76

CONVERGENCE OF CORRECTION-MATRIX ALGORITHMS FOR MULTIDIMENSIONAL SCALING

JAN DE LEEUW
DEPARTMENT OF DATA THEORY

WILLEM HEISER
DEPARTMENT OF PSYCHOLOGY

UNIVERSITY OF LEIDEN
THE NETHERLANDS

We recently developed a convergence theory for multidimensional scaling algorithmsclosely related to the Guttman-Lingoes-Roskam C-matrix method. The theory uses ideas from convex analysis if applied to general Minkowski metrics, but in the Euclidean case it can be derived completely from the Cauchy-Schwartz inequality.

This paper consists of three parts. In the first part we repeat some of the knownresults for the Euclidean case in a slightly modified and extended form. In the second part we extend the approach to partitioned data sets and loss functions that are normalized differently. And in the third part we extend the theory to Euclidean individual differences scaling.

Proofs of the theorems in the first part of the paper have been published elsewhere (De Leeuw, 1977a). The second part of the paper is based on De Leeuw (1977b) and will be elaborated on in Young, De Leeuw, & Takane (1977). The third part is closely related to Heiser (1975) and will be published in more detail elsewhere (De Leeuw & Heiser, 1977).

1:1

In metric multidimensional scaling we minimize loss functions of the form:

$$(1) \qquad S_0(X) = \Sigma_k^m \Sigma_i^n \Sigma_j^n w_{ijk}(\delta_{ijk} - d_{ij}(X))^2.$$

Here X is the n x p *configuration matrix*, assumed to be centered, the δ_{ijk} are given *dissimilarities*, the w_{ijk} are given *weights*, and the $d_{ij}(X)$ are Euclidean *distances*, defined on the rows of X by:

$$(2) \qquad d_{ij}^2(X) = (x_i - x_j)'(x_i - x_j).$$

In De Leeuw [1977] a simple convergent algorithm for minimizing $S_0(X)$ was proposed, which is closely related to Guttman's C-matrix method [1968]. We review this algorithm briefly, without going into all the technical details.

1:2

As a first step in the derivation of the algorithm we use the homogeneity of the distance function $d_{ij}(X)$. Because $d_{ij}(\beta X) = \beta d_{ij}(X)$ for all $\beta \geq 0$, we can minimize $S_0(X)$ by minimizing:

$$(3) \qquad \Sigma_k \Sigma_i \Sigma_j w_{ijk}(\delta_{ijk} - \beta d_{ij}(X))^2$$

over all non-negative β *and* over all *normalized* X. We call a centered configuration matrix X normalized if $\eta(X) = 1$, where

$$(4) \qquad \eta^2(X) = \Sigma_k \Sigma_i \Sigma_j w_{ijk} d^2_{ij}(X).$$

For convenience we also assume that the dissimilarities Δ are normalized in the sense that $\eta(\Delta) = 1$, where

$$(5) \qquad \eta^2(\Delta) = \Sigma_k \Sigma_i \Sigma_j w_{ijk} \delta^2_{ijk}.$$

The minimum of (3) over $\beta \geq 0$ for a fixed normalized X is $1 - \rho^2(X)$, with:

$$(6) \qquad \rho(X) = \Sigma_k \Sigma_i \Sigma_j w_{ijk} \delta_{ijk} d_{ij}(X).$$

The minimum is attained for $\beta = \rho(X)$. Thus, we can minimize $S_0(X)$ by maximizing $\rho(X)$ over all normalized configurations. If **X** solves this maximization problem, then $\rho(\mathbf{X})\cdot\mathbf{X}$ minimizes $S_0(X)$.

1:3

The second step in the derivation of the algorithm uses matrix notation. From (4) it follows that there is a matrix V such that

$$(7) \qquad \eta^2(X) = trX'VX,$$

and from (6) it follows that there is a matrix B(X), depending on X, such that

(8) $$\rho(X) = trX'B(X)X.$$

More precisely, V has off-diagonal elements

(9a) $$v_{ij} = -\Sigma_k(w_{ijk} + w_{jik}),$$

and B(X) has off-diagonal elements

(9b) $$b_{ij}(X) = -\Sigma_k(w_{ijk}\delta_{ijk} + w_{jik}\delta_{jik})/d_{ij}(X).$$

