PSYCHOMETRIKA—VOL. 49, NO. 3, 311–323
SEPTEMBER 1984

LEAST SQUARES METRIC, UNIDIMENSIONAL UNFOLDING

KEITH T. POOLE

GRADUATE SCHOOL OF INDUSTRIAL ADMINISTRATION
CARNEGIE-MELLON UNIVERSITY

The partial derivatives of the squared error loss function for the metric unfolding problem have a unique geometry which can be exploited to produce unfolding methods with very desirable properties. This paper details a simple unidimensional unfolding method which uses the geometry of the partial derivatives to find conditional global minima; i.e., one set of points is held fixed and the global minimum is found for the other set. The two sets are then interchanged. The procedure is very robust. It converges to a minimum very quickly from a random or non-random starting configuration and is particularly useful for the analysis of large data sets with missing entries.

Introduction

In the metric unfolding problem the data are assumed to be Euclidean distances plus some unknown observational error. Let x_{ik} be the ith individual's estimated coordinate ($i = 1, \ldots, p$) on the kth dimension ($k = 1, \ldots, s$) and let z_{jk} be the estimated coordinate of the jth stimulus ($j = 1, \ldots, q$) on the kth dimension. The corresponding s length vectors will be denoted as $\mathbf{x}_i$ and $\mathbf{z}_j$. The estimated distance between the ith individual and the jth stimulus is therefore

$$d_{ij} = \left[\sum_{k=1}^{s} (x_{ik} - z_{jk})^2 \right]^{1/2}. \tag{1}$$

Let $D^{*1/2}$ be the p by q matrix of data, let $\tilde{D}^{1/2}$ be the p by q matrix of "true" but unknown distances, and let E be a p by q matrix of error. Assume that

$$D^{*1/2} = \tilde{D}^{1/2} + E. \tag{2}$$

I will work with the standard squared error loss function

$$\mu = \text{tr } E'E = \sum_{i=1}^{p} \sum_{j=1}^{q} e_{ij}^2 = \sum_{i=1}^{p} \sum_{j=1}^{q} (d_{ij}^* - d_{ij})^2. \tag{3}$$

The partial derivatives of the z_{jk} and x_{ik} can be written as

$$\frac{\partial \mu}{\partial z_{jk}} = -2 \sum_{i=1}^{p} (z_{jki} - z_{jk}), \tag{4}$$

$$\frac{\partial \mu}{\partial x_{ik}} = 2 \sum_{j=1}^{q} (x_{ikj} - x_{ik}), \tag{5}$$

where

$$z_{jki} = x_{ik} + \frac{d_{ij}^*}{d_{ij}} (z_{jk} - x_{ik}), \tag{6}$$

$$x_{ikj} = z_{jk} + \frac{d_{ij}^*}{d_{ij}} (x_{ik} - z_{jk}). \tag{7}$$

This paper benefits from many conversations with and suggestions from Howard Rosenthal.

Requests for reprints should be addressed to Keith T. Poole, Graduate School of Industrial Administration, Carnegie-Mellon University, Pittsburgh, Pennsylvania 15213.

0033-3123/84/0900-5071$00.75/0

The Geometry of the Partial Derivatives

Gleason (1967) was the first to point out one aspect of the unique geometry of these partial derivatives; namely, as (6) and (7) show, they involve a summation of equations of straight lines. For example, Figure 1 shows the geometry of (6) in the case of two dimensions.

A further unique property of the geometry of these partial derivatives is that the squared distance between the points $\mathbf{z}_{j,i}$ (the k-length vector of the z_{jki}) and $\mathbf{z}_j$ is equal to the squared error between d_{ij}^* and d_{ij}. That is,

$$e_{ij}^2 = \sum_{k=1}^{s} (z_{jki} - z_{jk})^2 = (d_{ij}^* - d_{ij})^2$$

(see figure 1). A similar expression holds for $\mathbf{x}_{i,j}$ and $\mathbf{x}_i$.

In terms of this geometry, the update formulas for the steepest descent method (when the step size is fixed at one) have an interesting form. They can be written as

$$z_{jk}^{(h+1)} = \frac{\sum_{i=1}^{p} z_{jki}^{(h)}}{p}, \tag{8}$$

$$x_{ik}^{(h+1)} = \frac{\sum_{j=1}^{q} x_{ikj}^{(h)}}{q}, \tag{9}$$

where h is the iteration number. The new x_{ik} and z_{jk} are simply the centroids of the points produced by the corresponding straight line equations. Intuitively, the process can be conceptualized as follows. Imagine that the X set of points is fixed and $\mathbf{z}_j$ is placed somewhere in the space. Think of the d_{ij}^* as vectors attached to the respective $\mathbf{x}_i$ and aim them at $\mathbf{z}_j$ (see Figure 2 for a $p = 5$, $s = 2$ example). At the end of the vectors aimed at $\mathbf{z}_j$, place points (the $z_{j,i}$ in Figure 2A). Now move $\mathbf{z}_j$ to the centroid of these points and once again aim the d_{ij}^* vectors so they are pointing at $\mathbf{z}_j$ ($z_j^{(2)}$ in Figure 2B).

