Journal of Classification 5:163-180 (1988)

Convergence of the Majorization Method for Multidimensional Scaling

Jan de Leeuw

University of California Los Angeles

Abstract: In this paper we study the convergence properties of an important class of multidimensional scaling algorithms. We unify and extend earlier qualitative results on convergence, which tell us when the algorithms are convergent. In order to prove global convergence results we use the majorization method. We also derive, for the first time, some quantitative convergence theorems, which give information about the speed of convergence. It turns out that in almost all cases convergence is linear, with a convergence rate close to unity. This has the practical consequence that convergence will usually be very slow, and this makes techniques to speed up convergence very important. It is pointed out that step-size techniques will generally not succeed in producing marked improvements in this respect.

Keywords: Multidimensional scaling; Convergence; Step size; Local minima.

1. Introduction

Recent research in multidimensional scaling has moved in the direction of proposing more and more complicated models, often with a very large number of parameters, and sometimes even with severe discontinuities in the model. The emphasis has been on producing computer programs that work, and comparatively little attention has been paid to theoretical problems associated with the loss functions and the algorithms used to minimize them. We

Author's Address: Jan de Leeuw, Departments of Psychology and Mathematics, University of California Los Angeles, 405 Hilgard Avenue, Los Angeles, CA 90024, USA.

think that such a more theoretical study is long overdue. In fact we think that at the moment an in-depth study of some of the more simple models and techniques is more urgent than the development of even more complicated ones. This paper is a contribution to the study of a simple algorithm for fitting the simplest of multidimensional scaling models.

Thus, our paper does not present a new algorithm or a new model. It also makes no claim about superiority of the algorithm studied here over other possible or actual algorithms. The model is chosen from a large spectrum of possible models, and the loss function is one among many. The only reason for studying this particular model, loss function, and algorithm is that they are simple and direct. This choice makes the mathematical study of their properties relatively easy. It is also the reason why they have been around for quite some time now, and why they have been used in a majority of the applications of multidimensional scaling.

The paper will mainly be about metric multidimensional scaling; towards the end we shall briefly discuss the nonmetric case. Results for nonmetric scaling are often simple extensions of the metric results, and in this sense metric scaling is more basic.

2. Notation and Terminology

We introduce the notation and terminology more or less standard for metric multidimensional scaling (Kruskal and Wish 1978). The data in a classical multidimensional scaling problem are collected in a symmetric non-negative matrix $\Delta = \{ \delta_{ij} \}$. The elements of Δ are called *dissimilarities*; δ_{ij} is the dissimilarity between *objects* i and j. There are n objects, and thus Δ is of order n. We suppose that self-dissimilarities are zero; as a consequence Δ has a zero diagonal. A common purpose of multidimensional scaling is to represent the objects as *points* in a low-dimensional Euclidean space, in such a way that the *distance* between points i and j is approximately equal to the given dissimilarity of objects i and j. The $\mathbf{x}_i$ denote n points in p-space, with coordinates in the $n \times p$ matrix X, called the *configuration*. The matrix D(X), with elements $d_{ij}(\mathrm{X})$, contains the Euclidean distances between the points $\mathbf{x}_i$. It follows that

$$\begin{aligned} d_{ij}^2(\mathrm{X}) &= (\mathbf{x}_i - \mathbf{x}_j)'(\mathbf{x}_i - \mathbf{x}_j) && \text{(1a)} \\ &= (\mathbf{e}_i - \mathbf{e}_j)'\,\mathrm{X}\,\mathrm{X}'\,(\mathbf{e}_i - \mathbf{e}_j) && \text{(1b)} \\ &= \mathrm{tr}\ \mathrm{X}'\,\mathrm{A}_{ij}\,\mathrm{X}\ . && \text{(1c)} \end{aligned}$$

In (1b) the $\mathbf{e}_i$ are unit vectors (columns of the identity matrix of order n), and in (1c) we have $\mathrm{A}_{ij} = (\mathbf{e}_i - \mathbf{e}_j)(\mathbf{e}_i - \mathbf{e}_j)'$. In order to find out how successful a *representation* is we compute the value of a *loss function*, defined to be

$$\sigma(\mathbf{X}) = 1/2\, \Sigma_i\, \Sigma_j\, w_{ij}\, (\delta_{ij} - d_{ij}(\mathbf{X}))^2 \ , \tag{2}$$

where $\mathbf{W} = \{w_{ij}\}$ is a symmetric, non-negative matrix of *weights*, with a zero diagonal.

Matrix **W** is known and fixed throughout the computations. Its nondiagonal elements can be used to code various forms of supplementary information. If there are missing data, for instance, we can set w_{ij} corresponding to missing dissimilarities equal to zero. If there are replications of the dissimilarities in a cell we can estimate their variability, and use this information to choose weights. It sometimes also makes sense to set w_{ij} equal to a fixed function of the dissimilarity values, such as the square or the inverse square, in order to differentially weight errors. This strategy can be used to simulate the behavior of other multidimensional scaling loss functions.

The purpose of the particular form of multidimensional scaling we have presented can now be stated more precisely: given weights, dissimilarities, and a dimensionality p, we want to find **X** that minimizes $\sigma(\mathbf{X})$.

3. Basic Algorithm

The algorithm we discuss in this paper was first given by Guttman (1968). He derived it by setting the stationary equations for the minimization of $\sigma(\mathbf{X})$ equal to zero, and he observed that the algorithm could be interpreted as a gradient algorithm with constant step-size. Compare also Lingoes and Roskam (1973, p. 8-10), Hartman (1979, p. 74-82), Borg (1981, p. 88-92). In de Leeuw (1977) the very same algorithm was derived in a somewhat more general context from convex analysis as a subgradient method. It was observed that in the simple Euclidean case, which is the one we are interested in here, the algorithm could be derived from the Cauchy-Schwartz inequality, without using either differentiation or subdifferentiation. This is the derivation we present here. It is a much simplified version of the one given in de Leeuw and Heiser (1980).