The diagonal elements of both V and B(X) are then defined in such a way that their rows and columns sum to zero. B(X) is closely related to the matrix C (X) discussed by Guttman [1968], *cf.*, also Lingoes & Roskam [1973]. Some properties of B(X) are discussed by Guttman [1968] and by De Leeuw [1977]. Observe that we have not derived B(X) by differentiating $S_0(X)$, but simply as a notational device that makes it possible to write $\rho(X)$ *as if* it was a quadratic form on the space of configuration matrices. In this notation the fundamental inequality on which our form of scaling is based can be written simply as

(10) $$\rho(X) \geq trX'B(Y)Y.$$

The above inequality is true for all pairs of configuration matrices X,Y. It is derived in De Leeuw [1977] by a straight forward application of the Cauchy-Schwartz inequality to the definition (6) of $\rho(X)$. The inequality shows that $\rho(X)$ majorizes a family of linear functions, and the algorithm derives from this fact. Consequently, the algorithm is not based on local linear approximation, but on global linear minorization. In the terminology of convex analysis it is not a gradient but a subgradient method.

1:4

We shall now describe the actual algorithm. It turns out to be identical to Guttman's C-matrix method, which was derived by differentiation and by setting the partials equal to zero. We use a subscript μ for iterations. Suppose X_μ is our current best normalized solution. We first compute (using the Moore-Penrose inverse V^+ of V)

$$\text{(11a)} \qquad Y_\mu = V^+B(X_\mu)X_\mu,$$

and we then compute $X_{\mu+1}$ by normalizing Y_μ:

$$\text{(11b)} \qquad X_{\mu+1} = Y_\mu/\eta(Y_\mu).$$

In fact, it is clear from (11a) that it is not at all necessary to normalize; if we substitute (11b) into (11a) we get the more direct algorithm:

$$\text{(11c)} \qquad Y_{\mu+1} = V^+B(X_{\mu+1})X_{\mu+1} = V^+B(Y_\mu)Y_\mu.$$

From (11a) and (8) we obtain

$$\text{(12)} \qquad tr X'_\mu V Y_\mu = tr X'_\mu B(X_\mu)X_\mu = \rho(X_\mu),$$

which implies, by using Cauchy-Schwartz and (7),

$$\text{(13)} \qquad \rho(X_\mu) \leq \{tr X'_\mu V X_\mu \cdot tr Y'_\mu V Y_\mu\}^{1/2} = \eta(Y_\mu).$$

On the other-hand, from (11a), (11b), and (10),

$$\text{(14)} \qquad \eta(Y_\mu) = tr X'_{\mu+1} V Y_\mu = tr X'_{\mu+1} B(X_\mu)X_\mu \leq \rho(X_{\mu+1}).$$

Taken together (13) and (14) imply that both $\rho(X_\mu)$ and $\eta(Y_\mu)$ are bounded increasing sequences, converging to the same limit. Because both X_μ and $X_{\mu+1}$

are normalized

$$(15)\qquad \tfrac{1}{2}tr(X_\mu - X_{\mu+1})'V(X_\mu - X_{\mu+1}) = 1 - trX_\mu'VX_{\mu+1}$$
$$= 1 - trX_\mu'VY_\mu/\eta(Y_\mu) = 1 - \rho(X_\mu)/\eta(Y_\mu).$$

The above implies that $||X_\mu - X_{\mu+1}||$ converges to zero, which, in turn, implies, by a familiar theorem of Ostrowski, that either the sequence X_μ converges or that X_μ has a continuum of accumulation points, with each accumulation point having the same value of $\rho(X)$ and $\eta(X)$. In this specific sense the algorithm converges. In theory, there may be pathological cases in which it is not strictly true that we have convergence, but in those cases we clearly would be satisfied with any one of the accumulation points.

For completeness, we also prove that the simpler algorithm (11c) converges in the above same sense. We have $\rho(Y_\mu) = \rho(X_{\mu+1})\cdot\eta(Y_\mu)$. Thus, if $\rho(X_\mu)$ and $\eta(Y_\mu)$ converge to, say, ρ_∞, then $\rho(Y_\mu)$ converges to ρ^2_∞. Moreover, the identity

$$(16)\qquad tr(Y_\mu - Y_{\mu+1})'V(Y_\mu - Y_{\mu+1}) = \eta^2(Y_\mu) + \eta^2(Y_{\mu+1}) - 2\eta(Y_\mu)\cdot\rho(X_{\mu+1})$$

proves that $||Y_\mu - Y_{\mu+1}||$ also converges to zero. All accumulation points of both X_μ and Y_μ are stationary points of the algorithm, which implies that they also satisfy the stationary equations. For some technical problems connected with the possibility that the loss function is not differentiable at an accumulation point, which occurs if at least one of the $d_{ij}(X)$ vanishes, we refer to De Leeuw [1977].