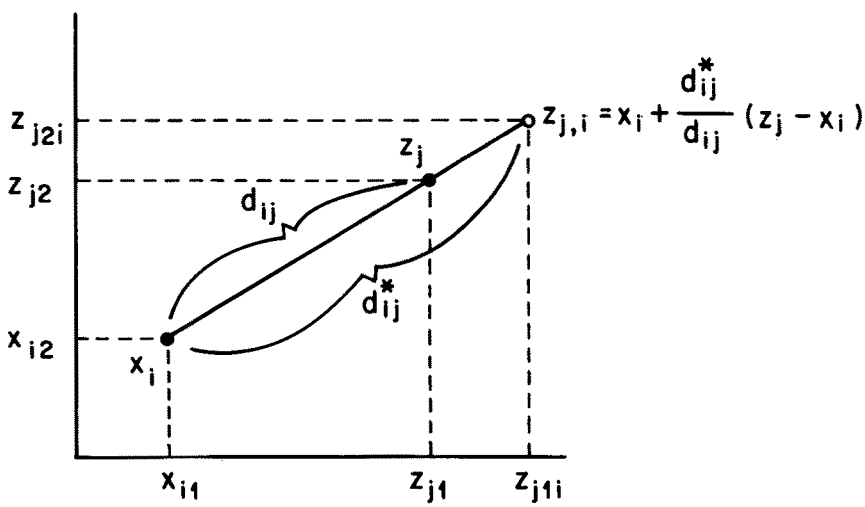

FIGURE 1
Parametric equation of a straight line.

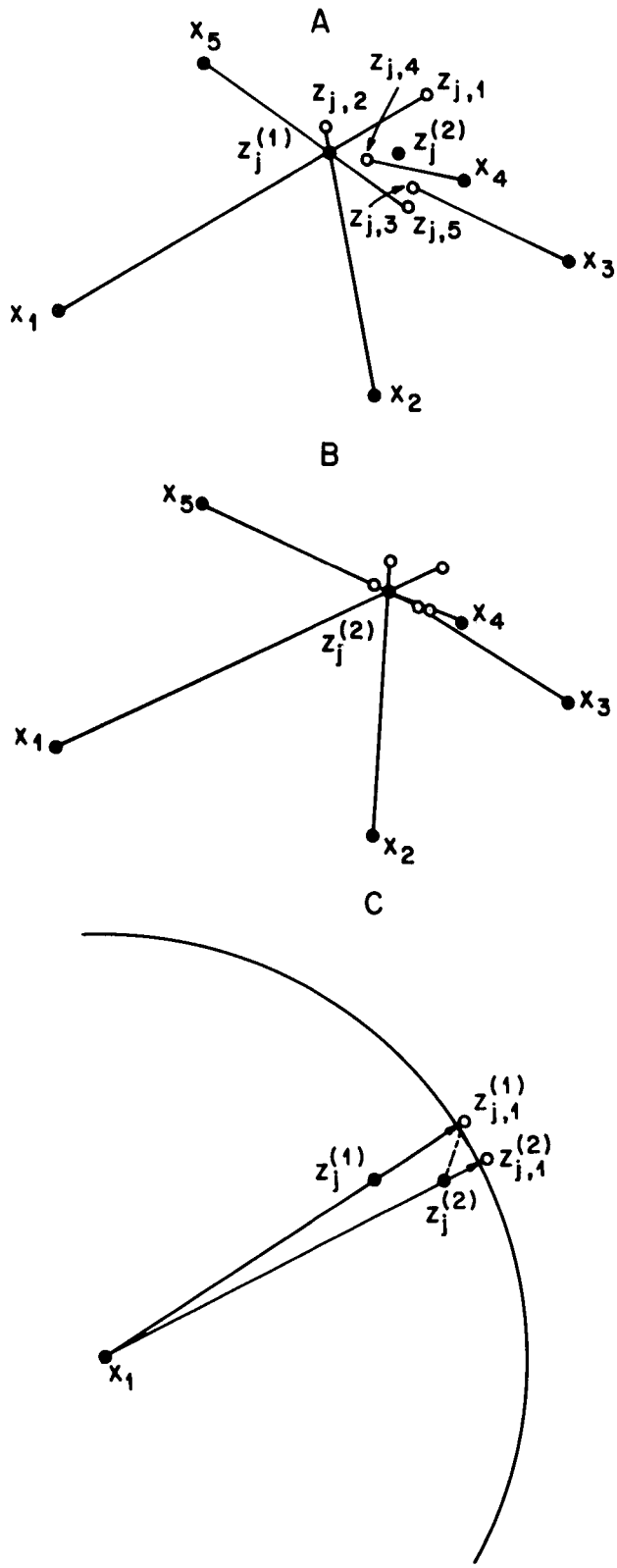

FIGURE 2
Five point example.

That the sum of squared error is reduced is clear from the geometry; that is

$$\sum_{i=1}^{p} e_{ij}^{(2)2} = \sum_{i=1}^{p}\sum_{k=1}^{s}(z_{jki}^{(2)} - z_{jk}^{(2)})^2 \leq \sum_{i=1}^{p}\sum_{k=1}^{s}(z_{jki}^{(1)} - z_{jk}^{(2)})^2 \leq \sum_{i=1}^{p}\sum_{k=1}^{s}(z_{jki}^{(1)} - z_{jk}^{(1)})^2 = \sum_{i=1}^{p} e_{ij}^{(1)2}.$$

The third and fourth terms of the inequality are true because $\mathbf{z}_{jk}^{(2)}$, by (8), is the centroid of the $z_{jki}^{(1)}$. The second and third terms of the inequality are true because the $z_{jki}^{(2)}$ are computed by aiming the d_{ij}^{*} vectors at $\mathbf{z}_j^{(2)}$. The $z_{jki}^{(1)}$ are computed by aiming at $\mathbf{z}_j^{(1)}$. Hence, for every ith term in the summation, it will be the case that

$$\sum_{k=1}^{s}(z_{jki}^{(2)} - z_{jk}^{(2)})^2 \leq \sum_{k=1}^{s}(z_{jki}^{(1)} - z_{jk}^{(2)})^2$$

This geometrical fact is shown in Figure 2C for two dimensions. Figure 2C is the same as Figures 2A and 2B except that only $\mathbf{x}_1$, $\mathbf{z}_j^{(1)}$, $\mathbf{z}_j^{(2)}$, $\mathbf{z}_{j,1}^{(1)}$ and $\mathbf{z}_{j,1}^{(2)}$, are shown for clarity. The partial circle in Figure 2C has radius d_{ij}^{*}. For any point in the plane, the shortest distance from the point to the circle is found by passing a line from the point through the center of the circle. Hence the distance from $\mathbf{z}_j^{(2)}$ to $\mathbf{z}_{j,1}^{(2)}$ must be shorter than the distance from $\mathbf{z}_j^{(2)}$ to $\mathbf{z}_{j,1}^{(1)}$ (the dotted line). A similar argument holds for higher dimensional spaces. Hence, the inequality holds.

