13

Special Solutions, Degeneracies, and Local Minima

In this chapter, we explain several technical peculiarities of MDS. First, we discuss degenerate solutions in ordinal MDS, where Stress approaches zero even though the MDS distances do not represent the data properly. Then we consider MDS of a constant dissimilarity matrix (all dissimilarities are equal) and indicate what configurations are found in this case. Another problem in MDS is the existence of multiple local minima solutions. This problem is especially severe for unidimensional scaling. For this case, several strategies are discussed that are less prone to local minima. For full-dimensional scaling, in contrast, it is shown that the majorization algorithm always finds a globally optimal solution. For other dimensionalities, several methods for finding a global minimum exist, for example, the tunneling method and distance smoothing.

13.1 A Degenerate Solution in Ordinal MDS

In the various MDS applications discussed so far in this book, we assumed that the loss function employed to find the MDS configuration $\mathbf{X}$ would actually work in the desired sense. In particular, a low Stress value was interpreted as an index that the given proximities were well represented by the distances of $\mathbf{X}$. But is that always true? In Section 3.2, we noticed, for example, that if one minimizes raw Stress, a trivial solution is possible: if $\mathbf{X}$ is made smaller and smaller over the iterations, raw Stress can be arbitrarily reduced, even though proximities and distances are not systematically

TABLE 13.1. Correlations of some KIPT subtests of Guthrie (1973). The lower triangular elements contain the correlations, the upper triangular the rank-order of the correlations in decreasing order.

Subtest	NP	LVP	SVP	CCP	NR	SLP	CCR	ILR
Nonsense word production (NP)	-	9	4	1	6	19	10	12
Long vowel production (LVP)	.78	-	1	7	5	21	20	22
Short vowel production (SVP)	.87	.94	-	3	2	17	16	23
Consonant cluster production (CCP)	.94	.83	.90	-	7	14	11	16
Nonsense word recognition (NR)	.84	.85	.91	.83	-	17	15	18
Single letter production (SLP)	.53	.47	.56	.60	.56	-	13	16
Consonant cluster recognition(CCR)	.72	.48	.57	.69	.59	.62	-	8
Initial letter recognition (ILR)	.66	.45	.44	.57	.55	.57	.82	-

related. Therefore, one has to avoid this outcome (e.g., by using normalized Stress as a loss criterion), because it may—and usually does—lead to a pseudo solution that does not represent the data in the desired sense.

MDS configurations where the loss criterion can be made arbitrarily small irrespective of the relationship of data and distances are called *degenerate* solutions of the particular loss function. They can be avoided, in general, by imposing additional constraints onto the loss function. One example was shown above for raw Stress, where the constraint is a normalization of raw Stress or the requirement that $\mathbf{X}$ must not shrink.

In ordinal MDS, there exist further degenerate solutions, even when using normalized Stress. These solutions arise for particular data. Consider an example. Table 13.1 presents a matrix of correlation coefficients on eight subtests of the Kennedy Institute Phonics Test (KIPT), a reading skills test (Guthrie, 1973). If we scale these data by ordinal MDS in a plane, we obtain the configuration shown in Figure 13.1a. There are just three groups of points. One contains the subtests NP, LVP, SVP, CCP, and NR in a very tight cluster in the upper right-hand corner; a second contains CCR and ILR, also very close together in the upper left-hand corner; finally, point SLP is clearly separated from both of these clusters, with essentially the same distance to either one of them. We find, furthermore, that the MDS solution appears to be almost perfect, because its Stress value is practically equal to zero (i.e., smaller than the stopping criterion for the MDS algorithm).

The Shepard diagram in Figure 13.1b reveals, however, some peculiarities. The data, which are relatively evenly distributed over an interval from $r = .44$ to $r = .94$ (see Table 13.1), are not represented by distances with a similar distribution but rather by two clearly distinct classes of distances. In fact, the MDS procedure maps all correlations $r \geq .78$ into almost the same small distance, and all correlations $r < .72$ into almost the same large distance. Even though the resulting step function is perfectly admissible within ordinal MDS, we would probably be reluctant to accept it as a

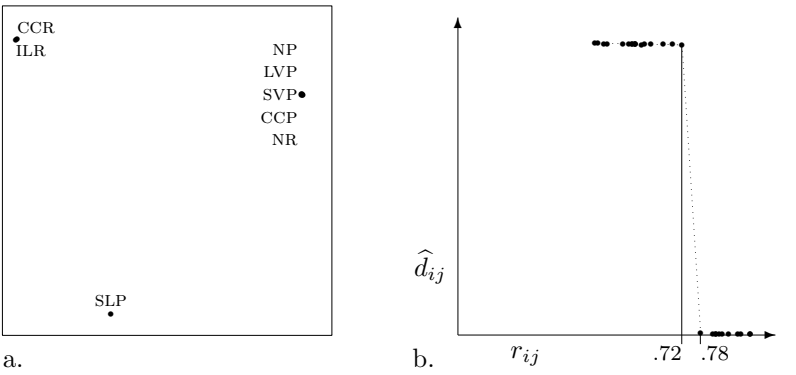


FIGURE 13.1. Ordinal MDS solution (a) and Shepard diagram (b) for correlations in Table 13.1.

sensible transformation of the empirical data, because the transformation simply dichotomizes our data. Using ordinal MDS does not mean that we are indifferent to which monotonic function is chosen as being optimal for the procedure. For our correlations in Table 13.1, it appears reasonable to assume that their differences are also meaningful to some extent, even though their order relations may be more reliable. Hence, we should insist that the correlations be mapped into distances by a more smoothly increasing monotone function. The regression line in the Shepard diagram could then be approximated by a parametric function, for example, a power function or a monotone spline. However, the exact type of the regression function is not known a priori. Otherwise, we would simply choose it and specify a metric MDS model.

On closer analysis, one finds that the solution in Figure 13.1 does not only have an odd transformation function, but it also possesses a peculiar relationship to the data. We can see this as follows. Table 13.1 has been arranged so that the subtests are lumped together into three blocks, where one cluster consists of only one element, SLP. This reveals that: (1) the five subtests in the block {NP, ... , NR} correlate higher with each other than with any subtest in the other blocks, CCR, ILR, or SLP; the lowest *within-block* correlation is r(NR,LVP) = .78, but the highest correlation with any other subtest is r(NR,CCR) = .72; (2) for the block {CCR, ILR}, the within-block correlation is r(CCR, ILR) = .82, which is higher than any of the *between-block* correlations; (3) the same holds trivially for the block {SLP}, where r(SLP, SLP) = 1.00. Because all correlations $r \geq .78$ are mapped into (almost) the same very small distance and all $r < .78$ into (almost) the same much larger distance, the MDS procedure shrinks all within-block distances to almost zero and makes all between-block distances almost equally large. This represents a formal solution to

TABLE 13.2. The rank-order of KIPT subtests in the upper half, the optimal disparities in ordinal MDS in the lower half.