1:5

We now explicate metric unidimensional scaling, a special case having both theoretical and practical interest, as Guttman [1968] has already pointed out. The basic algorithm becomes (in the simplest case where all weights are equal and there are no replications):

$$(17)\qquad x_i^+ = \Sigma_j \frac{\delta_{ij}}{d_{ij}}(x_i - x_j) = \Sigma_j \delta_{ij} sign(x_i - x_j).$$

The above means that x^+, the update of x, depends only on the rank order of the x_i. Because there are only a finite number of possible rank orders andbecause the algorithm cannot repeat any given rank order, this implies that we converge on a stationary point of (17) in a finite number of steps. This sounds nice, but, unfortunately, it only reflects the fact that the one-dimensional metric MDS problem is a combinatorial optimization problem. If we do not know (or can not assume) what the optimal order of the points on the dimension is then it becomes almost impossible to find the global optimum (if n is at all large). If, however, we do know the order (or can assume it for scaling purposes), the problem becomes very simple indeed. We have to minimize:

$$(18) \qquad \sigma(x) = \Sigma_i\Sigma_j[\delta_{ij} - s_{ij}(x_i - x_j)]^2$$

over all x that satisfy $s_{ij}(x_i - x_j) \geq 0$. The s_{ij} are known numbers equal to ±1. But minimizing (18) is easily seen to be equivalent to minimizing:

$$(19) \qquad \Sigma_i(x_i - t_i)^2,$$

with

$$(20) \qquad t_i = n^{-1}\Sigma_j s_{ij}\delta_{ij}$$

over all x satisfying $s_{ij}(x_i - x_j) \geq 0$. And, this last problem is a simple monotone regression problem. Thus: if the order of the points on the dimension is known (or fixed), then the metric one-dimensional MDS problem is a monotone regression problem; if, on the other hand, the order is not known or assumed, we then need a combinatorial search over the set of all possible orders, and the finite algorithm (17) does not help very much.

1:6

In De Leeuw [1977] we have also generalized the basic algorithm (11) in such a way that it can also solve nonmetric scaling problems. This generalization turns out to be surprisingly simple. Define the loss function:

$$(21) \qquad S_1(X,\Delta) = \Sigma_k\Sigma_i\Sigma_j w_{ijk}[\delta_{ijk} - d_{ij}(X)]^2/\Sigma_k\Sigma_i\Sigma_j w_{ijk} d_{ij}^2(X).$$

Now the loss is a function of both the configuration X and the dissimilarities

δ_{ijk}, which are also partially unknown. We do know that the admissible dissimilarity quantifications are in some convex cone Γ (usually the cone of monotone matrices) and that we have to choose the optimally scaled δ_{ijk} from this cone. In minimizing $S_1(X,\Delta)$ we first apply homogeneity again, as in §1:2. We have to minimize:

$$(22) \qquad \Sigma_k\Sigma_i\Sigma_j w_{ijk}[\alpha\delta_{ijk} - \beta d_{ij}(X)]^2/\Sigma_k\Sigma_i\Sigma_j w_{ijk}[\beta d_{ij}(X)]^2$$

over all $\alpha,\beta \geq 0$ and over all normalized configurations and dissimilarities. Normalized configurations are defined by $\eta(X) = 1$, as before, normalized dissimilarities satisfy $\eta(\Delta) = 1$, where

$$(23) \qquad \eta^2(\Delta) = \Sigma_k\Sigma_i\Sigma_j w_{ijk}\delta_{ijk}^2.$$

Moreover, we also define

$$(24) \qquad \rho(X,\Delta) = \Sigma_k\Sigma_i\Sigma_j w_{ijk}\delta_{ijk}d_{ij}(X).$$

The minimum of (22) over all α and β for fixed normalized X and Δ turns out to be $1 - \rho^2(X,\Delta)$. Thus, minimizing $S_1(X,\Delta)$ can be accomplished by maximizing $\rho(X,\Delta)$ over all normalized X and over all normalized admissible Δ.

