8
A Majorization Algorithm for Solving MDS

An elegant algorithm for computing an MDS solution is discussed in this chapter. We reintroduce the Stress function that measures the deviance of the distances between points in a geometric space and their corresponding dissimilarities. Then, we focus on how a function can be minimized. An easy and powerful minimization strategy is the principle of minimizing a function by iterative majorization. An intuitive explanation for iterative majorization in MDS is given using a simplified example. Then, the method is applied in the SMACOF algorithm for minimizing Stress.

8.1 The Stress Function for MDS

We now place the concepts introduced into a common framework to allow the derivation of mathematically justifiable rather than just intuitively plausible methods for solving the MDS construction problem. The methods can then be extended and generalized to MDS models not considered so far. We need the following six basic definitions, most of which have been introduced before.

D1 n denotes the number of empirical objects (stimuli, variables, items, questions, and so on, depending on the context).

D2 If an observation has been made for a pair of objects, i and j, a proximity value p_{ij} is given. If p_{ij} is undefined, we speak of a *missing value*. The term *proximity* is used in a generic way to denote both similarity and dissimilarity values. For similarities, a high p_{ij} indicates that the objects i and j are similar.

D3 A *dissimilarity* is a proximity that indicates how dissimilar two objects are. A small score indicates that the objects are similar, a high score that they are dissimilar. A dissimilarity is denoted by δ_{ij}.

D4 **X** denotes (a) a point configuration (i.e., a set of n points in m-dimensional space) and (b) the $n \times m$ matrix of the coordinates of the n points relative to m Cartesian coordinate axes. A Cartesian coordinate system is a set of pairwise perpendicular straight lines (coordinate axes). All axes intersect at one point, the *origin*, O. The coordinate of a point on axis a is the directed (signed) distance of the point's perpendicular projection onto axis a from the origin. The m-tuple $(x_{i1}, \ldots, x_{im})$ denotes the coordinates of point i with respect to axes $a = 1, \ldots, m$. The origin has the coordinates $(0, \ldots, 0)$.

D5 The Euclidean distance between any two points i and j in **X** is the length of a straight line connecting points i and j in **X**. It is computed by the value resulting from the formula $d_{ij} = [\sum_{a=1}^{m}(x_{ia} - x_{ja})^2]^{1/2}$, where x_{ia} is the coordinate of point i relative to axis a of the Cartesian coordinate system. We also use $d_{ij}(\mathbf{X})$ for the distance to show explicitly that the distance is a function of the coordinates **X**.

D6 The term $f(p_{ij})$ denotes a mapping of p_{ij}, that is, the number assigned to p_{ij} according to rule f. This is sometimes written as $f : p_{ij} \mapsto f(p_{ij})$. We also say that $f(p_{ij})$ is a *transformation* of p_{ij}. (The terms function, transformation, and mapping are synonymous in this context.) Instead of $f(p_{ij})$ we often write $\widehat{d}_{ij}$.

So far, the task of MDS was defined as finding a low-dimensional configuration of points representing objects such that the distance between any two points matches their dissimilarity *as closely as possible*. Of course, we would prefer that each dissimilarity should be mapped exactly into its corresponding distance in the MDS space. But that requires too much, because empirical data always contain some component of error (see, e.g., Section 3.2). We define an error of representation by

$$e_{ij}^2 = (d_{ij} - \delta_{ij})^2. \tag{8.1}$$

Summing (8.1) over i and j yields the total error (of approximation) of an MDS representation,

$$\sigma_r(\mathbf{X}) = \sum_{i=1}^{n} \sum_{j=i+1}^{n} (d_{ij} - \delta_{ij})^2, \text{ for all available } \delta_{ij}, \tag{8.2}$$

which is often written as

$$\sigma_r(\mathbf{X}) = \sum_{i<j} (d_{ij} - \delta_{ij})^2, \text{ for all available } \delta_{ij}. \tag{8.3}$$

The relation $i < j$ in (8.3) simply says that it is sufficient, in general, to sum over half of the data, because dissimilarities and distances are symmetric.

What does "for all available δ_{ij}" mean? In practical research, we sometimes have *missing values*, so that some δ_{ij} are undefined. Missing values impose no restriction on any distances in $\mathbf{X}$. Therefore, we define fixed weights w_{ij} with value 1 if δ_{ij} is known and $w_{ij} = 0$ if δ_{ij} is missing. Other values of w_{ij} are also allowed, as long as $w_{ij} \geq 0$. This defines the final version of *raw Stress* (Kruskal, 1964b),

$$\sigma_r(\mathbf{X}) = \sum_{i<j} w_{ij}(d_{ij}(\mathbf{X}) - \delta_{ij})^2. \tag{8.4}$$

We use the notations σ_r and $\sigma_r(\mathbf{X})$ interchangeably to denote raw Stress.