Subtest	NP	LVP	SVP	CCP	NR	SLP	CCR	ILR
NP	-	9	4	1	6	19	10	12
LVP	0	-	1	7	5	21	20	22
SVP	0	0	-	3	2	17	16	23
CCP	0	0	0	-	7	14	11	16
NR	0	0	0	0	-	17	15	18
SLP	1	1	1	1	1	-	13	16
CCR	1	1	1	1	1	1	-	8
ILR	1	1	1	1	1	1	0	-

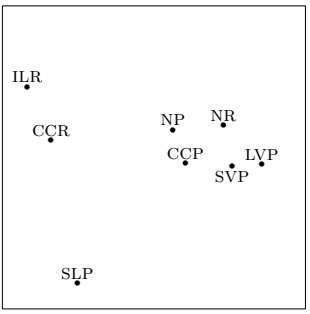
the MDS problem, because it reduces the loss function to a very small value indeed—whether or not the within-block and the between-block distances, respectively, are ordered as the data are! The only aspect of the data that is properly represented, therefore, is that between-block distances are all larger than within-block distances.

It is not difficult to see why Stress is so small in the example above. The lower half of Table 13.2 shows the optimal disparities. One notes that as long as the ranking number in the upper half of the matrix is 9 or smaller, the disparities are all zero, and for rank-order 10 or larger, the disparities are all one. These disparities perfectly match the rank-order information of the data. Ordinal MDS assigns the subtests to three clusters. The within-cluster disparities are zero, so that all points within the cluster have the same coordinates and thus zero distance. Between the cluster points, the distances should be one.

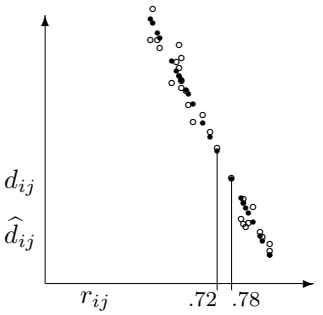
This type of degeneracy can be expected with ordinal MDS when the dimensionality is high compared to the number of objects. It all depends, though, on how many within-blocks of zero exist. In our example, we have three blocks of zero disparities (counting SLP as one cluster). With four within-blocks of zeros, one obtains four clusters for which a perfect solution exists in three dimensions, and so on. The only information that this ordinal MDS solution correctly represents is the partitioning of items in clusters.

13.2 Avoiding Degenerate Solutions

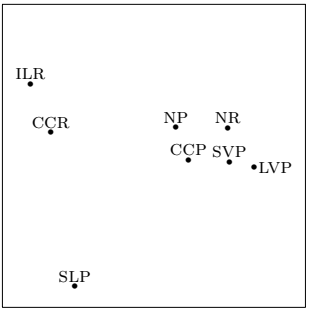
The general solution to degeneracy is to impose stronger restrictions onto the function that maps data into distances. In many instances, a degenerate solution occurs because there are not enough constraints to avoid it. In Table 9.1, we ordered the transformations from strong to weak. Because an ordinal transformation is the weakest possible form of transformation, we can choose any of the stronger transformations as an alternative. We have applied two stronger transformations to the data in Table 13.1, a



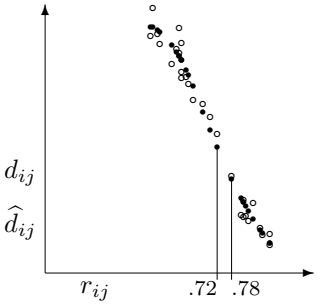
a. linear MDS



b. Shepard plot (linear MDS)



c. MDS with spline



d. Shepard plot (splines)

FIGURE 13.2. Solution of linear MDS with intercept (a) on correlations ($\sigma_n = .0065$) in Table 13.1 and the transformation plot (b), and the solution of MDS with monotonic spline (one interior knot, of order 2, $\sigma_n = .0054$).

linear transformation (with intercept) and a spline transformation (with one interior knot and order 2). The results are in Figure 13.2. Both MDS solutions fit well (interval MDS $\sigma_n = .0065$, monotone spline $\sigma_n = .0054$), and both, of course, map the correlations *smoothly* into distances.

Interval scaling of the data is not the only possibility for arriving at a reasonable MDS configuration when the data possess the peculiar block pattern discussed above. Indeed, any kind of metric representation of the data prevents degenerate solutions. The transformation could also be defined, for example, by $\widehat{d}_{ij} = a + b \cdot \exp(\delta_{ij})$. Depending on the context, such a model may be more attractive a priori, because it specifies a theory about the relation of data and distances that is more precise than to admit just *any* monotone mapping.

13.3 Special Solutions: Almost Equal Dissimilarities

An interesting special case of MDS is concerned with equal dissimilarities. By the *constant dissimilarity case* we mean that $\delta_{ij} = c$, for all i, j, with $c > 0$.[1] We may regard these data as null-data: the differences between all pairs of objects are the same.[2] If we do a ratio MDS on these dissimilarities, the solution has a particular pattern. Consider a simple example of a 3×3 dissimilarity matrix with all dissimilarities equal to 1. An MDS solution with $\sigma_n = 0$ in two dimensions is obtained by placing the points on the corners of an equilateral triangle. It is not hard to extend this result to a solution of a 4×4 constant dissimilarity matrix in three dimensions, where a perfect solution consists of the corner points of a regular tetrahedron (a three-sided pyramid, all sides of equal length). Such a figure is called a *simplex*.[3] The perfect solution for a general $n \times n$ constant dissimilarity matrix is a simplex in $n - 1$ dimensions.

But what happens in lower dimensionality? The optimal MDS solution for constant dissimilarities in one dimension consists of points equally spread on a line. In two dimensions, the points lie on concentric circles (De Leeuw & Stoop, 1984). In three dimensions (or higher), the points lie equally spaced on the surface of a sphere (Buja, Logan, Reeds, & Shepp, 1994). Any permutation of these points gives an equally good fit. Examples of these solutions are shown in Figure 13.3.

For ordinal MDS, we allowed that $p_{ij} \leq p_{kl}$ can be admissibly transformed by a weak monotone function into $\widehat{d}_{ij} = \widehat{d}_{kl}$. Yet, this means that if we choose *all* disparities equal, then the disparities satisfy any rank-order of the proximities, and equal disparities, in turn, ask for an MDS configuration with equal distances. Generally, though, monotone regression should find disparities with a stronger relation to the order of the data (see Section 9.2 and Table 9.4 for an example). However, the equal-disparities scenario can be used to compute a particular upper bound for Stress values in ordinal MDS. Such bounds were determined as follows. We entered a matrix of constant dissimilarities into an MDS program and let the program determine the local minimum Stress. This was done for a range of different ns in one to six dimensions. The Stress values are given in Table 13.3 and can be

---

[1] The dissimilarities do not have to be exactly equal; they may also be approximately equal; that is, $c - \epsilon \leq \delta_{ij} \leq c + \epsilon$ for some small ϵ $(0 \leq \epsilon \leq c)$.