The fact that the sum of squared error, using (8) and (9), will *always* decrease or remain the same, is very unusual. For most nonlinear procedures, a step size of one does *not* guarantee error reduction. Here it does.

Because of this geometry, several different gradient procedures are possible. One set of points can be held fixed and the other set repeatedly estimated as described above. This process eventually arranges the d_{ij}^{*} vectors so that they all are aimed at the centroid of the points placed at the ends of the vectors—when this occurs, a minimum is reached. The process can then be repeated with the other set of points held fixed. Another procedure, which is basically the same as the first, is to hold one set fixed and estimate the other set only *once* (in contrast to the first procedure where the set is repeatedly reestimated), hold it fixed and estimate the first only once; and so on (hereafter referred to as the *standard gradient procedure*).

Given this geometry, μ can be minimized without the use of update formulas of the form (8) and (9). The update formulas are, in effect, providing new "targets" for the d_{ij}^{*} vectors. No targets are necessary for convergence to occur. Conceptually, the gradient process embodying (6)–(9) can be seen as a way of drawing the ends of the d_{ij}^{*} vectors attached to one set of points as close together as possible. Viewed from this perspective, *it does not matter whether or not all the d_{ij}^{*} vectors are moved simultaneously.* The $\mathbf{z}_{j,i}$ can be brought closer together by moving one d_{ij}^{*} vector at a time. This will always reduce μ. In fact, any subset of $\mathbf{z}_{j,i}$ can be moved closer to the remaining $\mathbf{z}_{j,i}$ by aiming them at the centroid of the second set. This process is fundamentally different from a standard gradient procedure and other procedures that use modifications of (4) and (5) to arrive at update formulas (e.g., Guttman's (1968) C-matrix method; DeLeeuw (1977), and Heiser and DeLeeuw (1979a,b)). In these and other similar procedures, the new estimates of one set of points are weighted or unweighted sums of *all the points in the other set.*

Conditional Global Minima in One Dimension

All the procedures discussed above will converge to a solution of the partial derivatives (however, provision must be made in all the procedures for the possibility that some $\mathbf{x}_i$ may equal some $\mathbf{z}_j$ at each step of the iteration). There is no way to guarantee that the solution found by any of the procedures is the global minimum. *Even if one set of points is held fixed,* there is no way to guarantee that the *conditional global minimum* (i.e., con-

ditioned on one set being fixed) for the other set will be found unless a grid search is performed. This is a problem particularly in one dimension. As Spence (1978, p. 216) notes: "Most scaling algorithms are fairly robust in most situations. However, suboptimal solutions do occasionally occur, especially in one dimension...."

In one dimension, the geometry shown in Figures 1 and 2 can be exploited to produce a simple procedure that finds the global minimum for one set of points when the other set is held fixed. Figure 3 shows five individual points arrayed along one dimension. The five points are shown as the centers of circles with the distances as the radii at the top of Figure 3 for pictoral convenience.