Now, define $\tau(X)$ as the maximum of $\rho(X,\Delta)$ over the normalized admissible Δ, for fixed normalized X, and define $\hat{\Delta}(X)$ as the maximizer. Thus, $\tau(X) = \rho[X,\hat{\Delta}(X)]$ and maximizing $\rho(X,\Delta)$ is equivalent to maximizing $\tau(X)$ over all normalized configurations. In mathematical programming, the process of eliminating one set of variables by 'inner maximization' is called *projection*; in the MDS literature we could use Guttman's terminology [1968] and call $\tau(X)$ the goodness of fit measure for a *single phase* algorithm. By generalizing the matrix notation of §1:3 in an obvious way, we can write:

$$(25) \qquad \tau(X) = tr X'B(X,\hat{\Delta}(X))X,$$

and it turns out that even the inequality

(26) $$\tau(X) \geq tr X'B(Y,\hat{\Delta}(Y))Y$$

remains valid. As a consequence, the complete convergence proof of §1:4 also remains valid for the nonmetric algorithm:

(27a) $$Y_{\mu} = B(X_{\mu},\hat{\Delta}(X_{\mu}))X_{\mu},$$

(27b) $$X_{\mu+1} = Y_{\mu}/\eta(Y_{\mu}).$$

The only difference between the metric and the nonmetric case is in the computation of Y_{μ}. To find $\hat{\Delta}(X_{\mu})$ we have to solve a normalized cone regression problem [De Leeuw, 1977b], which usually transforms to a simple monotone regression problem (but in the additive constant case it is a simple linear regression problem). Observe that our use of projection forces us to employ monotone regression and not, for example, rank images. Only in this sense is our algorithm (27) equivalent to Guttman's single-phase C-matrix algorithm. It is, of course, perfectly legitimate to use the rank images $\Delta^*(X)$ in the earlier iterations (this may speed up the process, *cf.*, Lingoes & Roskam [1973]). As long as one switches to $\hat{\Delta}(X)$ in the final iterations convergence will be achieved.

2:1

It has been pointed out by Roskam [1968], Kruskal & Carroll [1969] that loss function (21) is not appropriate for partitioned data sets, *i.e.*, data sets in which the dissimilarities can be partitioned into subsets in such a way that all restrictions are within subsets and no restrictions are between subsets. A familiar special case is matrix partitioning, in which there are *m* cones: $\Gamma_1, \Gamma_2, \ldots, \Gamma_m$ and Δ_k is restricted to be in Γ_k. In these cases we would like to minimize the partitioned loss function:

(28) $$S_2(X,\Delta) = \Sigma_k\{\Sigma_i\Sigma_j w_{ijk}[\delta_{ijk} - d_{ij}(X)]^2/\Sigma_i\Sigma_j w_{ijk}d_{ij}^2(X)\}.$$

Unfortunately, however, it is impossible to apply the elegant analysis of §1:6 to this particular loss function. The main reason that we can not is that we can not define normalization as

$$\Sigma_i\Sigma_j w_{ijk} d_{ij}^2(X) = m^{-1} \tag{29}$$

for all k=1,2,...,m. The latter would be a requirement that is much too strong in the general case; in fact, the system (29) may not even have a solution. It does have a solution when $w_{ijk} = w_{ijl}$ for all k,l and in that case (which is the ordinary case in practice), it turns out that the analysis in §1:6 can be used and, in fact, the very same algorithm (27) minimizes (28). The only difference is that in the partitioned case we must solve m different regressions, while in the unpartitioned case, only one regression over all m matrices is required.

In the general case (where not all weight matrices are the same), the single-phase approach cannot be used any more, but we might try a double-phase (or normalized alternating least-squares) approach. Minimizing $S_2(X,\Delta)$ for fixed X over Δ is easy enough, but minimizing $S_2(X,\Delta)$ over X for fixed Δ is not at all simple and we do not see any method of simplifying this problem.

The above considerations suggest that we use a different loss function in the matrix conditional case with weights. Consider:

$$S_3(X,\Delta) = \Sigma_k\{\Sigma_i\Sigma_j w_{ijk}(\delta_{ijk} - d_{ij}(X))^2/\Sigma_i\Sigma_j w_{ijk}\delta_{ijk}^2\}. \tag{30}$$

In the case of (30) it is possible to require that

$$\Sigma_i\Sigma_j w_{ijk}\delta_{ijk}^2 = m^{-1}, \tag{31}$$

because the cones Γ_k are completely independent and we can introduce separate normalization factors α_k for each of the cones. Minimization of $S_3(X,\Delta)$ under the conditions (31) is again possible by the algorithm (27) with some trivial modifications in the normalization. But, unfortunately, the minimization of (30) under the conditions (31) is not equivalent to unconditional minimization of (30). In the unpartitioned case the normalization conditions $\eta(X) = 1$ and $\eta(\Delta) = 1$ could be imposed without loss of generality. If we try to apply the same tactic here, the denominators do not vanish and the simplifications of §1:6 are not possible. Again, the exception is the case of equal weights, but this is not surprising, since Kruskal and Carroll and, more generally, De Leeuw [1977b] have already shown that in the equal weights case normalizing loss functions by using dissimilarities or distances does not make a difference.