[2] We may regard the constant dissimilarity case as one variety of a *formal* null hypothesis. Another, more common, form of such a null hypothesis is the assumption that the dissimilarities are "random" (see Chapter 3). A *substantively* motivated null hypothesis, in contrast, is derived from the incumbent theory on the domain of interest, whereas the alternative hypothesis relates to the challenging theory.

[3] Note that this simplex (of points) is not equivalent to the simplex of ordered regions discussed in Chapter 5.

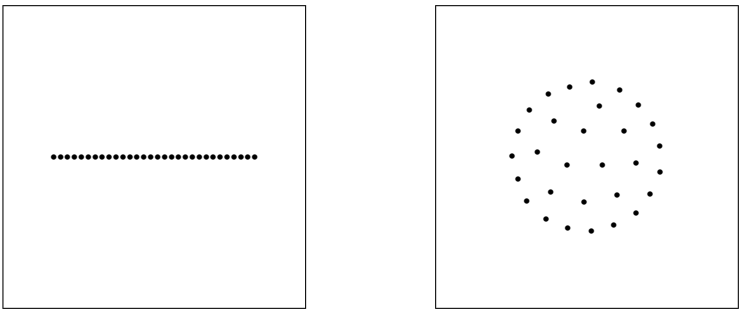


FIGURE 13.3. Solutions for constant dissimilarities with $n = 30$. The left plot shows the unidimensional solution and the right plot a 2D solution.

used as a reference. For example, the Stress found for ordinal MDS on the Morse code data in Chapter 4 was .18 ($n = 36$, 2D). In Table 13.3, we find that for $n = 35$ in 2D the worst expected Stress under the equal-disparity scenario is .3957. Thus, from the Stress value alone, we can safely assume that the 2D solution of the Morse code data shows more structure than the constant dissimilarities case, which was verified by the interpretation.

De Leeuw and Stoop (1984) proved, using theoretical arguments, that for unidimensional scaling Stress could never be larger than $[(n-2)/3n]^{1/2}$, which for large n becomes $1/\sqrt{3} = .5774$. In 2D, they derive the upper bound of Stress by assuming that the points lie equally spaced on a circle (which need not be the optimal solution for constant dissimilarities; see, e.g., the panel on the right-hand side of Figure 13.3). Then, Stress is smaller than $[1 - 2\cot^2(\pi/2n)/(n^2 - n)]^{1/2}$, with the limit $[1 - 8/\pi^2]^{1/2} = .4352$ for large n.

The all-disparities-being-equal degenerate solution seems uncommon in practice. In any case, if it occurs it can be most easily detected by checking the Shepard diagram for numerically highly similar dissimilarities or d-hats. For example, if ratio MDS is used on dissimilarities that fall into the interval [.85, .95] and, thus, have quite similar ratios, a solution is found that is close to the one obtained for constant dissimilarities. Thus, the strong ratio MDS model is not always optimal for showing the data structure. Rather, in such a case we advise redoing the analysis with interval MDS or by using monotone splines. The intercept estimates the constant part of the dissimilarities, and the varying part of the dissimilarities is shown by the MDS configuration. In other words: if the Shepard diagram shows signs of constant dissimilarities or d-hats, the MDS user's strategy should not consist in mechanically choosing a stronger transformation, but rather one that has at least an intercept.

TABLE 13.3. Upper bound values of Stress for ordinal MDS based on MDS of constant dissimilarities.

n	1D	2D	3D	4D	5D	6D
2	.0000	.0000	.0000	.0000	.0000	.0000
3	.3333	.0000	.0000	.0000	.0000	.0000
4	.4083	.1691	.0000	.0000	.0000	.0000
5	.4472	.2598	.1277	.0000	.0000	.0000
6	.4714	.2674	.1513	.1005	.0000	.0000
7	.4880	.2933	.1838	.1265	.0843	.0000
8	.5000	.3084	.2027	.1356	.1091	.0728
9	.5092	.3209	.2145	.1568	.1192	.0949
10	.5164	.3315	.2280	.1688	.1237	.1072
12	.5271	.3473	.2423	.1847	.1473	.1140
14	.5345	.3579	.2555	.1977	.1612	.1334
16	.5401	.3658	.2648	.2069	.1691	.1442
18	.5443	.3719	.2718	.2145	.1780	.1520
20	.5477	.3767	.2777	.2200	.1838	.1572
25	.5538	.3855	.2883	.2311	.1949	.1694
30	.5578	.3914	.2955	.2387	.2022	.1766
35	.5606	.3957	.3007	.2439	.2078	.1822
40	.5628	.3987	.3045	.2480	.2121	.1868
45	.5644	.4012	.3076	.2512	.2154	.1900
50	.5657	.4032	.3100	.2538	.2179	.1926

13.4 Local Minima

MDS algorithms usually end up in a *local minimum*. This property guarantees that any small change of the configuration leads to a higher Stress. In contrast, for a *global minimum* MDS configuration, there is no other configuration with lower Stress. A simplified view of the Stress function is shown in Figure 13.4 for an MDS analysis with two local minima, $\mathbf{X}^*$ and $\mathbf{X}^{**}$, where $\mathbf{X}^{**}$ is a global minimum. The solution found by MDS algorithms is sometimes a global minimum, sometimes only a local minimum.[4] Note that more than one global minimum configuration may exist. Those configurations all have the same global minimum Stress, although the configurations are different (even when the freedom of rotation, translation, and reflection are taken into account). For this reason, we refer to *a* global minimum instead of *the* global minimum.

There are differences between the various MDS algorithms in the effectiveness of locating a global minimum. We limit our discussion of local minima to absolute MDS because for this MDS model the local minimum problem is complicated enough. The local minimum problem can be worse

[4]Local minima in MDS are not necessarily bad. A configuration with a slightly worse fit is acceptable if it has a clearer interpretation than a configuration with a better fit (see also Chapter 10).

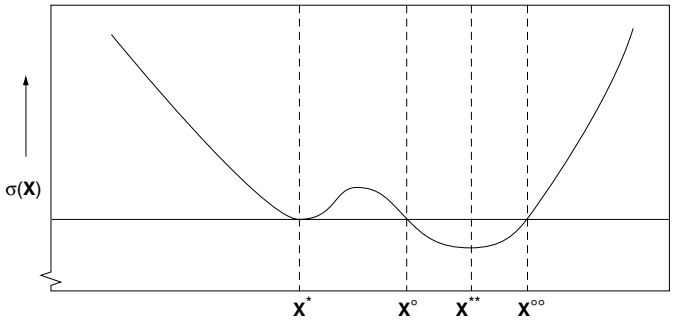


FIGURE 13.4. Example of local minima of a simplified Stress function $\sigma_r(\mathbf{X})$. $\mathbf{X}^*$ is a local minimum, whereas $\mathbf{X}^{**}$ is also a global minimum. $\mathbf{X}^\circ$ and $\mathbf{X}^{\circ\circ}$ have Stress $\sigma_r(\mathbf{X}^*)$.

for nonmetric MDS, or, in the case of nonmetric unidimensional scaling, be less severe.