For every set of coordinates $\mathbf{X}$, a Stress value can be computed. Clearly, we do not want just any $\mathbf{X}$, but we want to find an $\mathbf{X}$ such that the errors (8.1) are small or even zero. Mathematically spoken, we want to minimize $\sigma_r(\mathbf{X})$ over $\mathbf{X}$. For that purpose, we first introduce the concept of differentiating a function, which is explained in the next section.

8.2 Mathematical Excursus: Differentiation

Our aim is to find a coordinate matrix $\mathbf{X}$ such that $\sigma_r(\mathbf{X})$ is minimal. This is a rather complex problem because it requires us to pick $n \cdot m$ coordinates optimally with respect to the Stress function. Therefore, we start by looking at a more simple problem, that is, finding the minimum of a function $f(x)$ with one variable x only. This requires some notions of differential calculus. Consider an example. Let y be the dependent variable and x the independent variable in the function

$$f(x) = y = .3x^4 - 2x^3 + 3x^2 + 5, \tag{8.5}$$

and find the x value for which y attains its smallest value. A first rough estimate of the solution can be derived by looking at some points from the graph of this function, that is, points with the coordinates $(x, f(x))$ in a Cartesian coordinate system. A set of such points can be easily found by choosing some x values, plugging them into the right-hand side of (8.5), and solving for y. If we compute the coordinates of some such points on the graph, we arrive at Figure 8.1 and, with more and more points, at Figure 8.2.

It is clear that point E in Figure 8.2 represents the solution of the minimization problem. For $x = 3.6$ the smallest y value of function (8.5) is obtained: $y = 0.96$. However, point B has, in a sense, the same properties as E, provided we consider a limited interval of x values only, such as only those x values to the left of C. B is called a *local minimum* of the function,

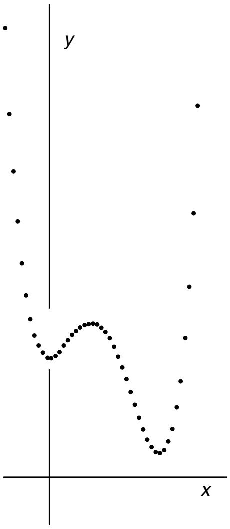


FIGURE 8.1. Some points for $y = 0.3x^4 - 2x^3 + 3x^2 + 5$.

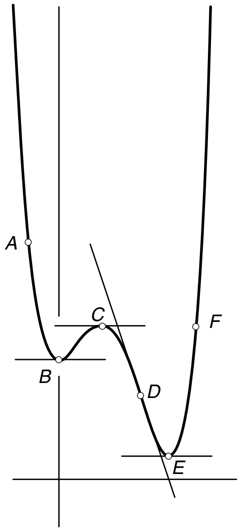


FIGURE 8.2. Graph of $y = 0.3x^4 - 2x^3 + 3x^2 + 5$, with tangent lines at points B, C, D, and E.

and E is the *global minimum*. Analogously, C is a local maximum. Function $f(x)$ has no global maximum.

If we determine the tangents for each point on the graph, it becomes evident that they are horizontal lines at the extrema of the displayed portion of the graph. Figure 8.2 shows this for the minima B and E, and the maximum C. The tangents for other points are not horizontal; that is, their slopes are not zero. This is a property that distinguishes extrema from other points and can be used to find extrema by computation rather than by inspection. If we know all of the extrema, we can select the point with the smallest y-coordinate.

The Slope of a Function

What exactly is a tangent and its slope? Consider Figure 8.3, where the points P and Q are distinguished on the graph for $y = f(x)$. P and Q have the coordinates (x_P, y_P) and (x_Q, y_Q), respectively, or, because $y = f(x)$, $(x_P, f(x_P))$ and $(x_Q, f(x_Q))$, respectively. The straight line through P and Q has the slope

$$\text{slope}(PQ) = \frac{y_Q - y_P}{x_Q - x_P}. \tag{8.6}$$

We now set $x_Q - x_P = \Delta x$. Then (8.6) can be written as

$$\text{slope}(PQ) = \frac{f(x_P + \Delta x) - f(x_P)}{\Delta x}, \tag{8.7}$$

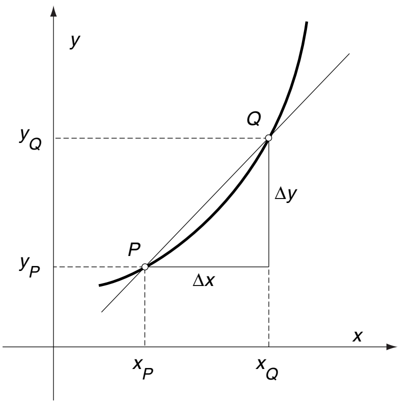


FIGURE 8.3. Some notions for finding tangent line at P.