In order to describe the algorithm efficiently we need some additional notation. First let

$$\eta^2(\mathbf{X}) = 1/2\, \Sigma_i\, \Sigma_j\, w_{ij}\, d_{ij}^2(\mathbf{X}) \ , \tag{3}$$

and

$$\rho(\mathbf{X}) = 1/2\, \Sigma_i\, \Sigma_j\, w_{ij}\, \delta_{ij}\, d_{ij}(\mathbf{X}) \ . \tag{4}$$

If we assume, without loss of generality that

$$1/2\, \Sigma_i\, \Sigma_j\, w_{ij}\, \delta_{ij}^2 = 1 \ , \tag{5}$$

then

$$\sigma(\mathbf{X}) = 1 - 2\rho(\mathbf{X}) + \eta^2(\mathbf{X}) \ . \tag{6}$$

It is clear that $\eta^2(\mathbf{X})$ is a convex quadratic function of **X**. From (1c) and (3) we have

$$\eta^2(\mathbf{X}) = \text{tr } \mathbf{X}'\ \mathbf{V}\,\mathbf{X} \ , \tag{7}$$

with

$$\mathbf{V} = 1/2\, \Sigma_i\, \Sigma_j\, w_{ij}\, \mathbf{A}_{ij} \ . \tag{8}$$

Throughout the paper we assume that **V** has rank $n-1$, an assumption which can be made without any loss of generality (de Leeuw 1977). It also follows from (8) that **V** is positive semidefinite, and that its (one-dimensional) null space consists of all vectors with constant elements.

The function $\rho(\mathbf{X})$ is somewhat more complicated than $\eta^2(\mathbf{X})$. We know that $d_{ij}(\mathbf{X})$ is a convex and positively homogeneous function of **X**. Thus, by (4), the same claim holds for $\rho(\mathbf{X})$. It is convenient to write $\rho(\mathbf{X})$ as

$$\rho(\mathbf{X}) = \text{tr } \mathbf{X}'\mathbf{B}(\mathbf{X})\mathbf{X} \ , \tag{9}$$

with

$$\mathbf{B}(\mathbf{X}) = 1/2\, \Sigma_i\, \Sigma_j\, w_{ij}\, \delta_{ij}\, s_{ij}(\mathbf{X})\mathbf{A}_{ij} \ , \tag{10}$$

where

$$s_{ij}(\mathbf{X}) = 1/d_{ij}(\mathbf{X}) \ \text{ if } \ d_{ij}(\mathbf{X}) \neq 0 \tag{11a}$$

$$s_{ij}(\mathbf{X}) = 0 \ \text{ otherwise} \tag{11b}$$

It is obvious that the difference between (7) and (9) is that in (7) **V** is a constant matrix,while in (9), **B(X)** varies with **X**. Thus, $\eta^2(\mathbf{X})$ is quadratic in **X**, while $\rho(\mathbf{X})$ is not.

With the notation developed so far it is easy to explain the algorithm. Clearly the partial derivatives of $\eta^2(\mathbf{X})$ are given by $\nabla\, \eta^2(\mathbf{X}) = 2\mathbf{V}\mathbf{X}$. Using the definitions it is also not difficult to see that $\nabla\, \rho(\mathbf{X}) = \mathbf{B}(\mathbf{X})\mathbf{X}$, *provided that* $\rho(\mathbf{X})$ *is differentiable at* **X**. Thus, $\nabla\, \sigma(\mathbf{X}) = 2(\mathbf{V}\mathbf{X} - \mathbf{B}(\mathbf{X})\mathbf{X})$. In de Leeuw and Heiser (1980), the *Guttman transform* $\Gamma(\mathbf{Y})$ of a configuration **Y** is defined as

$$\Gamma(\mathbf{Y}) = \mathbf{V}^{+}\mathbf{B}(\mathbf{Y})\mathbf{Y} \ , \tag{12}$$

with $\mathbf{V}^{+}$ the Moore-Penrose inverse of $\mathbf{V}$. Observe that the Guttman transform depends on weights and dissimilarities, and is consequently defined relative to a particular metric multidimensional scaling problem. Using the Guttman transform makes it possible to rewrite the gradient as $\nabla\, \sigma(\mathbf{X}) = 2\mathbf{V}(\mathbf{X} - \Gamma(\mathbf{X}))$, and thus $\nabla\, \sigma(\mathbf{X}) = 0$ if and only if $\mathbf{X} = \Gamma(\mathbf{X})$. As a consequence, a configuration $\mathbf{X}$ is called *stationary* for a particular metric multidimensional scaling problem if it is equal to its Guttman transform. The result also immediately suggests the algorithm $\mathbf{X}_{k+1} = \Gamma(\mathbf{X}_k)$, or more explicitly,

$$\mathbf{X}_{k+1} = \mathbf{V}^{+}\mathbf{B}(\mathbf{X}_k)\mathbf{X}_k \ . \tag{13}$$

Equivalently we can also write

$$\mathbf{X}_{k+1} = \mathbf{X}_k - 1/2\, \mathbf{V}^{+}\, \nabla\, \sigma(\mathbf{X}_k) \ , \tag{14}$$

which shows the gradient interpretation of the algorithm.

In this derivation of the algorithm we have made the provision that $\sigma(\mathbf{X})$ had to be differentiable at $\mathbf{X}$. This claim is true if and only if $d_{ij}(\mathbf{X}) > 0$ for all i, j for which $w_{ij}\, \delta_{ij} > 0$. Let us agree to call a configuration *usable* if this condition is true. Thus if $\mathbf{X}$ is not usable, then $\sigma(\mathbf{X})$ is not differentiable at $\mathbf{X}$, and interpretation (14) cannot be used. Using definition (10), however, the iteration (12) can still be carried out. In de Leeuw (1977) and de Leeuw and Heiser (1980) it is shown that in this case the algorithm still converges to a stationary point.

There are three reasons why we simply *assume* differentiability in the sequel. In the first place it was proved by de Leeuw (1984), that if σ has a local minimum at $\mathbf{X}$, then $\mathbf{X}$ is usable. Since we are interested in the behavior of our algorithm in the neighborhood of a local minimum in most of this paper, we might as well assume differentiability. In the second place, although we do not need differentiability for the qualitative study of convergence (i.e., for proving that the algorithm converges), we do need it for the quantitative study (i.e., for establishing the rate of convergence). And, finally, we shall base our convergence proof below directly on the Cauchy-Schwartz inequality, for which differentiability is not needed in the first place.

The convergence proof starts with a lemma, which is the foundation of our approach to the minimization of this particular multidimensional scaling loss function.