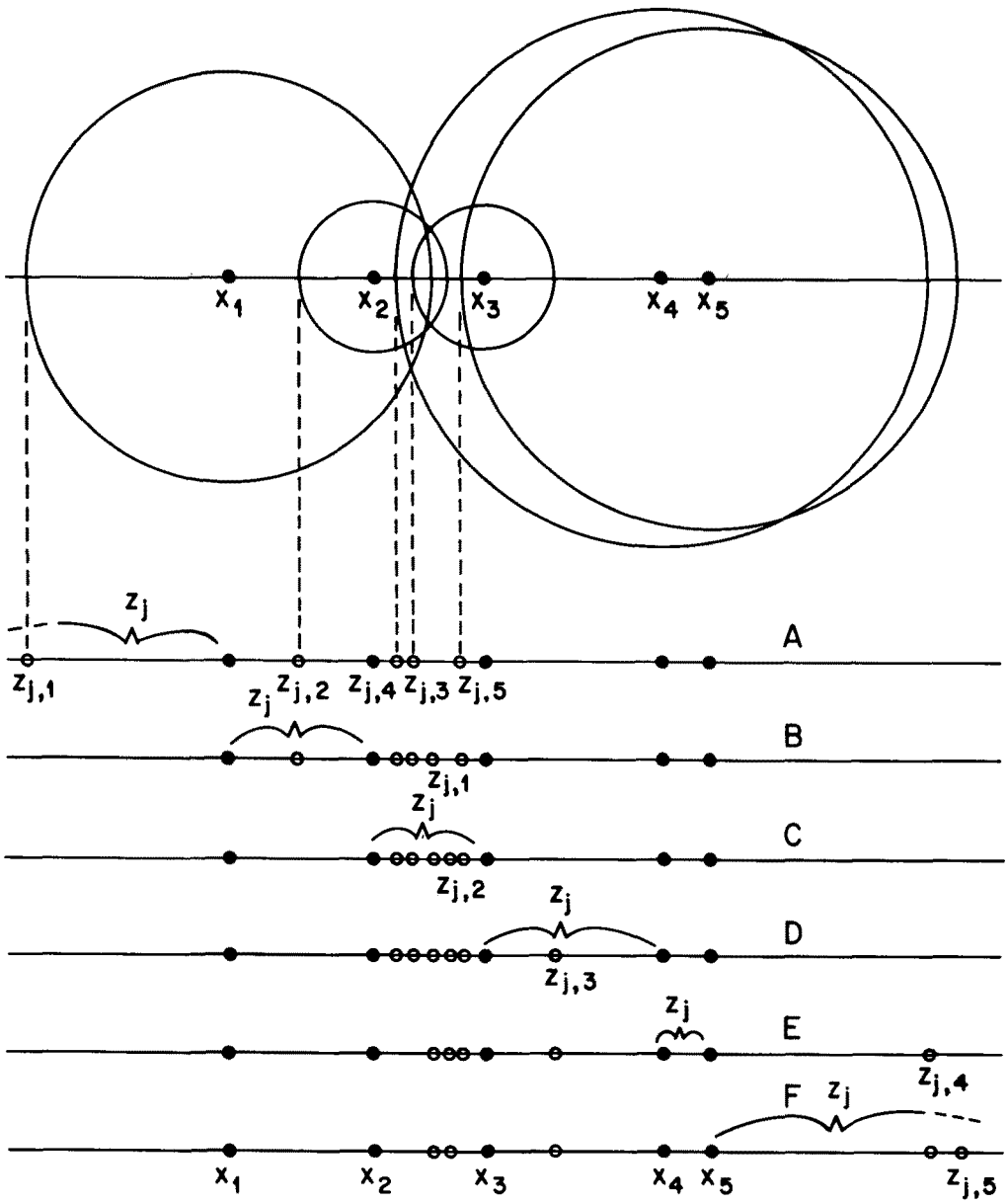

FIGURE 3
Five point one dimensional example.

Suppose the initial estimate of z_j is to the left of x_1, as shown in line A of Figure 3. The $z_{j,i}$ points produced by aiming the d_{ij}^* vectors at z_j are represented as small circles on the line. It is immediately obvious from the top part of the figure that the global minimum (the only minimum in this example) is between x_2 and x_3. This situation is represented on line C, which shows the $z_{j,i}$ produced by placing z_j between x_2 and x_3. Notice that when z_j is moved from anywhere in the region to the left of x_1 (line A) to anywhere in the region between x_1 and x_2 (line B) there is only one change in the configuration of $z_{j,i}$ points; namely, $z_{j,1}$, the point produced by aiming x_1 at z_j has switched from the left of x_1 to the right of x_1 because z_j is now to the right of x_1. Similarly, when z_j is moved from the region between x_1 and x_2 to the region between x_2 and x_3 (line C) the only change in the $z_{j,i}$ configuration is $z_{j,2}$. In sum, when z_j is moved across the dimension from the left to the right, each time it passes an x_i point the corresponding $z_{j,i}$ point switches to being on the same side of x_i as z_j.

The geometry of Figure 3 shows the truth of the following simple theorem:

Theorem: In one dimension, if X is held fixed, then a minimum exists if and only if $\sum_{i=1}^{p} z_{j,i}/p$ falls in the same region as z_j.

To see the truth of the theorem, consider Figure 3 again. There are $p + 1$ regions z_j could be in. For each of these regions, there is a corresponding configuration of $z_{j,i}$ (lines A through F). Each of these $p + 1$ configurations has a centroid—$\sum_{i=1}^{p} z_{j,i}/p$. The centroid of a particular configuration can also fall in any of the $p + 1$ regions. Label this centroid $z_j^{(2)}$. If $z_j^{(2)}$ falls in the same region as $z_j^{(1)}$, it will reproduce itself because when it is the target of the x_i, the same $z_{j,i}$ configuration will be produced. A similar theorem holds when Z is held fixed.

The geometry of Figure 3 can be exploited to produce a very simple algorithm which finds the global minimum for one set of points when the other set is held fixed. Suppose one set of points is placed in ascending order as shown in Figure 3. Following Figure 3, first compute the $pz_{j,i}$ points for z_j placed to the left of the smallest x_i (x_1 in the figure). The formula for the $z_{j,i}$ is simply

$$z_{j,i} = x_i - d_{ij}^*$$

Compute $\sum_{i=1}^{p} z_{j,i}$ and $\sum_{i=1}^{p} z_{j,i}^2$ and store them. In the second step, assume that z_j is between the two smallest x_i (x_1 and x_2 in the figure). To get the correct sum of the $z_{j,i}$, simply take the stored sum from the previous step, subtract off the $z_{j,i}$ corresponding to the smallest x_i from the previous step, and add on the new $z_{j,i}$ from this step. As noted above, only one $z_{j,i}$ value will change each time an x_i is crossed. In terms of Figure 3

$$\sum_{i=1}^{p} z_{j,i}^{(\text{new})} = \sum_{i=1}^{p} z_{j,i}^{(\text{old})} - (x_1 - d_{1j}^*) + x_1 + d_{1j}^*.$$

Now compute the sum of squares for the $z_{j,i}$ using

$$p \sum_{i=1}^{p} z_{j,i}^2 - \left(\sum_{i=1}^{p} z_{j,i} \right)^2$$

for the old and new and store whichever $\sum_{i=1}^{p} z_{j,i}$ corresponds to the minimum sum of squares (the one for which the $z_{j,i}$ are the closest together). Now assume that z_j falls between the second and third smallest x_i (x_2 and x_3 in Figure 3) and repeat the above steps storing the current sum of $z_{j,i}$ and $z_{j,i}^2$ and the sum of $z_{j,i}$ which corresponds to the minimum SSE found thus far. This process is continued until z_j is assumed to be the right of the largest x_i. At the end of this process $\sum_{i=1}^{p} z_{j,i}/p$ corresponding to the minimum

SSE now becomes z_j—the global minimum point. *Note that this procedure requires only two loops of size p for each* z_j. In the first loop, the starting values of the $z_{j,i}$ are sequentially replaced by new values produced by moving the "target" (z_j) past each x_i.