A simulation study of Groenen and Heiser (1996) showed that local minima are more likely to occur in low dimensionality and hardly occur or are absent in high dimensionality. Below, two special cases are discussed, unidimensional scaling and full-dimensional scaling, for which theoretical results exist concerning local and global optima.

The start configuration of the searching process is of crucial importance for the determination of the final minimum. A random configuration $\mathbf{X}$ is most likely not ideal for finding the lowest-Stress solution by the gradient method, because it does not pay any attention to the data. Therefore, all modern MDS programs use, by default, a *rational* starting configuration derived by some variant of the metric methods discussed in Chapter 12, usually the classical scaling solution of Torgerson (1958) and Gower (1966). Naturally, rationality in the above sense does not guarantee that the starting configuration is best for the particular purpose of an MDS analysis; we may therefore sometimes choose to construct a starting configuration according to given substantive expectations.

Several different methods exist for finding the global minimum. The *method of dimension reduction* repeats the MDS analysis, starting from a high dimensionality (say, 10) and then reducing the dimensionality of the solution space stepwise (down to 2, say). The local minimum configuration of the higher-dimensional analysis is used as a start configuration for the MDS analysis in one dimension lower by dropping the dimension that accounts for the least variance (i.e., the last principal component). Proceeding in this manner, one hopes that the low-dimensional solution is a global minimum.

A different method, called *multiple random starts*, or *multistart*, consists of running the MDS analysis from many (say, 100) different random starting configurations and choosing the one with the lowest Stress. Using multistart

and making some mild assumptions (see Boender, 1984), an estimate for the expected total number of local minima can be given. Let n_s be the number of multistart start configurations and n_m the number of different local minima obtained. Then, the total expected number of local minima n_t is

$$n_t = \frac{n_m(n_s - 1)}{n_s - n_m - 2}. \tag{13.1}$$

If n_s is approximately equal to n_t, then we may assume that all local minima are found. The one with the lowest Stress is the candidate global minimum. Multistart usually gives satisfactory results but is computationally intensive.

Yet another approach is the tunneling method, discussed in Section 13.7. For an overview of other global minimization methods, we refer to Groenen (1993). For a comparison of various global optimization methods on a large empirical data set, see Groenen, Mathar, and Trejos (2000).

13.5 Unidimensional Scaling

It has been noted by De Leeuw and Heiser (1977), Defays (1978), Hubert and Arabie (1986), and Pliner (1996) that minimizing the Stress function with equal weights changes to a combinatorial problem when $m = 1$. It turns out that Stress has many local minima. Therefore, when doing (absolute) MDS in one dimension, one *always* has to be concerned about the local minimum problem. If, however, transformations of the proximities are allowed, then the local minimum problem in unidimensional scaling may be less severe. What follows is a technical discussion of the local minimum problem in unidimensional absolute MDS.

Unidimensional Scaling: A Combinatorial Problem

Inasmuch we are dealing with one dimension, the matrix of coordinates $\mathbf{X}$ has one column and is presented by the $n \times 1$ column vector $\mathbf{x}$ in this section. The distance between two points in one dimension is equal to $d_{ij}(\mathbf{x}) = |x_i - x_j|$. This can be expressed as $d_{ij}(\mathbf{x}) = (x_i - x_j)\text{sign}(x_i - x_j)$, where $\text{sign}(x_i - x_j) = 1$ for $x_i > x_j$, $\text{sign}(x_i - x_j) = 0$ for $x_i = x_j$, and $\text{sign}(x_i - x_j) = -1$ for $x_i < x_j$. An important observation is that only the rank-order of $\mathbf{x}$ determines the $\text{sign}(x_i - x_j)$. In this case, Stress can be expressed as

$$\begin{aligned} \sigma_r(\mathbf{x}) &= \eta_\delta^2 + \eta^2(\mathbf{x}) - 2\rho(\mathbf{x}) \\ &= \sum_{i<j} w_{ij}\delta_{ij}^2 + \sum_{i<j} w_{ij}(x_i - x_j)^2 - 2\sum_{i<j} w_{ij}\delta_{ij}|x_i - x_j| \end{aligned}$$

$$= \eta_\delta^2 + \mathbf{x}'\mathbf{V}\mathbf{x} - 2\sum_{i<j} w_{ij}\delta_{ij}(x_i - x_j)\text{sign}(x_i - x_j). \qquad (13.2)$$

This shows that the cross-product term of Stress, $\rho(\mathbf{x})$, can be factored into a term that is linear in $\mathbf{x}$ and a term that depends only on the rank-order of the elements of $\mathbf{x}$. Therefore, $\rho(\mathbf{x})$ is a piecewise linear function, its pieces being linear within each rank-order of $\mathbf{x}$. For each rank-order, the Stress is consequently quadratic in $\mathbf{x}$. This suggests that the unidimensional scaling problem can be solved by minimizing Stress over all permutations, a *combinatorial* problem. We show that at a local optimum of a function that is only dependent on the rank-order of $\mathbf{x}$, the Guttman transform yields an $\mathbf{x}$ that has the same rank-order. For that rank-order, Stress has a local minimum.

Let ψ denote the rank-order of the vector $\mathbf{x}$, such that $x_{\psi(1)}$ denotes the smallest element of $\mathbf{x}$, and $x_{\psi(i)}$ the element of $\mathbf{x}$ with rank i, so that $x_{\psi(1)} \leq x_{\psi(2)} \leq \cdots \leq x_{\psi(i)} \leq \cdots \leq x_{\psi(n)}$. Let $\mathbf{R}$ be the corresponding permutation matrix, so that $\mathbf{R}\mathbf{x}$ is the vector with the elements ordered nondecreasingly. Define $l_i = \sum_{j<i} w_{\psi(i)\psi(j)}\delta_{\psi(i)\psi(j)}$ and $u_i = \sum_{j>i} w_{\psi(i)\psi(j)}\delta_{\psi(i)\psi(j)}$, which are, respectively, the row sum up to the main diagonal and the row sum from the main diagonal of the matrix with values $w_{\psi(i)\psi(j)}\delta_{\psi(i)\psi(j)}$. Using this notation, (13.2) can be written as

$$\sigma_r(\mathbf{x}) = \eta_\delta^2 + \mathbf{x}'\mathbf{V}\mathbf{x} - 2\mathbf{x}'\mathbf{R}'(\mathbf{l} - \mathbf{u}). \qquad (13.3)$$