Lemma 1: *For all* **X** *and* **Y**

$$\sigma(\mathbf{X}) \leq 1 - \eta^2(\Gamma(\mathbf{Y})) + \eta^2(\mathbf{X} - \Gamma(\mathbf{Y})) \ . \tag{15a}$$

Moreover for all **X**

$$\sigma(\mathbf{X}) = 1 - \eta^2(\Gamma(\mathbf{X})) + \eta^2(\mathbf{X} - \Gamma(\mathbf{X})) \ . \tag{15b}$$

Proof: By Cauchy-Schwartz

$$\text{tr } \mathbf{X}'\mathbf{A}_{ij}\mathbf{Y} \leq \{ \text{ tr } \mathbf{X}'\mathbf{A}_{ij}\mathbf{X}\}^{1/2} \ \{ \text{ tr } \mathbf{Y}'\mathbf{A}_{ij}\mathbf{Y} \}^{1/2} = d_{ij}(\mathbf{X})d_{ij}(\mathbf{Y}) \ . \tag{16}$$

Multiply both sides by w_{ij} δ_{ij} $s_{ij}(\mathbf{Y})$, and add over all i, j. This manipulation gives, using (4) and (10),

$$\text{tr } \mathbf{X}'\mathbf{B}(\mathbf{Y})\mathbf{Y} \leq \rho(\mathbf{X}) \ , \tag{17}$$

with equality if **Y** = **X**. We can also write (17) as

$$\rho(\mathbf{X}) \geq \text{tr } \mathbf{X}'\mathbf{V}\Gamma(\mathbf{Y}) \ , \tag{18}$$

and substitution in (6) gives

$$\sigma(\mathbf{X}) \leq 1 - 2 \text{ tr } \mathbf{X}'\mathbf{V}\Gamma(\mathbf{Y}) + \text{tr } \mathbf{X}'\mathbf{V}\mathbf{X} \ . \tag{19}$$

We can write (19) as

$$\sigma(\mathbf{X}) \leq 1 - \text{tr } \Gamma(\mathbf{Y})'\mathbf{V}\Gamma(\mathbf{Y}) + \text{tr } (\mathbf{X} - \Gamma(\mathbf{Y}))'\mathbf{V}(\mathbf{X} - \Gamma(\mathbf{Y})) \ , \tag{20}$$

which is (15a). We have equality if **X** = **Y**, which is (15b).
Q.E.D.

Figure 1 provides a useful interpretation of Lemma 1, and also shows the algorithmic implications. If we use the abbreviation

$$\omega_{\mathbf{Y}}(\mathbf{X}) = 1 - \eta^2(\Gamma(\mathbf{Y})) + \eta^2(\mathbf{X} - \Gamma(\mathbf{Y})) \ , \tag{21}$$

then for each **Y**, the function $\omega_{\mathbf{Y}}$ is quadratic in **X**. Moreover, by Lemma 1, $\sigma(\mathbf{X}) \leq \omega_{\mathbf{Y}}(\mathbf{X})$ and $\sigma(\mathbf{Y}) = \omega_{\mathbf{Y}}(\mathbf{Y})$. Thus, for each **Y**, the function $\omega_{\mathbf{Y}}$ *majorizes* the function σ, and the two functions *touch* only for $X = Y$. Moreover, $\omega_{\mathbf{Y}}$ is minimized over **X** by setting $\mathbf{X} = \Gamma(\mathbf{Y})$ and thus we see that the Guttman transform decreases the loss function. We write this result as

$$\sigma(\Gamma(\mathbf{Y})) \leq \omega_{\mathbf{Y}}(\Gamma(\mathbf{Y})) < \omega_{\mathbf{Y}}(\mathbf{Y}) = \sigma(\mathbf{Y}) \ , \tag{22}$$

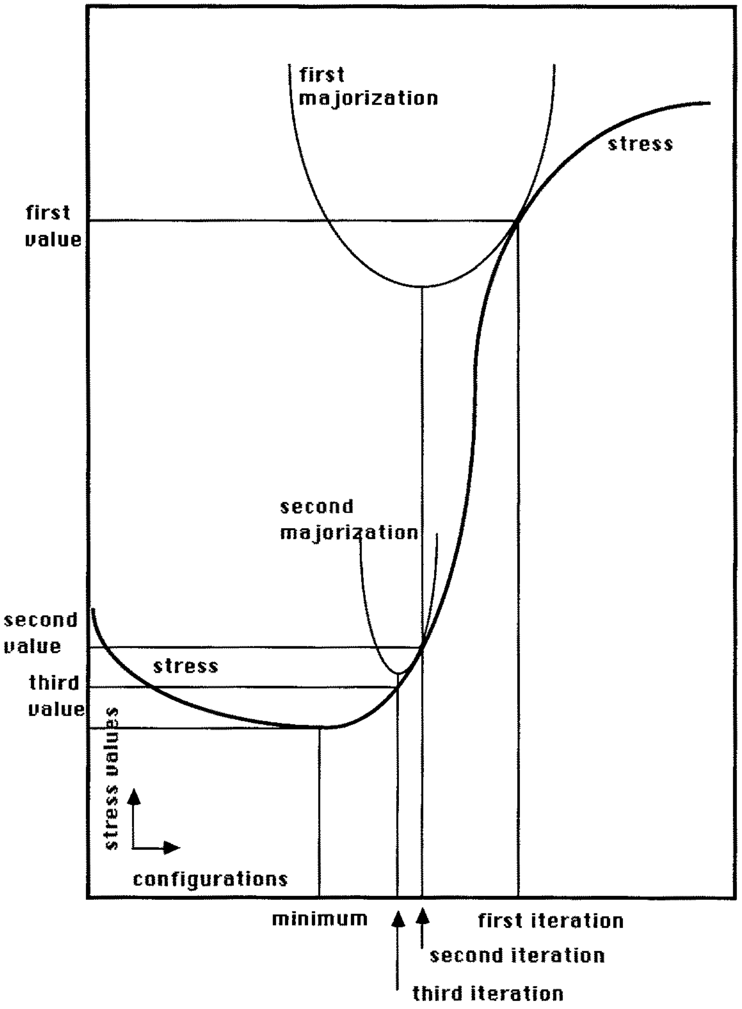

Figure 1. Three iterations of the majorization algorithm. We have drawn a section of the stress loss function, and two quadratic majorization functions. These touch the function at the current configuration, they are always above it, and their minimum provides the next configuration.

provided that $Y \neq \Gamma(Y)$. These results are illustrated in Figure 1. Thus either $Y = \Gamma(Y)$, in which case $\nabla \sigma(Y) = 0$ and we have found a solution of the stationary equations, or $Y \neq \Gamma(Y)$ and we can decrease the loss by replacing Y by its Guttman transform. This procedure constitutes the basic algorithm. It also explains why we call it a *majorization method:* instead of local linear approximation we use global quadratic majorization in each iteration step.

