &= |y_{2i,s} - y_{2i-1,s}|/2. \end{aligned}$$

Note that the solution for $\mathbf{X}$ is not unique as $\mathbf{B} = \mathbf{P}\mathbf{\Phi}^2\mathbf{P}' = \mathbf{Y}\mathbf{T}\mathbf{T}'\mathbf{Y}'$ for any $\mathbf{T}\mathbf{T}' = \mathbf{I}$. Any rigid rotation is an example of matrix of type $\mathbf{T}$. We choose the solution corresponding to principal axes. The first axis maximizes the variance of the vertices of the hypercube. However, since any rotation is also a solution, one may wish to rotate the principal axes solution in order to obtain axes which are more interpretable.

Let us consider the special case when all the intervals of of the dissimilarities are zero, that is, $\delta_{ij}^{(L)} = \delta_{ij}^{(U)} = \delta_{ij}$. In this case, $\overline{\delta}_{ij}$ also equals δ_{ij}. Then $\widetilde{\mathbf{\Delta}}$ has blocks of 2×2 matrices for all combination ij with all four elements equal to δ_{ij}. Then, INTERSCAL is exactly equal to classical scaling on the $n \times n$ matrix of dissimilarities with elements δ_{ij}, except that each object appears twice, hence the intervals collapse to points.

References

Bock, H.-H., & Diday, E. (2000). *Analysis of symbolic data.* Berlin: Springer.

Borg, I., & Groenen, P. J. F. (1997). *Modern multidimensional scaling: Theory and applications.* New York: Springer.

Carroll, J. D. (1972). Individual differences and multidimensional scaling. In *Theory and applications in the behavioral sciences vol i, theory.* New York: Seminar Press.

Carroll, J. D., & Winsberg, S. (1995). Fitting an extended INDSCAL model to 3-way proximity data. *Journal of Classification*, *12*, 57–71.

Chouakria, A., Cazes, P., & Diday, E. (2000). Symbolic principal components analysis. In H.-H. Bock & E. Diday (Eds.), *nalysis of symbolic data* (pp. 200–212). Berlin: Springer.

De Leeuw, J. (1994). Block relaxation algorithms in statistics. In H.-H. Bock, W. Lenski, & M. M. Richter (Eds.), *Information systems and data analysis* (pp. 308–324). Berlin: Springer.

Denœux, T., & Masson, M. (2002). Multidimensional scaling of interval-valued dissimilarity data. *Pattern Recognition Letters*, *21*, 83–92.

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

Heiser, W. J. (1995). Convergent computation by iterative majorization: Theory and applications in multidimensional data analysis. In W. J. Krzanowski (Ed.), *Recent advances in descriptive multivariate analysis* (pp. 157–189). Oxford: Oxford University Press.

Hunter, D. R., & Lange, K. (2004). A autorial on MM algorithms. *The American Statistician*, *39*, 30–37.

Kiers, H. A. L. (2002). Setting up alternating least squares and iterative majorization algorithms for solving various matrix optimization problems. *Computational Statistics and Data Analysis*, *41*, 157–170.

Kiers, H. A. L., & Groenen, P. J. F. (1996). A monotonically convergent algorithm for orthogonal congruence rotation. *Psychometrika*, *61*, 375–389.

Kruskal, J. B. (1964a). Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, *29*, 1–27.

Kruskal, J. B. (1964b). Nonmetric multidimensional scaling: A numerical method. *Psychometrika*, *29*, 115–129.

Masson, M., & Denœux, T. (2002). Multidimensional scaling of fuzzy dissimilarity data. *Fuzzy Sets and Systems*, *128*, 339–352.

McAdams, S., & Winsberg, S. (1999). Multidimensional scaling of musical timbre constrained by physical parameters. *The Journal of the Acoustical Society of America*, *105*, 1273.

McAdams, S., Winsberg, S., Donnadieu, S., De Soete, G., & Krimphoff, J. (1995). Perceptual scaling of synthesized musical timbres: Common dimensions, specificities, and latent subject classes. *Psychological Research*, *58*, 177–192.

Plomp, R. (1970). Timbre as a multidimensional attribute of complex tones. In R. Plomp & G. F. Smoorenburg (Eds.), *Frequency analysis and periodicity detection in hearing* (pp. 397–414). Leiden: Sijthoff.

Rodríguez, O. (2000). *Classification et modèles linéaires en analyse des données symboliques.* Unpublished doctoral dissertation, Université Paris IX Dauphine, Paris.

Torgerson, W. S. (1958). *Theory and methods of scaling.* New York: Wiley.

Winsberg, S., & Carroll, J. D. (1989). A quasi-nonmetric method for multidimensional sclaing via an extended Euclidean model. *Psychometrika*, *54*, 217–219.

Winsberg, S., & DeSoete, G. (1993). A latent class approach to fitting the weighted euclidean model, CLASCAL. *Psychometrika*, *58*, 315–330.

Winsberg, S., & De Soete, G. (1997). Multidimensional scaling with constrained dimensions: CONSCAL. *British Journal of Mathematical and Statistical Psychology*, *50*, 55–72.