For a given rank-order ψ, (13.3) is quadratic in $\mathbf{x}$ and has its minimum when $\mathbf{x}$ is equal to the Guttman transform $\mathbf{V}^+\mathbf{R}'(\mathbf{l}-\mathbf{u})$. The Guttman transform of the majorization approach only uses the rank-order information of the previous configuration, because $\mathbf{R}$, $\mathbf{l}$, and $\mathbf{u}$ only depend on the permutation of $\mathbf{x}$. Therefore, the majorizing algorithm stops if the rank-order of $\mathbf{x}$ does not change, which usually happens in a few iterations. At this point, Stress has a local minimum. Function (13.3) can also be expressed as

$$\sigma_r(\mathbf{x}) = \eta_\delta^2 + \|\mathbf{x} - \mathbf{V}^+\mathbf{R}'(\mathbf{l} - \mathbf{u})\|_{\mathbf{V}}^2 - \|\mathbf{l} - \mathbf{u}\|_{\mathbf{R}\mathbf{V}^+\mathbf{R}'}^2, \qquad (13.4)$$

where the term $t(\psi) = \|\mathbf{l}-\mathbf{u}\|_{\mathbf{R}\mathbf{V}^+\mathbf{R}'}^2$ is a function of the permutation only. Thus, if $t(\psi)$ is maximized, the second term of (13.4) vanishes if $\mathbf{x}$ is chosen equal to the Guttman transform $\mathbf{V}^+\mathbf{R}'(\mathbf{l} - \mathbf{u})$.

Defays (1978) minimizes (13.4) by maximizing $t(\psi)$. Suppose that we have found a permutation ψ that is locally optimal with respect to adjacent pairwise interchanges. That is, any local change of ψ, interchanging $\psi_{(i)}$ and $\psi_{(i+1)}$, does not increase the value of $t(\psi)$. We say that $t(\psi)$ has a local maximum if permutation ψ satisfies this condition. Note that this is a stronger formulation for a local minimum than we used for Stress, because Stress has a local minimum whenever the Guttman transform cannot change the order of $\mathbf{x}$. Groenen (1993) proves that, even for nonconstant w_{ij}, Stress has a local minimum whenever $t(\psi)$ has a local maximum. Suppose that we know how to find a ψ that makes $t(\psi)$ attain the highest

possible value. Then ψ defines the order of $\mathbf{x}$ for a *global* minimum of Stress.

Pliner (1996) gives a $100(1-\alpha)\%$ confidence interval for the number of local minima in unidimensional scaling. Let n_s be the number of (random) sample configurations ψ and n_m be the number of those permutations for which $\sigma_r(\mathbf{x})$ is a local minimum. Then, the confidence interval is given by

$$\left[n! \frac{n_m}{n_m + (n_s - n_m + 1) X_F(2(n_s - n_m + 1), 2n_m)}, \right.$$
$$\left. n! \frac{(n_m + 1) X_F(2(n_m + 1), 2(n_s - n_m))}{(n_s - n_m) + (n_m + 1) X_F(2(n_m + 1), 2(n_s - n_m))} \right],$$

where $X_F(\nu_1, \nu_2)$ is the critical point of an F distribution with (ν_1, ν_2) degrees of freedom such that the probability equals $\alpha/2$ for a similarly distributed t to have t larger or equal to the critical point. Pliner showed that for an 8×8 example (13.5) gave the exact number of local minima of 12770. For another (random data) example, he obtained a 95% confidence interval of $[2.6 \cdot 10^9, 3.4 \cdot 10^9]$ for the number of local minima.

Some Algorithms for Unidimensional Scaling

A whole variety of combinatorial optimization strategies is available for maximizing $t(\psi)$ over ψ. One obvious strategy is simply to try all different orders ψ of n objects, and choose the one for which $t(\psi)$ is maximal. This strategy of *complete search* guarantees a global maximum of $t(\psi)$ and thus a global minimum of Stress. However, because there are $n!$ different permutations, a complete search becomes impractical for $n \geq 10$. Other, more efficient strategies are available. For equal weights, the strategy of *dynamic programming* of Hubert and Golledge (1981) and Hubert and Arabie (1986) is very efficient for moderate n. Their strategy reduces the order of computation from $n!$ to 2^n while still finding a globally optimal solution. Groenen (1993) extended their approach to the case of nonidentical weights but loses the guarantee of reaching a global optimum and some of the computational efficiency. The strategy of *local pairwise interchange* (LOPI) does not guarantee global optimality, but it is very efficient and yields good results. LOPI strategies amount to choosing a pair of objects, interchanging them, and evaluating $t(\psi)$ for the changed rank-order. If $t(\psi)$ is higher than any rank-order we have found so far, then we accept the pairwise interchange. The search is stopped if the pairwise interchanges do not yield a higher $t(\psi)$. The resulting ψ defines a local minimum of Stress. The various implementations of the LOPI strategy result in better local minima of Stress compared to applying the SMACOF algorithm. In a simulation study of Groenen (1993), the LOPI strategies found a global maximum of $t(\psi)$ in the majority of the cases. Poole (1984, 1990) obtained good results in locating the global optimum for unidimensional unfolding. De Soete, Hubert,

and Arabie (1988) found that LOPI performed better than an alternative method called *simulated annealing*. Brusco (2001) studied the use of another implementation of simulated annealing in unidimensional scaling and reported that often a good candidate global minimum was found. Brusco and Stahl (2000) focused on good initial configurations for unidimensional scaling. They proposed to use the results of a related quadratic assignment problem as a start for unidimensional scaling. Their study showed that such an approach can indeed provide effective and efficient initial solutions for large-scale unidimensional scaling problems. A review of unidimensional scaling algorithms minimizing the sum of absolute errors instead of the usual squared errors can be found in Brusco (2002).

Instead of a combinatorial technique, Pliner (1996) used a *smoothing approach* to the local minimum problem in unidimensional scaling. The Stress function is replaced by the function

$$\sigma_\epsilon(\mathbf{X}) = \sum_{i<j} \delta_{ij}^2 + \sum_{i<j} (x_i - x_j)^2 - 2 \sum_{i<j} \delta_{ij} g_\epsilon(x_i - x_j) \text{ with} \tag{13.5}$$

$$g_\epsilon(t) = \begin{cases} t^2(3\epsilon - |t|)/3\epsilon^2 + \epsilon/3, & \text{if } |t| < \epsilon \\ |t|, & \text{if } |t| \geq \epsilon, \end{cases} \tag{13.6}$$

which smooths $-d_{ij}(\mathbf{x})$. The only difference of (13.5) with Stress is that for small distances ($d_{ij} < \epsilon$) the distance in the last term of (13.5) is replaced by a smooth function. Figure 13.5 shows how $g_\epsilon(x_i - x_j)$ smooths $d_{ij}(\mathbf{x}) = |x_i - x_j|$. Pliner recommends starting with the value $\epsilon = 2 \max_{1 \leq i \leq n} n^{-1} \sum_{j=1}^n \delta_{ij}$, minimizing $\sigma_\epsilon(\mathbf{X})$ over $\mathbf{X}$, and using the minimizer as a starting configuration for minimizing $\sigma_\epsilon(\mathbf{X})$ again, but with a smaller value of ϵ. This procedure is repeated until ϵ is very small. If we assume that all distances are greater than 0, then there exists an ϵ for which $\sigma_\epsilon(\mathbf{X})$ reduces to raw Stress. Because $-g_\epsilon(t)$ is a concave function in t, it can be linearly majorized, so that a convergent algorithm can be obtained [as proved by Pliner (1996) using a different argumentation]. More important, the smoothing algorithm turns out to yield global minima solutions very often. Numerical experiments of Pliner suggest that in at least 60% (sometimes even 100%) of the runs, a global minimum was found, which makes this smoothing strategy an important aid for finding the global minimum in unidimensional scaling. Section 13.8 discusses an extension of this smoothing strategy to higher dimensionality.