4. Sequences Generated by the Algorithm

The algorithm (13) generates a sequence $\mathbf{X}_k$, and also sequences of real numbers $\sigma_k = \sigma(\mathbf{X}_k)$, $\rho_k = \rho(\mathbf{X}_k)$, $\eta_k^2 = \eta^2(\mathbf{X}_k)$. Also define the sequences $\lambda_k = \rho(\mathbf{X}_k) / \eta(\mathbf{X}_k)$, and $\varepsilon_k^2 = \eta^2(\mathbf{X}_k - \Gamma(\mathbf{X}_k))$. The symbols $\uparrow$ and $\downarrow$ are used for convergence of monotone sequences which are, respectively, increasing and decreasing. The theorems in this section unify earlier results in de Leeuw (1977) and de Leeuw and Heiser (1980).

Theorem 1: (a) $\rho_k \uparrow \rho_\infty$,
(b) $\eta_k^2 \uparrow \eta_\infty^2 = \rho_\infty$,
(c) $\lambda_k \uparrow \lambda_\infty = \eta_\infty$,
(d) $\sigma_k \downarrow \sigma_\infty = 1 - \rho_\infty$,
(e) $\varepsilon_k^2 \rightarrow 0$.

Proof: From (3) and (4) we find, by using Cauchy-Schwartz, that $\rho(\mathbf{X}) \leq \eta(\mathbf{X})$ for all $\mathbf{X}$. Thus $\lambda_k \leq 1$ for all k. We can write (9) as $\rho(\mathbf{X}) = \text{tr } \mathbf{X}'\mathbf{V}\Gamma(\mathbf{X})$, and again by Cauchy-Schwartz, $\rho(\mathbf{X}) \leq \eta(\mathbf{X})\eta(\Gamma(\mathbf{X}))$ for all $\mathbf{X}$. Thus $\rho_k \leq \eta_k\eta_{k+1}$, and also $\lambda_k \leq \eta_{k+1}$. Now write (17) as $\rho(\Gamma(\mathbf{X})) \geq \text{tr } (\Gamma(\mathbf{X}))'\mathbf{B}(\mathbf{X})\mathbf{X} = \eta^2(\Gamma(\mathbf{X}))$, which implies that $\rho_k \geq \eta_k^2$ and $\lambda_k \geq \eta_k$. Summarizing the results so far gives

$$\eta_k \leq \lambda_k \leq \eta_{k+1} \leq \lambda_{k+1} \leq 1 \ , \tag{23a}$$

$$\eta_k^2 \leq \rho_k \leq \eta_k\eta_{k+1} \leq \eta_{k+1}^2 \leq \rho_{k+1} \leq 1 \ . \tag{23b}$$

These chains are sufficient to prove parts (a) (b) (c). We have already proved the decrease of σ_k in (22), the limit value follows trivially from $\sigma_k = 1 - 2\rho_k + \eta_k^2$, together with $\rho_\infty = \eta_\infty^2$. This proves (d). For (e) we write $\varepsilon_k^2 = \eta_k^2 + \eta_{k+1}^2 - 2\rho_k$, and again use $\rho_\infty = \eta_\infty^2$.
Q.E.D.

Observe that the theorem does *not* state that ε_k decreases *monotonically* to zero. In fact usually convergence of ε_k to zero is nonmonotonic. The theorem also does not say anything about the convergence of $\mathbf{X}_k$. The sequence $\mathbf{X}_k$ is studied in a separate theorem. Remember that $\mathbf{X}_\infty$ is an *accumulation point* of a sequence $\mathbf{X}_k$ if each neighborhood of $\mathbf{X}_\infty$ contains infinitely many points of the sequence.

Theorem 2: *Suppose S_∞ is the set of all accumulation points of the sequence $\mathbf{X}_k$. Then:*

(a) *S_∞ is nonempty,*

(*b*) *if* $\mathbf{X}_\infty \in S_\infty$ *then* $\sigma(\mathbf{X}_\infty) = \sigma_\infty$,

(*c*) *if* $\mathbf{X}_\infty \in S_\infty$ *and* $\mathbf{X}_\infty$ *is usable, then* $\mathbf{X}_\infty$ *is stationary (i.e., equal to its Guttman transform),*

(*d*) *if* S_∞ *is not a singleton, then it is a continuum.*

Proof: All $\mathbf{X}_k$ are column centered. In the $p(n-1)$ dimensional space of all column-centered $n \times p$ matrices, the function η defines a norm. Because $\eta_k \leq 1$ for all k, it follows that all $\mathbf{X}_k$ are in the unit ball of this normed space, and consequently they have at least one accumulation point. This implication proves (a).

Suppose $\mathbf{X}_\lambda$ is a subsequence converging to $\mathbf{X}_\infty$. Then, by continuity of σ, the sequence $\sigma(\mathbf{X}_\lambda)$ converges to $\sigma(\mathbf{X}_\infty)$. But the only accumulation point of $\sigma(\mathbf{X}_k)$ is σ_∞. This argument proves (b).

In the neighborhood of a usable configuration the Guttman transform is continuous. Thus, by the same argument that proved (b), we find that $\varepsilon(\mathbf{X}_\lambda) = \eta(\mathbf{X}_\lambda - \Gamma(\mathbf{X}_\lambda)) = \eta(\mathbf{X}_\lambda - \mathbf{X}_{\lambda+1})$ converges to $\eta(\mathbf{X}_\infty - \Gamma(\mathbf{X}_\infty))$, which must be zero by Theorem 1, part (e). This reasoning proves (c).

For result (d), we remember that a continuum is a closed set, which cannot be written as the union of two or more disjoint closed sets. A proof of (d) is given by Ostrowski (1966, Theorem 28.1).
Q.E.D.