It is possible, however, to apply a double-phase alternating least-squares procedure to (30). This is easily seen. Minimizing (30) over X for fixed Δ is equivalent to an ordinary metric MDS problem and we can use one or several steps of (11). Minimizing (30) over Δ for fixed X means solving *m* separate normalized cone regression problems, which reduce to monotone or linear regression problems in the usual cases. From the general theory of alternating least squares, these double-phase algorithms converge, independent of the number of steps (11) one takes in the first phase.

We summarize the situation for the matrix conditional case. Loss function (21) often gives unsatisfactory solutions. Loss function (28) is better in this respect, but there does not seem to be a simple algorithm to minimize it, comparable to the algorithm in §1:6. An exception is the case in which the weight matrices are the same. Loss function (30) can be minimized by our basic nonmetric algorithm (27), if we impose the conditions (31). If we do not want to impose these conditions, we need a double-phase algorithm, excepting again the case in which all weight matrices are equal.

2:2

The situation becomes slightly more complicated in the row-conditional case, *i.e.*, in those cases in which dissimilarities are compared only within rows and there are *mn* cones Γ_{ki}. In a sense, the matrix conditional algorithms can be used for some row-conditional analyses as well. Suppose $w_{ijk} = w_{ij}\delta^{ik}$, *i.e.*, weight matrix *k* is empty, except for row *k*. Define $\delta_{ij} = \delta_{iji}$. Then:

$$(32) \qquad S_3(X,\Lambda) = \Sigma_i\{\Sigma_j w_{ij}[\delta_{ij} - d_{ij}(X)]^2/\Sigma_j w_{ij}\delta_{ij}^2\}.$$

In some cases, however, we may need a more general loss function.

$$(33) \qquad S_4(X,\Delta) = \Sigma_k\Sigma_i\{\Sigma_j w_{ijk}[\delta_{ijk} - d_{ij}(X)]^2/\Sigma_j w_{ijk}\delta_{ijk}^2\}.$$

Again it turns out that the loss functions with $d_{ij}(X)$ instead of δ_{ijk} in the denominator are inferior from the point of view of the simplicity of algorithms. As Kruskal [1964] has already observed, they seem slightly superior from the intuitive point of view, such that we wish to do as little arithmetic as possible on the dissimilarities themselves. It is not known, at the moment, if the superior algorithmic properties of our loss functions also makes them more more well-behaved numerically.

2:3

There are other forms of partitioning which are interesting, for example, the one suggested by the method of triads [Roskam, 1970]. The principle of loss function and algorithm construction remain the same. Either we have to impose explicit normalization conditions, such as (29) or (31), or we must work with the implicit normalizations that are imposed by the minimization routine. In the first case, we can apply the single-phase approach of §1:6,but the normalizations we need may be much too restrictive or even contradictory. In the second case, there are no problems of this kind, but we have to use the less elegant (albeit more flexible) double-phase algorithm based on (11) and normalized cone regression. Using implicit normalizations may lead to degenerate solutions (as is sometimes found in unfolding, for example), using explicit normalizations, on the other hand, may lead to well-defined but uninteresting solutions.

As a further application of these considerations, we shall now study the loss functions which are obtained if we replace the sum of squares in the denominator with a sum of squared deviations from the mean (Kruskal's stress formula 2 is the simplest example, while Roskam's stress formula 3 is a more complex one). Again, we must choose between explicit and implicit normalizations. Consider:

$$(34) \qquad S_3(X,\Lambda) = \Sigma_k \Sigma_i \Sigma_j w_{ijk} [\delta_{ijk} - d_{ij}(X)]^2 / \Sigma_k \Sigma_i \Sigma_j w_{ijk} (\delta_{ijk} - \bar{\delta})^2$$