13.6 Full-Dimensional Scaling

To better understand the local minimum problem for Stress, we consider full-dimensional scaling (absolute MDS), where the dimensionality is $m = n - 1$. In full-dimensional scaling, there is only one minimum, a global one (De Leeuw, 1993). This can be seen as follows. Consider the matrix of

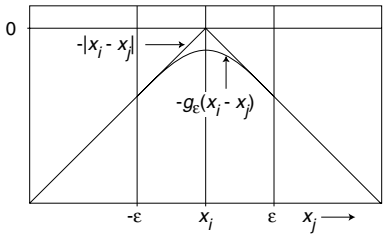


FIGURE 13.5. The function $-d_{ij}(\mathbf{x})$ and the smoothed version $-g_\epsilon(\mathbf{x})$ in (13.5) used by Pliner (1996).

squared distances $\mathbf{D}^{(2)}(\mathbf{X}) = \mathbf{1c}' + \mathbf{c1}' - 2\mathbf{XX}'$, with $\mathbf{X}$ being column centered and where $\mathbf{c}$ contains the diagonal elements of $\mathbf{XX}'$ (see also Section 7.3). Thus, the rank of $\mathbf{XX}'$ can never exceed $n-1$. For $m = n-1$, the cross-product term $\mathbf{XX}'$ is simply a double-centered positive semidefinite (p.s.d.) matrix $\mathbf{B}$, so that the squared distances are equal to $b_{ii} + b_{jj} - 2b_{ij}$. It can be verified that the set of p.s.d. matrices is convex, because for $\mathbf{B}_1, \mathbf{B}_2$ p.s.d. and $0 \leq \alpha \leq 1$, $\alpha\mathbf{B}_1 + (1-\alpha)\mathbf{B}_2$ is p.s.d., too. This allows us to express Stress as

$$\begin{aligned} \sigma_r(\mathbf{B}) &= \sum_{i<j} w_{ij}\delta_{ij}^2 + \sum_{i<j} w_{ij}(b_{ii} + b_{jj} - 2b_{ij}) \\ &\quad -2\sum_{i<j} w_{ij}\delta_{ij}(b_{ii} + b_{jj} - 2b_{ij})^{1/2}. \end{aligned} \tag{13.7}$$

The first term of (13.7) does not depend on $\mathbf{B}$, and the second term is a linear function of $\mathbf{B}$. The third term is minus the square root of the same linear function of $\mathbf{B}$, which is also a convex function in $\mathbf{B}$. It may be verified that the sum of a linear and a convex function is convex, so that $\sigma_r(\mathbf{B})$ is a convex function in $\mathbf{B}$. Thus, minimizing Stress over $\mathbf{B}$ is minimizing a convex function over a convex set, which has a local minimum that is a global minimum. Note that this result does not hold in the case where $\mathbf{B}$ is restricted to have $m < n-1$, because the set of $\mathbf{B}$s restricted to have rank $m < n-1$ is not convex.

Although one would expect $\mathbf{B}$ to be of rank $n-1$ at a minimum, this usually is not the case. In fact, numerical experiments suggest that at a minimum, the rank of $\mathbf{B}$ does not exceed the number of positive eigenvalues in classical scaling. Critchley (1986) and Bailey and Gower (1990) proved this conjecture for S-Stress, but no proof exists for Stress. This result implies that an MDS analysis (with or without transformations) in dimensionality $n-1$ usually ends with a solution of lower rank. De Leeuw and Groenen (1997) prove that at a minimum $\mathbf{B}$ has rank $n-1$ only in the case of a perfect representation of Stress zero with $\mathbf{\Delta}$ a Euclidean distance matrix. The converse is also true: at a minimum with nonzero Stress, $\mathbf{B}$ has rank $n-2$ or smaller.

In confirmatory MDS, the linear constraint $\mathbf{X} = \mathbf{YC}$ is used quite often (see Chapter 10). If, without loss of generality, $\mathbf{Y}$ has $r < n$ columns and is of full rank r, and the dimensionality m of $\mathbf{X}$ equals r, then confirmatory MDS with linear constraints has one minimum, which is global. The same reasoning as above can be used to verify this statement, with the additional constraint that $\mathbf{B} = \mathbf{YCC}'\mathbf{Y}'$, which is also convex if $\mathbf{C}$ is square. In the extreme case where $\mathbf{Y}$ has only one column, $\mathbf{C}$ becomes a scalar, for which the global minimum solution was given in Section 11.1 by b^*.

13.7 The Tunneling Method for Avoiding Local Minima

The problem of local minima is not limited to MDS but is also quite common in numerical optimization. There are many methods for finding a configuration that is not only locally optimal but also has the overall best minimum. One of these methods, called the *tunneling method*, was made suitable for MDS by Groenen and Heiser (1991), Groenen (1993), and Groenen and Heiser (1996). The basic idea of the tunneling method can be described by the following analogy. Suppose that our objective is to find the lowest spot in a mountainous area. First, we try to find the lowest spot in a small area by pouring water and following the water until it forms a small pool. Then, we start drilling a tunnel horizontally. If the tunnel gets out of the mountain, then we are sure that the water flows to a spot that is lower (or remains at the same height). Repeating these steps leads us eventually to the global minimum.

The same idea can be applied for finding the global minimum of the Stress function. Then, the tunneling method alternates over the following two steps.

- Find a local minimum $\mathbf{X}^*$ of the Stress function.
- Find another configuration that has the same Stress as $\mathbf{X}^*$.