Again Theorem 2 does not say that $\mathbf{X}_k$ converges. This fact is quite irrelevant from a practical point of view, however. If we define μ-optimal configurations as those configurations for which $\eta(\mathbf{X} - \Gamma(\mathbf{X})) < \mu$, then for all $\mu > 0$, the algorithm finds a μ-optimal configuration in a finite number of steps. This result is all the convergence we ever need in practice. In Theorem 2 there is a restriction in part (c), because we require that $\mathbf{X}_\infty$ is usable. This restriction can be removed by using subdifferentials (de Leeuw and Heiser 1980). But again, from a practical point of view, the restriction is not very important, because all local minima are usable (de Leeuw 1984).

The conclusion for Theorems 1 and 2 is that the sequences of loss and fit values generated by the algorithm converge monotonically. The difference between successive solutions converges to zero, which implies that the sequence of solutions converges. In a precise mathematical sense we have either convergence to a single point, or convergence to a continuum of stationary points, with all these stationary points having the same value of the loss function. We have not been able to exclude this last possibility, and indeed a natural candidate for such a continuum is available in any multidimensional scaling problem. If $\mathbf{X}_\infty$ is a stationary point, then for any $p \times p$ rotation matrix $\mathbf{K}$, the matrix $\mathbf{X}_\infty\mathbf{K}$ is also a stationary point, with the same loss function value. Thus there is the possibility that $\mathbf{X}_k$ converges to a continuum of the form $S_\infty = \{X \mid X = X_\infty K\}$, in the sense that *min* $\{\eta(X_k - X) \mid X \in S_\infty\}$ converges to zero, but X_k does not converge to a point in S_∞. Again it is clear that this fact is irrelevant from a practical point of view.

5. Derivatives of the Guttman Transform

A more detailed study into the convergence behavior of the algorithm is possible if we determine the derivative of the Guttman transform. This information makes it possible to investigate, for the first time, the rate of convergence of our basic algorithm. For a general discussion of the role of the derivative in one-step iterative processes, such as our (12), we refer to Ostrowski (1966, Chapter 22) or Ortega and Rheinboldt (1970, Chapter 10). For ease of reference we briefly summarize their key result here. If $X_{k+1} = \Phi(X_k)$ is any convergent iterative algorithm generating a sequence converging to X_∞, and if the largest eigenvalue τ_∞ of the derivative of Φ at X_∞ satisfies $0 < \tau_\infty < 1$, then $\|X_{k+1} - X_\infty\| / \|X_k - X_\infty\| \rightarrow \tau_\infty$, i.e., we have *linear convergence* with *rate* τ_∞.

The derivative is considered as a linear operator mapping the $(n-1)p$ dimensional space of column-centered configurations into itself. We can represent it computationally by an $np \times np$ matrix, but here we prefer to give it as a linear map which associates with each centered configuration Y another centered configuration $\Gamma_X(Y)$. The map Γ_X is the derivative of the Guttman transform at the usable configuration X. Thus we have, for example,

$$\Gamma(X + Y) = \Gamma(X) + \Gamma_X(Y) + o(\eta(Y)) \ , \tag{24}$$

showing the local linear approximation provided by the derivative. Observe that in (24), we have chosen η as the norm we are using in defining the derivative. This choice is not essential, of course, but it certainly is convenient.

We now give the formula for the derivative. It is most usefully written as

$$\Gamma_X(Y) = V^+ \{B(X)Y - U(X,Y)X\} \ , \tag{25}$$

with

$$U(X,Y) = 1/2 \, \Sigma_i \, \Sigma_j \, \{w_{ij}\delta_{ij}c_{ij}(X,Y) \, / \, d_{ij}^3(X)\} \ A_{ij} \ , \tag{26}$$

and $c_{ij}(X,Y) = \text{tr } X'A_{ij}Y$. Thus $U(X,X) = B(X)$.

Next we are interested in the eigenvalues and eigenvectors of Γ_X. Observe that we have interpreted it as an operator on the $(n-1)p$ dimensional space of centered configurations. It consequently has $(n-1)p$ eigenvalues, not necessarily all distinct. If we consider Γ_X as an operator on the space of all np matrices, then it has p additional eigenvalues equal to zero. The corresponding eigensubspace contains the solutions of $\eta(Y) = 0$.

Before proceeding, we have to single out one special case. If $p = 1$, then $U(X,Y)X = B(X)Y$, and thus $\Gamma_X(Y) = 0$. It is shown in de Leeuw and Heiser (1977) that the Guttman transform iterations converge in a finite number of steps if $p = 1$. This special case was already singled out by Guttman (1968). Defays (1978), Heiser (1981), and Hubert and Arabie (1986) show that one-dimensional scaling is essentially a combinatorial problem. Because either the result $\Gamma_X(Y) = 0$ or the fact of finite convergence stops all considerations having to do with rate of convergence, we assume from now on that $p > 1$.

Result 1. X is an eigenvector of Γ_X with eigenvalue zero. This result follows directly from $U(X,X) = B(X)$ and (24).

Result 2. Γ_X has simple structure, i.e., $(n-1)p$ linearly independent eigenvectors. This follows because $\Gamma_X(Y) = \lambda Y$ can be written in the form $(\nabla^2\rho(X))Y = \lambda VY$, where $\nabla^2\rho$ is the operator corresponding with the second partial derivatives of ρ, which is consequently symmetric. Thus the eigenvalues are real, and the eigenvectors Y_s can be chosen such that they are orthogonal, i.e., such that $\text{tr } Y_s'VY_t = \delta^{st}$, with δ^{st} the Kronecker delta.

Result 3. The eigenvalues of Γ_X are non-negative. This fact follows from the representation in the previous result. Because ρ is convex, the operator $\nabla^2\rho$ is positive semidefinite.

Result 4. If X is a local minimum of σ, then all eigenvalues of Γ_X are less than or equal to one. This result follows because if X is a local minimum, then we must have that $\nabla^2\sigma(X)$ is positive semidefinite. And $\nabla^2\sigma(X)$ is positive semidefinite if and only if $I - \Gamma_X$ is positive semidefinite, i.e., if and only if all eigenvalues of Γ_X are less than or equal to one.