with

$$(35) \qquad \bar{\delta} = \Sigma_k \Sigma_i \Sigma_j w_{ijk} \delta_{ijk},$$

where we suppose that

$$(36) \qquad \Sigma_k \Sigma_i \Sigma_j w_{ijk} = 1.$$

It is obvious from (34) that we can use explicit normalizations; minimizing (34) is equivalent to minimizing the numerator on the condition that the denominator is unity (*cf.*, Guttman's "soft-squeeze" [1968]). By applying β as in §1:2, this is equivalent to minimizing $\eta^2(\Delta) - \rho^2(X,\Delta)$ on the condition that $\eta^2(\Delta) - \bar{\delta}^2 = 1$. In this case the projection procedure of §1:6 would be tantamount to minimizing the ratio of two non-negative quadratic forms in Δ over the cone Γ. This is a nontrivial problem and, in general, it cannot be solved by a finite procedure. Thus, the approach of §1:6 is theoretically feasible, but practically more complicated than in the case of $S_1(X,\Delta)$.

Consequently, we have to use the two-step alternating least-squares procedure again, if we want to minimize (34) efficiently. This means, in this case, that we apply the metric algorithm (11) in the first phase. In the second phase we have to solve a normalized cone regression problem again. We can find the optimal δ_{ijk} by first solving the cone regression problem of minimizing

$$(37) \qquad \Sigma_k \Sigma_i \Sigma_j w_{ijk}[\delta_{ijk} - (d_{ij}(X) - \bar{d})]^2$$

over the admissible δ_{ijk} and then by adjusting the mean and scale factors afterwards [De Leeuw, 1977b]. In (37) we have used

$$(38) \qquad \bar{d} = \Sigma_k \Sigma_i \Sigma_j w_{ijk} d_{ij}(X).$$

The same reasoning applies in those cases where we want both to partition the data and to modify the denominator of the loss function. More complicated modifications of the denominator are also sometimes desirable.

3:1

We shall now apply our method to individual differences scaling. More specifically, we define the distance between points i and j on occasion k by

$$(39) \qquad d_{ij}^2(X,C_k) = (x_i - x_j)'C_k(x_i - x_j),$$

which generalizes formula (2) in an obvious way. The matrix C_k is restricted to be symmetric, but in some cases more restrictions may be imposed. It usually makes sense to require that C_k be non-negative definite, for example, and in the popular Horan-Bloxom-Carroll-Chang-Harshman model, we require that C_k is diagonal or diagonal and non-negative.

Because of the extra parameters C_k in (39) the problems with partitioned data sets have to be considered all over again for individual difference models. We shall restrict ourselves to a relatively simple, but very common, special case: matrix conditional data without weights. We use the loss function:

$$(40) \qquad S_6(X,C,\Delta) = \Sigma_k\{\Sigma_i\Sigma_j[\delta_{ijk} - d_{ij}(X,C_k)]^2/\Sigma_i\Sigma_j d_{ij}^2(X,C_k)\}.$$

By using homogeneity, this reduces, in the familiar way, to maximization of

$$(41) \qquad \rho_0(X,C,\Delta) = \Sigma_k \rho_k^2(X,C_k,\Delta_k),$$

where

$$(42) \qquad \rho_k(X,C_k,\Delta_k) = \Sigma_i\Sigma_j\delta_{ijk}d_{ij}(X,C_k).$$

The normalization restrictions are $\eta(\Delta_k) = 1$ for all k and $\eta(X,C_k) = 1$ for all k. We maximize (41) by a three-phase process. Maximization over the admissible Δ_k for fixed X and C is easy, but maximization over X for fixed Δ_k and C is more difficult. If C_k is positive definite, however, the Cauchy-Schwartz inequality once again comes to our aid.

We first use the inequality in the form:

$$(43) \qquad d_{ij}(X,C_k) \geq [d_{ij}(Y,C_k)]^{-1}(x_i - x_j)'C_k(y_i - y_j).$$

Define the matrices Z_k by:

$$(44) \qquad z_{is}^k = \Sigma_{j=1}^n\Sigma_{t=1}^p[(\delta_{ijk} + \delta_{jik})/d_{ij}(Y,C_k)]c_{kst}(y_{it} - y_{jt}).$$

Then, generalizing (10)

$$(45a) \qquad \rho_k(X,C_k,\Delta_k) \geq tr X'Z_k(Y),$$

while

(45b) $$\rho_k(X,C_k,\Delta_k) = tr X'Z_k(X).$$

Results (45) generalize the use of C-matrix type algorithms to individual difference scaling. But, it is not entirely obvious as yet how to use (45) in the maximization of (41). The first possibility for consideration is:

(46) $$\rho_0(X,C,\Delta) \geq \Sigma_{i,j}^{n}\Sigma_{s,t}^{p}x_{is}x_{jt}\Sigma_k^m z_{is}^k z_{jt}^k.$$