The second step is the crux of the method and is called the tunneling step. It is performed by minimizing the *tunneling function* $\tau(\mathbf{X})$. Suppose that the Stress function to be minimized is the one graphed in Figure 13.4. For this Stress function, the tunneling function $\tau(\mathbf{X})$ is shown in Figure 13.6. Near the local minimum $\mathbf{X}^*$, the tunneling function $\tau(\mathbf{X})$ has a *pole* (peak) to avoid finding $\mathbf{X}^*$ as a solution of the tunneling step. Furthermore, $\tau(\mathbf{X})$ becomes zero at $\mathbf{X}^\circ$ and $\mathbf{X}^{\circ\circ}$, which are exactly those points in Figure 13.4 that have the same Stress as $\mathbf{X}^*$. Thus, finding the minimum of $\tau(\mathbf{X})$ gives the solution of the second step of the tunneling method. The precise

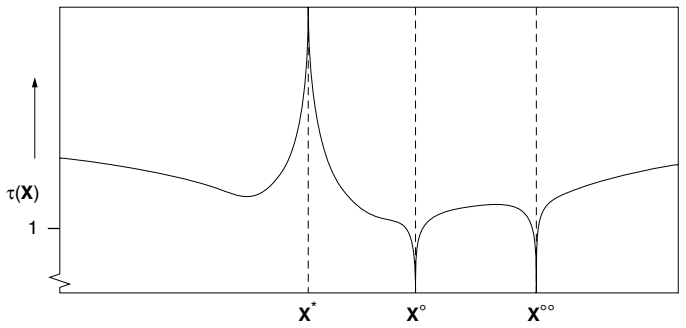


FIGURE 13.6. The tunneling function $\tau(\mathbf{X})$. $\mathbf{X}^*$ is a local minimum of Stress (see also Figure 13.4), $\mathbf{X}^\circ$ and $\mathbf{X}^{\circ\circ}$ are configurations with the same Stress as $\mathbf{X}^*$.

definition of the tunneling function is

$$\tau(\mathbf{X}) = |\sigma_r(\mathbf{X}) - \sigma_r(\mathbf{X}^*)|^\lambda \left(1 + \frac{\omega}{\sum_{i<j} w_{ij}[d_{ij}(\mathbf{X}^*) - d_{ij}(\mathbf{X})]^2}\right). \quad (13.8)$$

Here λ is the *pole strength* parameter that determines how steep the peak is near the local minimum $\mathbf{X}^*$. The *pole width* parameter ω determines the width of activity of the pole. Groenen and Heiser (1996) suggest that $\lambda \leq 1/3$ and $\omega \approx n/2$ are needed to have an effective pole, although the latter seems to depend much on the particular data set.

The effectiveness of the tunneling method is determined by the success of the tunneling step. Clearly, if we start the tunneling step from the global minimum $\mathbf{X}^*$, then $\tau(\mathbf{X})$ cannot become zero (assuming that there is no other global minimum with the same global minimum Stress). Therefore, at some point the tunneling step must be stopped. However, if the tunneling step is stopped too early, then the global minimum can be missed. Experiments of Groenen and Heiser (1996) showed that the tunneling method is able to find the global minimum systematically. However, for some combinations of λ and ω and for certain data sets, the tunneling method fails.

For more details about the tunneling method and the iterative majorization algorithm used for minimizing $\tau(\mathbf{X})$, we refer to Groenen (1993) or Groenen and Heiser (1996). The latter also contains an extension of the tunneling method with Minkowski distances.

13.8 Distance Smoothing for Avoiding Local Minima

In Section 13.5, we discussed the idea of Pliner (1996) to avoid local minima by gradually introducing the rough edges of the Stress function. However,

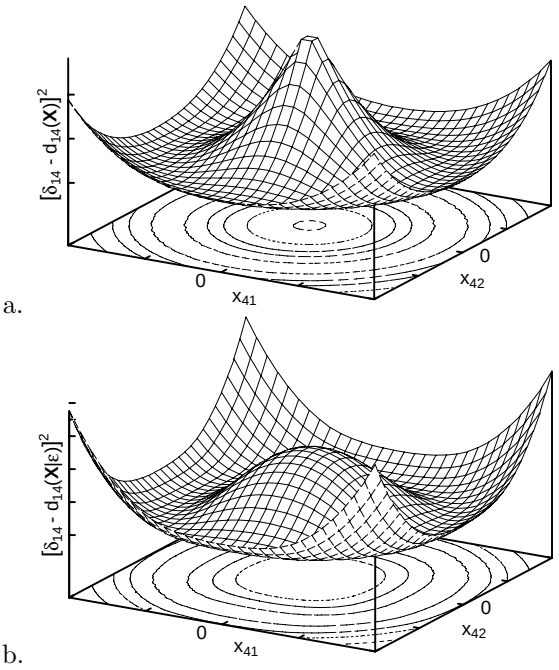


FIGURE 13.7. Surface of the error term $(5 - d_{14}(\mathbf{X}))^2$ in panel (a) and of the corresponding error term $(5 - d_{14}(\mathbf{X}|\epsilon))^2$ in distance smoothing with $\epsilon = 2$.

he only implemented his idea for unidimensional scaling and no algorithm was developed or tested for higher dimensionality. Groenen et al. (1999) continued this line of research by extending this method to more than one dimension. In addition, they also allowed for any Minkowski distance and derived a majorizing algorithm. Their method for avoiding local minima in MDS was called distance smoothing. Here, we explain the basic ideas.

Consider a toy example to visualize the raw Stress function in two dimensions. Suppose that we have $n = 4$ points in 2D, keeping point 1 fixed at $(0, 0)$, point 2 at $(5, 0)$, and point 3 at $(2, -1)$ and leaving the coordinates (x_{41}, x_{42}) for point 4 free, so that

$$\mathbf{X} = \begin{bmatrix} 0 & 0 \\ 5 & 0 \\ 2 & -1 \\ x_{41} & x_{42} \end{bmatrix}.$$

The only relevant dissimilarities are those that involve point 4. Assume that $\delta_{14} = 5, \delta_{24} = 3$, and $\delta_{34} = 2$. Then, minimizing Stress amounts to finding the optimal coordinates x_{41} and x_{42}. For this example, the Stress function can be written as

$$\sigma_r(x_{41}, x_{42}) = (5 - d_{14}(\mathbf{X}))^2 + (3 - d_{24}(\mathbf{X}))^2 + (2 - d_{34}(\mathbf{X}))^2 + c,$$

where $d_{ij}(\mathbf{X})$) are Euclidean distances and c takes all constant terms. The error term $(5 - d_{14}(\mathbf{X}))^2$ is visualized in Figure 13.7a and shows a peak at the origin.

Now, we show what happens if we smooth the peak of the distance. Groenen et al. (1999) do this by using the smoothed distance

$$d_{ij}(\mathbf{X}|\epsilon) = \left(\sum_{s=1}^{p} h_\epsilon^2(x_{is} - x_{js}) \right)^{1/2}, \tag{13.9}$$

where

$$h_\epsilon(t) = \begin{cases} \frac{1}{2}t^2/\epsilon + \frac{1}{2}\epsilon, & \text{if } |t| < \epsilon, \\ |t|, & \text{if } |t| \geq \epsilon, \end{cases} \tag{13.10}$$

Note that $h_\epsilon(t)$ is slightly different from the definition of $g_\epsilon(t)$ in (13.5), but has almost the same form. Now the smoothed Stress becomes

$$\sigma_\epsilon(x_{41}, x_{42}) = (5 - d_{14}(\mathbf{X}|\epsilon))^2 + (3 - d_{24}(\mathbf{X}|\epsilon))^2 + (2 - d_{34}(\mathbf{X}|\epsilon))^2 + c.$$

The effect of distance smoothing on a single error term is shown in Figure 13.7b for $\epsilon = 2$. Clearly, the peak is replaced by a smoothed form. The smoothing is governed by the parameter ϵ: for a large ϵ, there is much smoothing and for ϵ approaching zero no smoothing occurs, so that the error $(5 - d_{14}(\mathbf{X}|\epsilon))^2$ approaches $(5 - d_{14}(\mathbf{X}))^2$.