or, more generally, for any point $P = (x, f(x))$,

$$\text{slope}(PQ) = \frac{f(x+\Delta x) - f(x)}{\Delta x}. \tag{8.8}$$

To find the tangent at point P on the graph, it is necessary to move Q very close to P. However, Q should not become equal to P, because we need two points to uniquely identify the tangent line. This is expressed as follows:

$$\frac{dy}{dx} = \lim_{\Delta x \to 0} \frac{f(x+\Delta x) - f(x)}{\Delta x}, \tag{8.9}$$

where $\lim_{\Delta x \to 0}$ is the *limit operator*. The limit operator makes the difference term Δx in the function $[f(x+\Delta x) - f(x)]/\Delta x$ smaller and smaller, so that Δx approaches 0 without ever reaching it. We say that Δx is made *arbitrarily* or *infinitesimally* small. The symbol dy/dx denotes the resulting *limit* of this operation. Note carefully that the limit dy/dx is *not* generated by setting $\Delta x = 0$, but by approximating $\Delta x = 0$ arbitrarily closely. [Setting $\Delta x = 0$ would turn the right-hand side of (8.9) into 0/0.]

Equations (8.8) and (8.9) are formulated for any point P, not just the particular one in Figure 8.3. Hence, by choosing different Ps, a function of the respective limits is obtained, that is, a function giving the slope of the tangents or the *growth rate* of y relative to x at each point P. This function is called the *derivative* of $y = f(x)$, usually denoted by y'. To illustrate this, let $y = x^2$. The derivative of $y = x^2$ can be found by considering the slope of the tangent at point P:

$$\begin{aligned} \frac{dy}{dx} &= \lim_{\Delta x \to 0} \frac{(x+\Delta x)^2 - (x)^2}{\Delta x} \\ &= \lim_{\Delta x \to 0} \frac{x^2 + (\Delta x)^2 + 2x\Delta x - x^2}{\Delta x} \end{aligned}$$

$$\begin{aligned} &= \lim_{\Delta x \to 0} \left(\frac{(\Delta x)^2}{\Delta x} + \frac{2x\Delta x}{\Delta x} \right) \\ &= \lim_{\Delta x \to 0} (\Delta x + 2x) \\ &= \lim_{\Delta x \to 0} (\Delta x) + \lim_{\Delta x \to 0} (2x) = 2x. \end{aligned} \tag{8.10}$$

Because x is not restricted to a particular point P, we have established a function that gives the slope of $y = x^2$ for any x-value. Hence, $y' = 2x$; that is, the slope of the tangent at each point is simply twice its x-coordinate. For $x = 5$, say, we obtain the slope $dy/dx = 10$, which means that $y = x^2$ grows at this point at the rate of 10 y-units per 1 x-unit (compare Figure 8.3). We can check whether these derivations are correct by setting $x = 5$ and $\Delta x = 3$, say, and then making Δx ever smaller; the smaller Δx gets, the more the limiting value $y' = 10$ is approximated.

Finding the Minimum of a Function

The slope at the minimum must be equal to 0. The derivative gives us an expression for the slope, and thus we can find a minimum by checking all points where the derivative is zero. Points with a zero derivative are called *stationary points*. Given the derivative $y' = 2x$, we can find the minimum of $y = x^2$. We first set $y' = 2x = 0$. But $2x = 0$ only if $x = 0$. So we know that $y = x^2$ has a tangent with slope 0 at $x = 0$. Whether this is a minimum can be checked by looking at the graph of the function. Alternatively, we can compute what the function yields at two[1] neighboring points at $x = 0$. For $x_1 = 1$ and $x_2 = -1$, say, we determine $y_1 = 1^2 = 1$ and $y^2 = (-1)^2 = 1$, respectively, both values greater than the y at $x = 0$, which indicates that we have found a minimum at $x = 0$.

The method of setting the derivative of a function equal to zero and then finding the values that solve this equation has identified only one point. This turned out to be a minimum. We might ask where the maxima are. They can be found by considering the bounds of the interval that x should cover. If we do not restrict x, then these bounds are $-\infty$ and $+\infty$, and this is where the maxima are, as we can see by inserting larger and larger x values into $y = x^2$. Therefore, we also must always test the bounds of the x-interval in which we are interested.

Just as we did in equations (8.10) for the function $y = x^2$, we can find the derivative for any other (continuous and smooth) function. Because *differentiation* (i.e., finding the derivative) is useful in many fields of math-

[1] We test two rather than just one neighboring point at $x = 0$ because the tangent has a zero slope not only at extreme points but also in other cases. Consider, for example, a function that first increases, then runs on a plateau, and then increases again. For all of the points on the plateau, the function has a zero slope. Thus, the zero slope condition for stationarity is *only necessary, but not sufficient*, for identifying an extremum.