In summary, given a starting configuration, the conditional global minimum (CGM) algorithm is

1. Rank order X
2. Compute starting values of $z_{j,i}$ (loop of size p for each z_j)
3. Find conditional global minimum (loop of size p for each z_j)
4. Rank order Z computed at step 3
5. Compute starting values of $x_{i,j}$ (loop of size q for each x_i)
6. Find conditional global minimum (loop of size q for each x_i)

One pass through 1–6 (or 4–6, 1–3) is equivalent to roughly $2\frac{1}{2}$ iterations of a standard gradient procedure. It takes $2pq$ computations in a standard gradient to get new X and Z in contrast to $4pq$ computations plus the rank ordering calculations. The rank orderings do not add significantly to the time it takes to compute 1–6 because very efficient algorithms exist to rank order real numbers.

The CGM procedure guarantees that one set of points is at the global minimum conditioned on the other set being fixed. The sets are interchanged and the set that was fixed is now placed at the global minimum conditioned on the other set being fixed. The sum of squared error never increases—it decreases monotonically. When convergence occurs, the two sets reproduce each other. This is also true, by definition, of other procedures. However, the CGM procedure converges to a configuration in which *every point is at its global minimum when the other points are fixed.* Other procedures only converge to a configuration in which every point is at a *local* minimum when the others are fixed. If such a configuration was used as the starting coordinates for the CGM procedure it would descend to the stronger form of local minimum.

The overall global minimum corresponds to a configuration in which no subset of points can be moved vis a vis the remaining points without increasing the sum of squared error. Although the CGM procedure cannot guarantee convergence to this overall global minimum, it does converge to a particularly stable form of local minimum. Monte Carlo work, summarized below, shows that the configurations that correspond to these stable local minima are almost always closer to the true configurations than those produced by other procedures.

I performed a Monte-Carlo study in order to test the ability of the CGM procedure to recover $\tilde{D}^{1/2}$ when error is present in the distances ($D^{*1/2}$). The distance matrices were produced by generating random coordinates for X and Z from a uniform [0, 1] distribution. Three different types of error were extensively tested. In the first, the d^* were assumed to be normally distributed with mean $\tilde{d}$ and constant variance σ^2. Under this assumption, X and Z which minimize the loss function (3) will be maximum likelihood estimates. In the second error model, the d^* were assumed to be normally distributed with mean $\tilde{d}$ and variance $(\sigma\tilde{d})^2$. This model is more realistic than the former since one would expect that perceptual error would be small for stimuli near an individual and somewhat larger for distant stimuli. Finally, in the third model, the logarithm of d^* was assumed to be normally distributed with mean $\log \tilde{d}$ with constant variance σ^2. As Ramsay (1977, p. 245) notes: "The lognormal distribution attaches positive probability to all positive values of [d^*], has an expectation which is close but not exactly equal to [$\tilde{d}$], and has considerable positive skewness." This distribution is more realistic than the first two because of its skewness. When $\tilde{d}$ is small, the perceptual error must be positively biased making a sym-

metric distribution of error dubious. (In the first two models, if a negative d^* was produced, it was replaced by its absolute value.)

A representative portion ($p = 40$, $q = 10$, five levels of error) of the results of the Monte-Carlo study are shown in Table 1. Other results are summarized below. The error introduced into the $\tilde{D}^{1/2}$ matrices was increased in stages to correspond to a $0 - .7$ range of STRESS (first column of Table 1). The error level was controlled by varying the standard deviation.

The top third of Table 1 reports the results for the normal, constant variance model, the center third reports the results for the normal, variable variance model, and the bottom third reports the results for the log normal model. Each third of Table 1 is divided into two sections. The first section, "Eckart-Young Starts", shows the Monte-Carlo results when the starting configuration for each procedure was generated from an Eckart-Young (1936) decomposition of the double centered distance matrix. The second section, "Random Starts", shows the results when the starting configurations were randomly generated. Each section shows the recovery of the d^* as measured by STRESS and the recov-