We now have a quadratic form, majorized by $\rho_0(X,C,\Delta)$. The normalization condition $\eta(X,C_k) = trC_kX'X = 1$ for all k is not specific enough for our purposes. If C_k is not restricted to be diagonal (three-mode scaling or IDIOSCAL), then we can require, without loss of generality, $X'X = I$ and $trC_k = 1$ for all k; if C_k is restricted to be diagonal (INDSCAL-PARAFAC), then we can require $diag(X'X) = I$ and also $trC_k = 1$. Maximizing (46) under quadratic constraints such as $X'X = I$ or $diag(X'X) = I$ is not easy. But again, the majorization principle can be used in combination with the Cauchy-Schwartz inequality.

If we denote the quadratic form in (46) by $\xi(X,X)$, then Cauchy-Schwartz says that $\xi(X,X) \geq \xi^{-1}(Y,Y)\xi^2(X,Y)$. It follows that we can maximize $\xi(X,X)$ by an iterative algorithm that chooses $X_{\mu+1}$ as the maximizer of the linear form $\xi(Y,X_\mu)$ over all normalized Y. If the normalization is $X'X = I$, then we must solve a Procrustes problem in each step; for the normalization, on the other hand, $diag(X'X) = I$, the steps are much simpler.

There is another, more direct, way in which we can use (45) in the maximization of (41). If we apply Cauchy-Schwartz directly to (41) we find:

(47) $$\rho_0^{1/2}(X,C,\Delta) \geq \rho_0^{-1/2}(Y,C,\Delta)\Sigma_k\rho_k(Y,C_k,\Delta_k)\cdot\rho_k(X,C_k,\Delta_k).$$

If we define Z, or more precisely, Z(Y) by

(48) $$z_{is} = \rho_0^{-1/2}(Y,C,\Delta)\Sigma_k\rho_k(Y,C_k,\Delta_k)z_{is}^k,$$

then

(49a) $\rho_{\delta}^{1/2}(X,C,\Delta) \geq trX'Z(Y),$

and

(49b) $\rho_{\delta}^{1/2}(X,C,\Delta) = trX'Z(X).$

Now (49) can be used in the obvious way. In any case, we have proved that maximization of (41) over X for fixed C and Δ can be done if we use the majorization principle and the Cauchy-Schwartz inequality twice. Of course, it is neither necessary nor desirable to solve the maximization problem in this second phase completely; if we use (49), for example, we will usually solve only one Procrustes problem in this phase and then proceed to the next phase. The question of what the best policy is with respect to the number of inner iterations, the path to be followed through the various phases, *etc.*, has not as yet been answered. It seems difficult to give an analytical answer and considerable numerical experimentation is required before we could give good practical guide lines. It seems fairly obvious, however, that using the majorization principle twice in one phase will make the inequality estimates less sharp and the convergence slower than in the non-individual-differences case.

In phase three, we have to maximize (41) over C for fixed X and Δ, which comes to the same thing as maximizing (42) over C_k for fixed X and Δ_k. We must consider two separate cases: either C_k is diagonal or it is not. In both cases the normalization is $trC_k = 1$ and in both cases we must take care that C_k is non-negative definite, for otherwise the second phase may not work in the next cycle. One easy way to guarantee that C_k is non-negative definite is to write $C_k = T_k'T_k$, with T_k square matrices of order p, and to use the T_k as parameters. The constraint is now $trT_k'T_k = 1$; symmetry and non-negative definiteness are guaranteed automatically. To maximize (42) in this third phase we use, surprisingly enough, the majorization principle and the Cauchy-Schwartz inequality, which is:

(50) $d_{ij}(X,T_k) \geq [d_{ij}(X,S_k)]^{-1}(x_i - x_j)'T_k S_k(x_i - x_j).$

If we define the matrix

(51) $G = \Sigma_i \Sigma_j [\delta_{ijk}/d_{ij}(X,S_k)](x_i - x_j)(x_i - x_j)',$

then

$$(52) \qquad \rho_k(X,T_k,\Delta_k) \geq tr T_k G S_k'.$$

In the IDIOSCAL case, the above means that we must choose the successor of T_k proportional to $T_k G$; in the INDSCAL case, we must choose the successor proportional to $diag(T_k G)$. Again, we can make one or several inner iterations based on (52) in phase three of the algorithm.