Result 5. If X is stationary, then Γ_X has $1/2p(p-1)$ eigenvalues equal to one. To prove this result, choose $Y = XS$, with S anti-symmetric, i.e., $S = -S'$. Then $c_{ij}(X,Y) = \text{tr } X'A_{ij}XS = 0$, and thus from (24) and (25) we have $\Gamma_X(Y) = V^+B(X)XS = XS = Y$. The anti-symmetric matrix S can be chosen in $1/2p(p-1)$ linearly independent ways.

6. A Small Example

The example we use has $n = 4$ objects, and has all dissimilarities δ_{ij}, with $i \neq j$, equal to $1/\sqrt{6}$. The weights w_{ij} are all equal to one, and consequently normalization (5) is true. Of course this example does not constitute a realistic multidimensional scaling problem, because it is much too small. We use the example to illustrate what kind of stationary points we can expect in a metric scaling problem, and how our simple algorithm will behave in a neighborhood of these stationary points. We do not intend to prove, in any sense, that our algorithm is *better* than other existing algorithms. In fact we think that the other algorithms will behave quite similarly. It seems to us that the example can be used to illustrate the great difficulties multidimensional scaling methods can encounter if the iterations are started at unfortunate places, and to illustrate the slow rate of convergence that seems to be typical for multidimensional scaling problems. Both the local minimum problem and the slow convergence problem will tend to become more serious if we increase the size of the problem.

We first make a small list of stationary points. This list may not be exhaustive, but it contains some of the more interesting types of stationary points.

Stationary point 1. Take four points equally spaced on a line. This actually defines a whole family of stationary points, because any permutation of four equally spaced points will do. Moreover we can think of this solution as embedded in one-dimensional space, in two-dimensional space, and so on. If $p = 1$ we know that $\Gamma_X = 0$, we also know that the equally spaced points define the global minimum (de Leeuw and Stoop 1983). But, as we said before, we are really only interested in $p > 1$. If $p = 2$, i.e., if we have four points equally spaced on a line in two-space, then the eigenvalues of Γ_X are four zeros together with the eigenvalues of $1/4B(X)$, which are 0, 1, 1.5, and 1.8333. Thus for $p = 2$ the four points on a line are not a local minimum, in fact they define a saddle-point. The function value in this saddle-point is .1666666667.

Stationary point 2. Take three points in the corners of an equilateral triangle, and the fourth one in the centroid of the triangle. This is, again, a whole family of stationary points, all with loss function value $\sigma = .06698729811$. For $p = 2$ the operator Γ_X has 8 eigenvalues. Three are equal to zero, three are equal to one, and two are equal to .2321. Thus the triangle plus centroid defines a local minimum, but not an isolated one. There are three eigenvalues equal to one and only one of them corresponds with the trivial eigenvalue equal to one of result (5) in the previous section. If $p = 3$ we have the same 8 eigenvalues, plus the 4 eigenvalues of $1/2B(X)$, which are 0, 1, 1, and 1.4641 in this case. Thus for $p = 3$ this stationary point is not a local minimum any more.

Stationary point 3. Take four points in the corners of a square. Loss is .02859547921. For $p = 2$ the eigenvalues of Γ_X are three zeros, .4142, three times .5858, and one trivial unit eigenvalue. Thus we have local minimum here that is isolated in the sense that the manifold of all rotations of the square is isolated from other stationary values. Again the square is not a local minimum for $p = 3$, although it remains stationary.

Stationary point 4. Take four points in the corners of a regular tetrahedron. The loss function is now zero, which shows directly that this three-dimensional solution is certainly the global minimum. The eigenvalues of Γ_X are zero four times, 1/2 three times, 3/4 two times, and one three times. The unit eigenvalues correspond with the $1/2p(p-1)$ trivial eigenvalues defining the manifold of rotations.

7. Rate of Convergence

The discussion in the previous sections shows that at least part of the difficulty with proving actual convergence of our iterations comes from the rotational indeterminancy of multidimensional scaling. Because of this rotational indeterminancy, Γ_X has at least $1/2p(p-1)$ unit eigenvalues at a stationary point X. If we eliminate rotational indeterminancy, then we eliminate these difficulties. First, we call a regular stationary point *isolated* if Γ_X has exactly $1/2p(p-1)$ unit eigenvalues. We call it an *isolated local minimum* if all other eigenvalues are strictly less than one. In the small example in the previous section, the four points in the corner of a square are an isolated local minimum for $p = 2$. For $p = 3$ the solution is neither isolated nor a local minimum. The equilateral triangle with centroid is a local minimum for $p = 2$, but not an isolated local minimum. The regular tetrahedron is an isolated local minimum for $p = 3$. At an isolated local minimum, we use the symbol κ for the largest eigenvalue less than one. We call it the *level* of the isolated local minimum.

Theorem 3: *If X_k has an accumulation point, which is an isolated local minimum, and has level κ, then $\varepsilon_{k+1} / \varepsilon_k \rightarrow \kappa$.*

Proof: We want to apply the general theorems of Ostrowski and of Ortega and Rheinboldt referred to above. First, we eliminate rotational indeterminancy by defining a new sequence X_k^o. For each k the configuration X_k^o is a rotation of X_k; moreover it is a specific rotation which identifies the configuration uniquely in the manifold of rotations. We can rotate to principal components, for example, with some special provision for equal eigenvalues. Because X_k^o is a rotation of X_k, the sequences $\sigma_k, \rho_k, \eta_k, \lambda_k$ generated by this modified algorithm are exactly the same. So is $\varepsilon_k^2 = (\eta_{k+1})^2 + \eta_k^2 - 2\rho_k$, although now $\varepsilon_k \neq \eta((X_{k+1})^o - X_k^o)$. The transformation which maps X_k^o to $(X_{k+1})^o$ has a derivative Γ_X^o at a stationary point with exactly the same eigenvalues as Γ_X, except for the $1/2p(p-1)$ unit eigenvalues, which are replaced by zeroes. Thus $\kappa < 1$ is actually the largest eigenvalue of Γ_X^o, so that X_k^o converges linearly, with rate κ.
Q.E.D.

If the stationary point is a non-isolated local minimum, or not even a local minimum, then Theorem 3 does not say anything about the rate of convergence. This fact does not seem to be a very important restriction of generality in practice, because it seems difficult to get our algorithm to converge to a non-isolated local optimum. We illustrate this finding with the small example from the previous section.