Table 1

MONTE-CARLO RESULTS

NORMAL

Error Introduced	Eckart-Young Starts: Recovery of Distances			Eckart-Young Starts: Recovery of Coordinates			Random Starts: Recovery of Distances			Random Starts: Recovery of Coordinates		
	CGM	Std. Grad.	C Matrix	CGM	Std. Grad.	C Matrix	CGM	Std. Grad.	C Matrix	CGM	Std. Grad.	C Matrix
.10@	.082*	.082	.082	.998#	.998	.998	.082*	.355	.331	.998#	.672	.720
	(.003)	(.003)	(.003)	(.000)	(.001)	(.000)	(.003)	(.205)	(.219)	(.000)	(.249)	(.272)
.25	.215	.218	.218	.982	.978	.978	.218	.449	.386	.982	.577	.704
	(.009)	(.011)	(.011)	(.005)	(.010)	(.009)	(.009)	(.130)	(.168)	(.006)	(.199)	(.285)
.40	.335	.343	.344	.945	.929	.927	.375	.537	.467	.883	.474	.654
	(.017)	(.014)	(.014)	(.015)	(.019)	(.019)	(.030)	(.099)	(.128)	(.038)	(.177)	(.271)
.55	.452	.455	.459	.851	.853	.850	.469	.611	.597	.791	.370	.459
	(.016)	(.016)	(.020)	(.034)	(.038)	(.032)	(.022)	(.082)	(.078)	(.048)	(.164)	(.218)
.70	.568	.591	.589	.590	.600	.610	.575	.674	.664	.452	.320	.349
	(.014)	(.034)	(.031)	(.130)	(.118)	(.120)	(.019)	(.044)	(.067)	(.188)	(.131)	(.202)
				NORMAL VARIABLE VARIANCE								
.10	.096	.096	.096	.998	.998	.096	.096	.398	.343	.998	.632	.724
	(.004)	(.004)	(.004)	(.001)	(.001)	(.001)	(.004)	(.178)	(.218)	(.001)	(.213)	(.270)
.25	.221	.221	.221	.988	.987	.987	.221	.372	.389	.988	.736	.758
	(.010)	(.010)	(.010)	(.004)	(.004)	(.003)	(.011)	(.133)	(.191)	(.004)	(.204)	(.265)
.40	.349	.349	.349	.966	.966	.964	.353	.460	.501	.965	.736	.704
	(.016)	(.016)	(.016)	(.010)	(.015)	(.015)	(.016)	(.113)	(.133)	(.009)	(.208)	(.238)
.55	.482	.488	.488	.931	.905	.905	.509	.592	.600	.905	.669	.676
	(.022)	(.029)	(.029)	(.020)	(.057)	(.057)	(.039)	(.103)	(.196)	(.027)	(.211)	(.212)
.70	.586	.606	.605	.875	.804	.800	.606	.692	.685	.858	.615	.634
	(.022)	(.049)	(.049)	(.034)	(.129)	(.141)	(.033)	(.104)	(.096)	(.053)	(.233)	(.209)
				LOG-NORMAL								
.10	.097	.097	.097	.998	.998	.998	.097	.395	.344	.988	.635	.724
	(.004)	(.004)	(.004)	(.001)	(.001)	(.001)	(.004)	(.175)	(.217)	(.000)	(.212)	(.270)
.25	.214	.214	.214	.989	.988	.989	.214	.357	.384	.989	.763	.760
	(.008)	(.008)	(.008)	(.002)	(.002)	(.002)	(.008)	(.129)	(.190)	(.003)	(.192)	(.264)
.40	.345	.345	.345	.962	.962	.961	.349	.473	.485	.961	.704	.710
	(.014)	(.014)	(.014)	(.015)	(.014)	(.014)	(.016)	(.130)	(.133)	(.014)	(.248)	(.252)
.55	.473	.526	.523	.922	.762	.777	.493	.596	.603	.910	.636	.634
	(.026)	(.111)	(.106)	(.021)	(.278)	(.257)	(.037)	(.139)	(.129)	(.020)	(.311)	(.271)
.70	.631	.727	.720	.717	.444	.453	.689	.769	.781	.722	.445	.410
	(.103)	(.134)	(.125)	(.314)	(.340)	(.324)	(.135)	(.124)	(.099)	(.210)	(.278)	(.234)

@Mean STRESS introduced into the true distance matrices
*Entries are mean STRESS values. Standard deviations are in parentheses.
#Entries are mean Pearson r-square values. Standard deviations are in parentheses.

ery of the true coordinates as measured by the Pearson r-square between the true and reproduced coordinates. Each entry is the average of 10 runs made at the indicated error level with the cooresponding standard deviation shown in parentheses below the average. A step-size of 1 was used in the standard gradient procedure.

When Eckart-Young starts are used at low levels of all three types of error, the standard gradient and C-matrix procedures achieve results equally as good as the CGM procedure. At higher levels of error, the CGM procedure is clearly better—especially with the log-normal model. When random coordinates are used as starts, the CGM procedure is markedly superior to the other two procedures. The CGM procedure recovers the input distances and the true coordinates almost as well from random starts as it does from Eckart-Young starts. In addition, the standard deviations for the CGM procedure are about the same level of magnitude for either type of starts which indicates that the procedure is very stable. That is, the CGM procedure recovers basically the same configuration regardless of starting configuration. The r-square between the configurations recovered from an Eckart-Young as opposed to a random start is almost always greater than .95 regardless of the error model.

Less extensive testing of other matrices and at other levels of error besides those shown in Table 1 revealed that the accuracy advantage of the CGM procedure over the other procedures increases with matrix size. Holding p fixed and increasing q or holding q fixed and increasing p—with the level of error held constant—resulted in an increase in the accuracy advantage for the CGM procedure.

Finally, in addition to being more accurate and stable, the CGM procedure is, on average, 30 to 35 percent faster in terms of CPU time to convergence. It converges in about three iterations (an iteration being one pass through steps 1–6 above) from an Eckart-Young start and about six iterations from a random start.

The robustness of the CGM procedure with starting configurations is particularly useful when there is missing data. When a substantial number of distances are missing in $D^{*1/2}$, producing a good starting configuration is quite difficult. This is no hindrance to the CGM procedure. In fact, the CGM procedure has the further advantage that only coordinates for *one* set of points are necessary to start the process. If starting coordinates are available for both sets of points, then the CGM procedure can be conducted in "parallel" for both sets which provides a check upon the final configuration. For example, given starting values for X, steps 1–3 can be performed to arrive at an estimate of Z. This estimate plus the original (but unused) starting values for Z produce two starting configurations for 4–6 thereby producing two new estimates for X; and so on. Given the simplicity and speed of the CGM algorithm, a plethora of starting configurations (including random starts) can be tried in one run of the program thereby providing the researcher with some confidence in the final configuration.

The CGM algorithm can also be applied with equal success to the metric similarities problem ($p = q$, $X = Z$). For small matrices, the CGM procedure is always faster and more accurate but the accuracy advantage is not significant. However, for large matrices the accuracy advantage is significant and the speed of the procedure gives it a particular advantage. When the similarities matrix is quite large, decomposition of the matrix to obtained starting coordinates can be computationally burdensome. Because the CGM procedure produces basically the same coordinates from a random start as it does from an Eckart-Young start, large similarities matrices can be scaled easily with the CGM procedure. As a consequence, because of its speed and its accuracy and stability over a variety of error models, the CGM procedure would be a useful adjunct to any general purpose scaling program.