3:2

After all these formulas, it seems sensible to look back and see what we have accomplished. In the first place, the algorithm of §3:1 seems to be the first of its type that directly uses distances (see, however, PINDIS, *this book*). INDSCAL and IDIOSCAL first have to transform to scalar products, which makes nonmetric extensions of these approaches complicated. ALSCAL [Takane, Young, & De Leeuw, 1977] must first transform to squared distances, which makes treatment of ordinal data comparatively simple, but which makes the analysis of the additive constant problem complicated. Moreover, it seems to us that if the model is formulated in terms of distances, then the loss function should preferably be the normalized sum of squares of deviations between optimally scaled dissimilarities and distances and not squared distances and not scalar products. In the second place, it is obvious that the majorization approach is a natural one, especially for distances of the form (39). It makes it possible to treat the case with C diagonal and with C full in very much the same way; it also makes it possible to treat metric and nonmetric applications in very much the same way. The first advantage is also true for the scalar product approaches (Schönemann [1972]; De Leeuw & Pruzansky [1977c]), but they fail in the second respect. The second advantage is also true for ALSCAL, but it fails in the first respect. It is clear that our approach can be extended to row-conditional or unconditional data without too much trouble, although all the possible special cases have not as yet been investigated. By using the results of De Leeuw [1977a] we can easily construct individual difference versions of general Minkowski metrics; more specifically, of power metrics and apply the majorization approach to the resulting loss functions. It is clear, however, that general Minkowski metrics lead to more unpleasant computational problems and, furthermore, we do not know how practical these models and algorithms will be. It seems to us that since the majorization approach can be applied to Minkowski metrics without any theoretical complications, this constitutes an additional advantage. Both scalar product and squared distance approaches critically depend upon the Euclidean assumption. If the distances are non-Euclidean, there is no reason at all to square and double center. The majorization approach can also be applied to dual algorithms for individual differences scaling. Dual algorithms do not insist on the model (39); they allow each subject his own configuration matrix Y_k, but they penalize if the Y_k do not satisfy the constraints $Y_k = XT_k$, with T_k either full, or diagonal, or the identity. This generalizes an idea of McGee [1968] and is explained more fully in Heiser [1975] and De Leeuw & Heiser [1977].

REFERENCES

De Leeuw, J. Applications of convex analysis to multidimensional scaling. In: J. R. Barra, *et al.* (Eds.). ***Recent Developments in Statistics.*** Amsterdam, North Holland Publishing Company, 1977, p. 133-145.

De Leeuw, J. Normalized cone regression. (submitted for publication), 1977b

De Leeuw, J. & Pruzansky, S. A new computational method to fit the weighted Euclidean distance model. (submitted for publication), 1977c.

De Leeuw, J. & Heiser, W. Primal and dual algorithms for individual difference scaling. (in preparation), 1977.

Guttman, L. A general nonmetric technique for finding the smallest coordinate space for a configuration of points. **Psychometrika**, 33, 1968, 469-506.

Heiser, W. Individual difference scaling. **Rpt. MT-001-75**, Univ. of Leiden, Psychol. Inst. (Holland), 1975.

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

Kruskal, J. B. Multidimensional scaling: a numerical method. **Psychometrika**, 29, 1964, 115-129.

Kruskal, J. B. & Carroll, J. D. Geometrical models and badness of fit functions. Krishnaiah, P. R. (Ed.). **Multivariate Analysis - II.** Academic Press (New York), 1969.

Lingoes, J. C. & Roskam, E. E. A mathematical and empirical analysis of two multidimensional scaling algorithms. **Psychometrika**, 38, 1973, Monograph Supplement.

McGee, V. E. Multidimensional scaling of N sets of similarity measures: a nonmetric individual differences approach. **Mult. Behav. Res.**, 3, 1968, 233-248.

Roskam, E. E. **Metric Analysis of Ordinal Data in Psychology.** Voorschoten (Holland), VAM, 1968.

Roskam, E. E. The method of triads for nonmetric multidimensional scaling. **Neds. Tijdsch. v. de Psychol.**, 25, 1970, 404-417.

Schönemann, P. H. An algebraic solution for a class of subjective metrics models. **Psychometrika, 37**, 1972, 441-451.

Takane, Y., Young, F. W., & De Leeuw, J. Nonmetric individual differences scaling: an alternating least squares method with optimal scaling features. **Psychometrika, 42**, 1977, 7-67.

Young, F. W., De Leeuw, J., & Takane, Y. **Multidimensional Scaling.** 1977, (in preparation).