The equilateral triangle with centroid is a non-isolated local minimum for $p = 2$. Start the iterations from a small perturbation of this stationary point. With a very close start ($\sigma = .0669873151$), we have convergence to the stationary value with 10-decimal precision within 5 iterations. The ratio $(\varepsilon_{k+1})^2 / \varepsilon_k^2$ continues to increase, however, although extremely slowly. It is .99 at iteration 8, and .999 at iteration 10. We have stopped the process at iteration 30, at which point we are still equally close to the stationary value, and the ratio is still increasing.

We have restarted the iterations somewhat further away from the stationary point ($\sigma = .0669895385$). After 10 iterations σ is down to .0669877606 and ε^2 is 7×10^{-9}. The ratio $(\varepsilon_{k+1})^2 / \varepsilon_k^2$ is .9739919946. At iteration 50 we have $\sigma = .0669873813$, $\varepsilon^2 = 36 \times 10^{-10}$, and $(\varepsilon_{k+1})^2 / \varepsilon_k^2 = .9926593514$. Around iteration 60 the value of σ drops below .066987298 (equilateral triangle with centroid) and ε^2 begins to rise, causing a ratio $\varepsilon_{k+1} / \varepsilon_k$ larger than one. This situation continues for a very long time. At iteration 200, for instance, we have $\sigma = .0669757275$ and $\varepsilon^2 = 3413 \times 10^{-10}$. The ratio of successive epsilons is still larger than one. This continues until iteration 250. In the meantime ε^2 has increased to .0026904972, and σ, which started dropping rapidly at iteration 225, is down to .0368738809. Convergence now becomes rapid, and within 20 iterations the configuration converges to the four corners of the square, which is an isolated local minimum (in fact the global minimum) for $p = 2$. At iteration 270 we have $\sigma = .0285954792$, $\varepsilon^2 < 10^{-10}$, and $(\varepsilon_{k+1})^2 / \varepsilon_k^2 = .3431684733$, which is for all practical purposes equal to κ^2. Thus we have started close to a non-isolated local minimum. The algorithm has great difficulty in getting away from it, but ultimately succeeds.

With a restart even further away ($\sigma = .0675512622$) the algorithm has difficulty escaping only until iteration 30. The ε^2 decreases rapidly again, and we have convergence to the square in 55 iterations. It is now not difficult to conjecture that in the first start, in which we seemed to converge on the equilateral triangle, we merely did not continue long enough. After hundreds or perhaps thousands of iterations we would converge on the square again, if we continued.

8. Nonmetric Scaling

We now introduce some additional terminology in order to define *nonmetric scaling*. The loss function for a nonmetric scaling problem can be written as

$$\sigma(\mathbf{X},\Delta) = 1/2\, \Sigma_i\, \Sigma_j\, w_{ij}\, (\delta_{ij} - d_{ij}(\mathbf{X}))^2 \ , \qquad (27)$$

where all symbols have the same meaning as before, except for $\Delta = \{\delta_{ij}\}$, which now contains the *disparities* and no longer the dissimilarities. In nonmetric scaling the loss function (27) must be minimized over both the configuration X and the disparities Δ, where the disparities are restricted by the ordinal information in the data. Thus the disparities are additional parameters in the nonmetric scaling problem over which we minimize, in contrast to the metric scaling problem in which the dissimilarities are fixed numbers. Thus we can choose Δ freely, provided it is *feasible*, which means in this case *monotone* with the given dissimilarities. In this sense nonmetric scaling generalizes metric scaling, in which Δ must not only be monotone but in fact *identical* to the dissimilarities. In order to prevent trivial solutions, we must also impose a normalization condition on the disparities, for instance that their sum of squares is unity, or that their variance is unity.

As indicated in de Leeuw (1977) we can think of nonmetric scaling algorithms in two different ways. First, we interpret them as alternating least squares methods, which alternate one gradient step (or Guttman-transform) with a monotone regression step. In the gradient step, the loss function (27) is minimized (or rather decreased) by choosing a new X; in the monotone regression step (27) is decreased by choosing a new Δ, consistent with the ordinal constraints. Of course it is possible, and perhaps sometimes advisable, to introduce some obvious modifications of this algorithm. Instead of alternating one Guttman transform with one monotone regression we can perform more Guttman transforms between monotone regressions. These Guttman steps have a rate of convergence which is described by our results above. In the alternating least squares interpretation, the loss function (27) is clearly interpreted as a function of two sets of parameters, the coordinates and the disparities.

Second, it is also possible to view the loss function in nonmetric scaling as a function of X alone. This is the original definition of stress as proposed by Kruskal (1964a, 1964b). If σ(X,Δ) is the "stress" used in the first approach, then Kruskal's stress is the minimum of σ(X,Δ) over all feasible disparities Δ. Thus Δ is "projected out," and the remaining function depends only on X. This elementary fact has caused a great deal of confusion in the early days of multidimensional scaling. The confusion was made even bigger by the fact that the derivative of $\sigma(\mathbf{X})$ is the same as the partial derivatives of $\sigma(\mathbf{X},\Delta)$ with respect to $\mathbf{X}$, evaluated at the optimal $\Delta(\mathbf{X})$. Thus $\sigma(\mathbf{X}) = \sigma(\mathbf{X},\Delta(\mathbf{X}))$, but also $\nabla\,\sigma(\mathbf{X}) = \nabla_x\sigma(\mathbf{X},\Delta(\mathbf{X}))$. This last result is due to Kruskal (1971), who used it to show that $\sigma(\mathbf{X})$ is differentiable (whenever $\sigma(\mathbf{X},\Delta)$ is differentiable). It is also used by de Leeuw (1977) to show that the iteration $\mathbf{X}_{k+1} = \mathbf{V}^+\mathbf{B}(\mathbf{X}_k)\mathbf{X}_k$ is still a convergent algorithm if we define $\mathbf{B}(\mathbf{X})$ as in (10) but with $\Delta(\mathbf{X})$ substituted for Δ. Thus, our qualitative convergence results remain true without modification, both in the alternating least squares and in the (sub)differential interpretation.