The effect of the combined error terms for $\sigma_r(x_{41}, x_{42})$ and $\sigma_\epsilon(x_{41}, x_{42})$ with $\epsilon = 2$ and $\epsilon = 5$ are shown in Figure 13.8. The irregularities in the Stress function of Figure 13.8a are caused by the peaks that appear in each of the error terms. Increasing ϵ smooths the irregularity as can be seen in Figures 13.8b and 13.8c. Distance smoothing starts from a large ϵ so that σ_ϵ is very smooth. Then smaller values of ϵ gradually introduce the irregularity. Eventually, for ϵ close to zero, $\sigma_\epsilon(x_{41}, x_{42})$ approaches $\sigma_r(x_{41}, x_{42})$ closely.

The distance smoothing strategy consists of the following steps. Start with a large value of ϵ and minimize σ_ϵ. Then reduce ϵ somewhat and continue minimizing σ_ϵ. Repeat these steps until ϵ is close to zero. Finally, continue minimization σ_r.

Groenen et al. (1999) studied the effectiveness of distance smoothing in comparison to the SMACOF algorithm and KYST. In a simulation study on error-free data using 100 random starts, distance smoothing recovered the true global minimum always for unidimensional scaling and almost always in 2D or 3D. SMACOF and KYST recovered the perfect data only in a small percentage of the random starts. However, for MDS with Minkowski distances close to the dominance distance, distance smoothing did not perform well and KYST yielded the same or better results. Similar results were obtained for error-perturbed data.

To be on the safe side, Groenen et al. (1999) recommend applying the distance smoothing strategy with 10 random starts and choosing the lowest local minimum as the candidate global minimum.

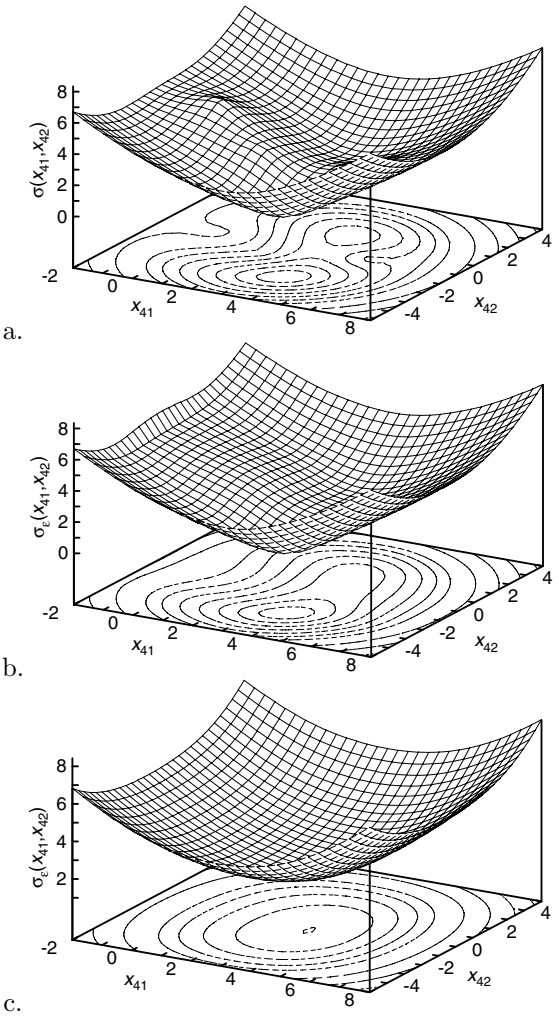


FIGURE 13.8. The surface of the original Stress function $\sigma_r(x_{41}, x_{42})$ panel (a), of the smoothed Stress function $\sigma_\epsilon(x_{41}, x_{42})$ for $\epsilon = 2$ in panel (b) and $\epsilon = 5$ in panel (c).

Note that throughout the discussion on global minima in this chapter, we have assumed ratio MDS. It is not clear how severe the local minimum problem is when we allow for optimal transformations of the d-hats.

13.9 Exercises

Exercise 13.1 Consider the multitrait-multimethod matrix in Exercise 1.6. Do both an ordinal and an interval MDS with these data. Study the Shepard diagrams of both solutions. What would you recommend to a user of MDS, given these findings?

Exercise 13.2 Consider the data matrix in Exercise 1.6.

(a) Use the T- and M-codings of the nine variables to define a starting configuration for MDS and then repeat Exercise 1.6 with this starting configuration. Point 1, thus, gets starting coordinates (1,1); point 2 gets (2,1), and so on.

(b) Study the Shepard diagram of an ordinal MDS and compare it to the Shepard diagram of a linear MDS. Discuss whether these data are better scaled with an ordinal or with an interval MDS (see also Borg & Groenen, 1997; Borg, 1999).

Exercise 13.3 Set up a data matrix (at least 5×5) with $\delta_{ij} = 1$ for all $i \neq j$ and $\delta_{ij} = 0$ for all $i = j$.

(a) Use an interactive MDS program (such as the freeware program PERMAP, see Appendix A) to find a 2D ratio MDS solution for these data.

(b) Click on one point of the solution and move this point to a different position. Then, rerun the MDS analysis with this new starting configuration. Possibly repeat this process, trying to find a different solution from the one obtained above. Compare your results to Figure 13.3.

(c) Find a 1D solution and compare it to Figure 13.3. Test the stability of this solution by the procedure described above. What do you conclude?

(d) Repeat the above analyses with ordinal MDS.

(e) Set up a new data matrix with "nearly equal" but all different dissimilarities ($i \neq j$) from the interval [.85, .95]. Run ratio, interval, and ordinal MDS analyses for these data, using different MDS programs

and forcing the program to do many iterations. Which approach represents the data best not just in terms of Stress, but in terms of describing the structure of the data? Explain why.

Exercise 13.4 Use the data in Table 10.1 on p. 229.

(a) Scale these data with ordinal MDS and compare the solution to the one in Figure 10.3.

(b) Redo the above scaling with two different starting configurations, one that corresponds to Figure 10.2 and one that corresponds to Figure 10.3. Does your MDS program lead to solutions similar to the starting configurations? Can you generate radically different local-minima solutions? How much do they differ in terms of Stress?

(c) Check whether the solutions generated with the different starting configurations remain the same when you force the program to do many (100, say) iterations. (Hint: You may also have to set a very small Stress target value to force your program to actually do that many iterations.)

(d) Use an interactive program (such as PERMAP) and test the stability of the MDS solutions by moving some points and then rerunning MDS from thereon.