An Application to Interest Group Ratings of Members of Congress

The robustness and speed of the CGM procedure make it very useful for performing unidimensional scaling of large data sets with missing data. As an example, consider interest group ratings of members of Congress.

Every year a wide variety of interest groups issue ratings of the members of the House and Senate. These ratings are expressed in percentage terms—that is, from 0 percent to 100 percent approval of the way the Representative/Senator voted on issues of concern to the group. In spatial terms, a rating of 100 means that the interest group is very close to the legislator because the legislator voted exactly as the group would have if it had been in the legislature. A lower rating means that the group is less pleased with a legislator and is therefore more distant than if the rating was higher.

To convert the ratings to distances, they were subtracted from 100 and divided by 50; that is

$$d_{ij}^{*} = (100 - \delta_{ij})/50 = d_{ij} + e_{ij}, \tag{10}$$

where δ_{ij} is the rating of the ith member of Congress by the jth interest group. This transforms the ratings from a 0–100 scale to a 2–0 scale. The division by 50 has no effect upon the recovery of the coordinates. It is done only to confine the recovered coordinates to a $-1/+1$ range. Elsewhere (Poole, 1981; Poole and Daniels, 1984), I have discussed issues relating to (10) in detail, and I will not repeat them here.

Table 2 shows the unfolding results for the 96th Congress. The interest group ratings are substantially one dimensional. A one dimensional Euclidean configuration explains approximately 82 percent of the variance of 19,000 ratings in 1979 and 77 percent of the variance of 15,000 ratings in 1980. The quality of the fit for 1979 can be seen clearly in Figure 4. Figure 4 is the Shepard diagram for the 1979 unfolding. Fitting a regression line through this scatter plot (with d^{*} as the dependent variable and d as the independent variable) produces an intercept of .040 and a slope of .970. The Pearson r-square of the regression is the value shown in Table 2.

Table 3 displays the unfolding results by interest group (not all of the groups had issued their ratings for 1980 when these data were analyzed). The interest groups cover almost the entire spectrum of contested issues in American politics: peace groups, womens' groups, labor unions, civil liberties groups, senior citizens' groups, consumer groups, Christian fundamentalists, and so on. The fits for all groups except two of the farmers' organizations (NFU and NFO) are uniformly high. The recovered dimension is the standard left/right liberal/conservative continuum familiar to students of politics.

TABLE 2

Unidimensional Unfolding Results for the 96th Congress

	1979	1980
One Dimensional Fit (Pearson r-square)	.8167	.7701
Interest Groups	37	28
Senators	100	100
Representatives	435	436
Number of Ratings	19,339	14,965

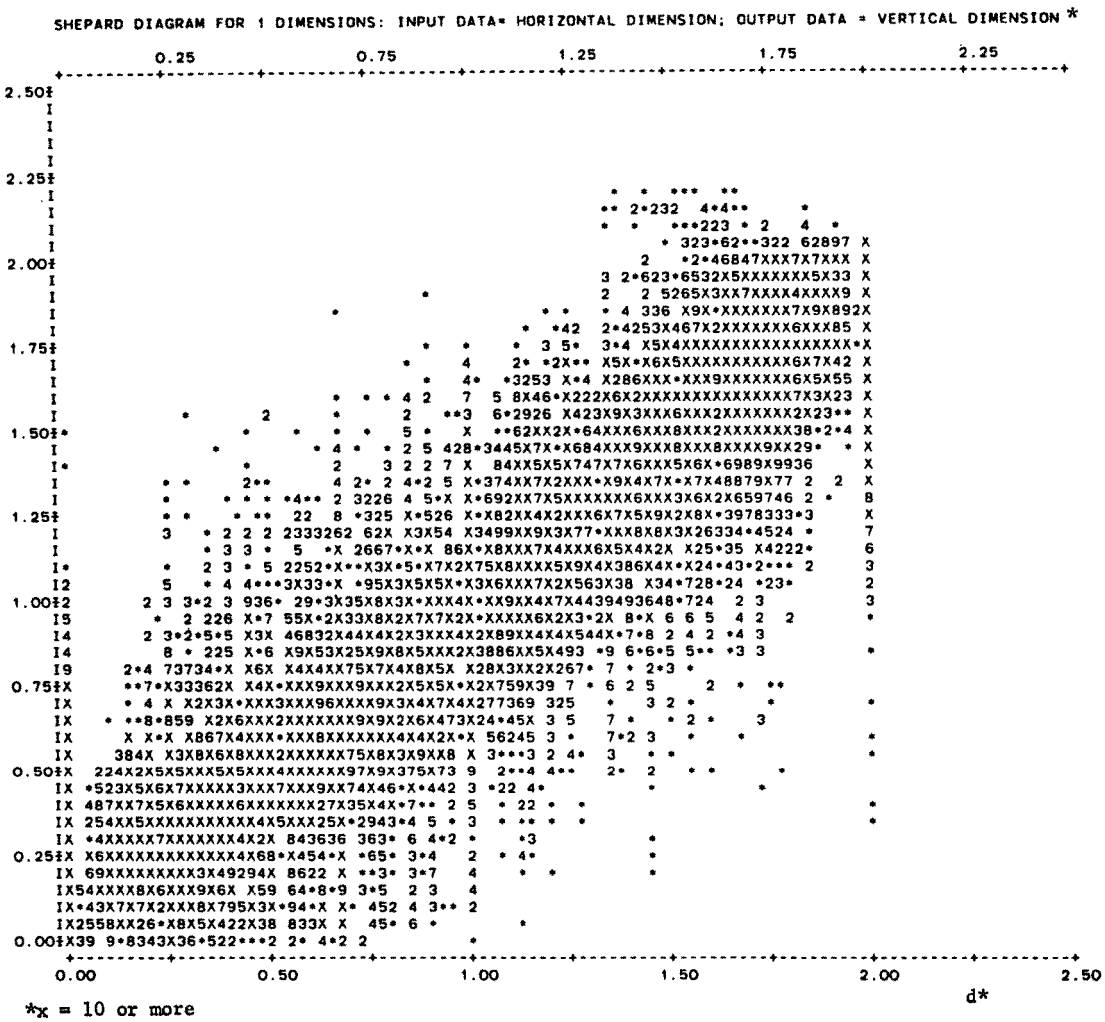

FIGURE 4
Shepard plot for 1979 interest group data.