Unfortunately the transformation $\mathbf{X} \rightarrow \Delta(\mathbf{X})$ is generally not differentiable. In fact, to find $\Delta(\mathbf{X})$ we have to project $\mathbf{D}(\mathbf{X})$ orthogonally on a polyhedral convex cone, which implies that the transformation is piecewise linear in the distances. The linear pieces are joined in a continuous but non-smooth way (compare Kruskal 1971). If we have convergence of the non-metric scaling algorithm to a point where the cone-projection is locally a constant linear map, then our convergence results apply. In monotone regression terms this means that for all points in an open neighborhood of the solution we are converging to, monotone regression finds the same partitioning into blocks in its computation of the disparities, i.e., for all these points it projects on the same face of the cone. In general, however, we cannot exclude the possibility that the convergence is to a point in the boundary of two regions with different projection maps. These linear maps again correspond to the partitioning into blocks found by the monotone regression algorithm. In this case our results do not apply, and they must be adapted.

9. Summary and Conclusions

We have shown in this paper that our basic majorization algorithm for multidimensional scaling converges to a stationary point, if convergence is defined using the *asymptotic regularity* of the generated sequence, i.e., in terms of the fact that the distance between two consecutive members of the sequence converges to zero. We have also shown that if one of the accumulation points of the sequence is an isolated local minimum, then convergence is linear. This condition seems to be all that is needed for practical applications. It follows from our small numerical example that it is possible for actual computer programs to stop at saddle points which are not local minima, and that in the neighborhood of such saddle points convergence may look sublinear. Our experience with many practical examples indicates that the level of isolated local minima (i.e., the convergence rate of the algorithm we have described) in multidimensional scaling is very often close to unity. Thus although convergence is theoretically linear, it can be extremely slow.

Consequently it becomes very important, at least in some cases, to look for ways to speed up linear convergence, or even for ways to attain supralinear convergence. These acceleration devices will be investigated in subsequent publications. Simple ways to speed up linear convergence were already investigated by de Leeuw and Heiser (1980); more complicated ones were studied by Stoop and de Leeuw (1983). In both papers the basic convergence of the Guttman-transform iterations is preserved, but a specially developed step-size procedure is added to the algorithm. Both on theoretical and on empirical grounds one can argue that stepsize procedures in MDS, including the ones devised by Kruskal (1964), will generally double the speed of convergence, i.e., half the number of iterations required for a given precision. In the case of very slow linear convergence, this does not really help very much. Of course the formulae derived in Section 5 can be used quite easily to derive the exact form of Newton's method applicable to multidimensional scaling, but our numerical experience so far suggests that Newton's method must also be used with much care in this context.

Our main conclusion is that the majorization method is reliable and very simple, but that it is generally slow, and sometimes intolerably slow. It seems to us that an additional conclusion is that one should always study the second derivatives of the loss function at the stopping point of the algorithm. This information indicates if we have stopped at a local minimum, and how much improvement we can expect in various directions. Such information on improvement can be used either to try to make one or more Newton-steps, or to derive information on the stability of the solution.

References

BORG, I. (1981), *Anwendungsorientierte Multidimensionale Skalierung*, Berlin, West Germany: Springer.

DEFAYS, D. (1978), "A Short Note on a Method of Seriation," *British Journal of Mathematical and Statistical Psychology, 31*, 49-53.

DE LEEUW, J. (1977), "Applications of Convex Analysis to Multidimensional Scaling," in *Recent Developments in Statistics*, eds. J.R. Barra, F. Brodeau, G. Romier and B. van Cutsem, Amsterdam: North Holland, 133-145.

DE LEEUW, J. (1984), "Differentiability of Kruskal's Stress at a Local Minimum." *Psychometrika, 49*, 111-113.

DE LEEUW, J., and HEISER, W. J. (1977), "Convergence of Correction Matrix Algorithm for Multidimensional Scaling," in *Geometric Representations of Relational Data*, ed. J. C. Lingoes, Ann Arbor: Mathesis Press, 735-752.

DE LEEUW, J., and HEISER, W. J. (1980), "Multidimensional Scaling with Restrictions on the Configuration," in *Multivariate Analysis, Vol. V*, ed. P. R. Krishnaiah, Amsterdam: North Holland.

DE LEEUW, J., and STOOP, I. (1984), "Upper Bounds for Kruskal's Stress," *Psychometrika, 49,* 391-402.

GUTTMAN, L. (1968), "A General Nonmetric Technique for Finding the Smallest Coordinate Space for a Configuration of Points," *Psychometrika, 33,* 469-506.

HARTMANN, W. (1979), *Geometrische Modelle zur Analyse empirischer Daten,* Berlin: Akademie Verlag.

HEISER, W. J. (1981), *Unfolding Analysis of Proximity Data,* Unpublished Doctoral Dissertation, University of Leiden.

HUBERT, L., and ARABIE, P. (1986), "Unidimensional Scaling and Combinatorial Optimization," in *Multidimensional Data Analysis,* eds. J. de Leeuw, et al, Leiden: DSWO-Press, 181-196.

KRUSKAL, J. B. (1964a), "Multidimensional Scaling by Optimizing Goodness of Fit to a Nonmetric Hypotheses," *Psychometrika, 29,* 1-28.

KRUSKAL, J. B. (1964b), "Nonmetric Multidimensional Scaling: A Numerical Method," *Psychometrika, 29,* 115-129.

KRUSKAL, J. B. (1971), "Monotone Regression: Continuity and Differentiability Properties," *Psychometrika, 36,* 57-62.

KRUSKAL, J. B., and WISH, M. (1978), *Multidimensional Scaling,* Newbury Park, CA: Sage.

LINGOES, J. C., and ROSKAM, E. E. (1973), "A Mathematical and Empirical Comparison of Two Multidimensional Scaling Algorithms," *Psychometrika, 38,* Monograph Supplement.

ORTEGA, J. M., and RHEINBOLDT, W. C. (1970), *Iterative Solution of Nonlinear Equations in Several Variables,* New York: Academic Press.

OSTROWSKI, A. M. (1966), *Solution of Equations and Systems of Equations,* New York: Academic Press.

STOOP, I., and DE LEEUW, J. (1983), *The Stepsize in Multidimensional Scaling Algorithms, Paper presented at the Third European Meeting of the Psychometric Society, Jouy-en-Josas, France, July 5-8, 1983.*