Conclusion

The Conditional Global Minimum procedure is a unidimensional scaling method which exploits the unique geometry of the partial derivatives of the squared error loss function. It is a very robust procedure; it converges to a minimum very quickly from a random or non-random starting configuration. This makes the CGM procedure particularly useful in analyzing matrices with many missing elements where it is difficult to obtain good starting coordinates as well as large matrices, that is, matrices where p and q are both large, thereby making decomposition of the matrix particularly burdensome computationally.

In addition, by definition, the CGM procedure does not get caught in local minima at any stage of the iteration. Local minima are a serious problem in unidimensional scaling. The robustness of the CGM procedure with arbitrary starting configurations, and the fact that only starting coordinates for one set of points (in the unfolding problem) are needed to begin the procedure, allows for great flexibility in program design. That is,

TABLE 3

Unfolding Results by Interest Group

Interest Group	Location 1979	Location 1980	r^2 1979	r^2 1980
American Civil Liberties Union	-1.062	-1.064	.684	.886
American Conservative Union	1.136	1.168	.943	.887
Americans for Constitutional Action	1.150	1.095	.916	.869
Americans for Democratic Action	-1.090	-1.032	.933	.934
American Farm Bureau Federation	.761		.607	
American Federation of State, County and Municipal Employees	-.946	-.976	.775	.479
American Federation of Teachers	-.995	-1.008	.836	.840
American Security Council	.942	.928	.917	.840
Building and Construction Trades	-.713		.562	
Bread for the World	-.795	-.809	.863	.867
Chamber of Commerce of U.S.	1.002	.569	.917	.274
CCUS 2nd Rating*	.895		.879	
Committee for the Survival of a Free Congress#	1.087	1.028	.958	.907
CFSC Economic Issues	1.066	1.031	.911	.867
CFSC Defense Issues	1.085	.930	.922	.839
CFSC Social Issues	1.119	1.116	.914	.830
Child Welfare League of America	-.789	-.976	.852	.780
Christian Voice	1.080		.850	
Coalition for a New Foreign Military Policy	-1.099	-1.115	.698	.798
Committee on Political Education	-.920	-.997	.907	.815
Congress Watch	-1.115	-1.070	.854	.849
Conservative Coalition@	.884	.915	.947	.952
Consumer Federation of America	-1.166		.884	
Friends Committee on National Legislation	-1.028	-1.125	.884	.861
League of Conservation Voters	-1.026	-1.003	.758	.575
League of Women Voters	-.920	-.876	.868	.641
National Alliance of Senior Citizens	.989	1.047	.907	.886
National Council of Senior Citizens	-.935	-.969	.877	.895
National Education Association	-.960		.679	
National Farmer's Organization	-.127	-.350	.109	.350
National Farmer's Union	-.405		.483	
National Federation of Dependent Business	.772	.788	.838	.801
National Taxpayer's Union	1.280	1.291	.761	.739
National Womens' Political Caucus	-.977		.818	
President Carter@	-.759	-.754	.882	.721
United Auto Workers	-.969	-.963	.956	.919
United Mine Workers	-1.029		.843	

*CCUS issued a second set of ratings for the Senate in 1979.

#CFSC issues four sets of ratings: one for all issues; one for Economic issues only; one for defense issues only; and one for "social" (busing, etc.) issues only.

@ Compiled by Congressional Quarterly. The scores were corrected to remove absences.

several different starting configurations can be tested simultaneously thereby increasing the reliability of the final configuration and any subsequent substantive interpretations.

REFERENCES

De Leeuw, Jan (1977). "Applications of Convex Analysis to Multidimensional Scaling." *Recent Developments in Statistics*, ed. by J. R. Barra et al. Amsterdam, North Holland Publishing Company, pp. 133–145.

Eckart, Carl and Gale Young (1936). "The Approximation of One Matrix by Another of Lower Rank." *Psychometrika*, 1:211–218.

Gleason, Terry C. (1967). "A General Model for Nonmetric Multidimensional Scaling." *Michigan Mathematical Psychology Program*, 3, Ann Arbor, Michigan.

Guttman, Louis (1968). "A General Nonmetric Technique for Finding the Smallest Coordinate Space for a Configuration of Points." *Psychometrika*, 33:469–506.

Heiser, Willem and Jan De Leeuw (1979a). "Metric Multidimensional Unfolding." Paper presented at the annual meeting of the Psychometric Society. Uppsala, Sweden, 1978.

Heiser, Willem and Jan De Leeuw (1979b). "Multidimensional Mapping of Preference Data". Department of Data Theory, University of Leyden, The Netherlands.

Poole, Keith T. (1981). "Dimensions of Interest Group Evaluation of the U.S. Senate, 1969–1978." *American Journal of Political Science* 25: 49–67.

Poole, Keith T. and R. Steven Daniels (1984). "Ideology, Party and Voting in the U.S. Congress." Working Paper #26-83-84, Graduate School of Industrial Administration, Carnegie-Mellon University.

Ramsay, J. O. (1977). "Maximum Likelihood Estimation in Multidimensional Scaling." *Psychometrika*, 42: 241–266.

Spense, Ian (1978). "Multidimensional Scaling," in *Quantitative Ethology*, Edited by Patrick W. Colgan. New York: Wiley.

Manuscript received 2/15/83
Final version received 2/28/84