TABLE 8.1. Some rules of differentiation.

Rule	Function	Derivative
1	$y = \text{constant} = a$	$dy/dx = 0$
2	$y = x$	$dy/dx = 1$
3	$y = a \cdot x$	$dy/dx = a$
4	$y = a \cdot x^n$	$dy/dx = a \cdot n \cdot x^{n-1}$
5	$y = e^x$	$dy/dx = e^x$
6	$y = \sin(x)$	$dy/dx = \cos(x)$
7	$y = \cos(x)$	$dy/dx = -\sin(x)$
Let $u = f(x)$ and $v = h(x)$ be functions of x. Then:		
8	$y = u + v$	$dy/dx = du/dx + dv/dx$
9	$y = u \cdot v$	$dy/dx = u(dv/dx) + v(du/dx)$
10	$y = u/v$	$dy/dx = [v(du/dx) - u(dv/dx)]/v^2$
Let $y = f(z)$ and $z = g(x)$. Then (*chain rule*):		
11	$y = f(g(x))$	$dy/dx = (dy/dz) \cdot (dz/dx)$

ematics, rules have been derived that greatly simplify finding y'. Some such rules are summarized in Table 8.1. Some of them are patent; others are explained later when we need them. For the example above, $y = x^2$, we find y' by applying rule 4: $y' = dy/dx = 1 \cdot 2 \cdot x^{2-1} = 2x$. For (8.5) we find by rules 1, 4, and 8: $dy/dx = (0.3)(4)x^3 - (2)(3)x^2 + (3)(2)x = 1.2x^3 - 6x^2 + 6x$. Setting this derivative equal to 0 yields the equation $1.2x^3 - 6x^2 + 6x = 0$. After factoring, we have $(x)(1.2x^2 - 6x + 6) = 0$. So, the sought x-values result from the equations $x = 0$ and $1.2x^2 - 6x + 6 = 0$. We find $x_1 = 0$ as one solution, which we identify immediately as a local minimum in the graph in Figure 8.2. The quadratic equation yields $x_2 = 3.618$ and $x_3 = 1.382$ for the other solutions. They correspond to points B and E in the graph.

Second- and Higher-Order Derivatives

The derivative of a function $y = f(x)$ is itself a function of x, $y'' = f'(x)$. One therefore can ask for the derivative of y', $y'' = f''(x)$, the derivative of y'', and so on. The second derivative, y'', indicates the rate of change of the rate of change of $f(x)$. For example, for $y = x^3$ we get $y' = 3x^2$. That is, at any point x, the cubic function grows by the factor $3x^2$. Now, differentiating $y' = 3x^2$ with respect to x (using rule 4 in Table 8.1), we get $y'' = 3.2x$. This means that the rate of change of the growth rate also depends on x: it is 6 times the value of x. So, with large x values, the growth of x^3 "accelerates" quite a bit. As a second example, the rate of change of the growth rate of $y = \sqrt{x} = x^{1/2}, x > 0$, is $y'' = f'(1/2 \cdot x^{-1/2}) = (-1/4) \cdot x^{-3/2} = -1/(4\sqrt{x^3})$. So, y' shows that this function has a positive slope at any point x, and y'' indicates that this slope decreases as x becomes larger. Another way of saying this is that $y = \sqrt{x}$ is concave downwards, whereas $y = x^3$ is convex downwards.

The second derivative is useful to answer the question of whether a stationary point is a minimum or a maximum. Consider Figure 8.2, where we have three stationary points: B, C, and E. C differs from B and E because the speed of growth of $f(x)$ is continuously shrinking when we approach C from the left. To the right of C, the growth rate of $f(x)$ is even negative ("decline"), and becomes more negative as a function of x. The opposite is true for points B and E. This means that if $y'' < 0$ at some stationary point x, then x is a maximum; if $y'' > 0$, x is a minimum. Thus, for the function in Figure 8.2, we have $y'' = 3.6x^2 - 12x + 6$, so that at $x = 0$ (stationary point B) we have $y'' = 6$, for example. Because $6 > 0$, B is a minimum. For $x = 1.382$ (point C), we get -3.708, so that this point is a maximum by the second derivative test.

8.3 Partial Derivatives and Matrix Traces

We often deal with functions that have more than one variable. Such functions are called functions with several variables, multivariable functions, vector functions, or functions with many arguments. An example of such a function is raw Stress, $\sigma_r(\mathbf{X})$. Because we attempt to minimize this function over every single one of its $n \cdot m$ coordinates, we naturally encounter the question of how to find the derivative of multivariable functions. The answer is simple: such functions have as many derivatives as they have arguments, and the derivative for each argument x_i is found by holding all other variables fixed and differentiating the function with respect to x_i as usual. For example, the derivative of the function $f(x, y, z) = x^2y + y^2z + z^2x$ with respect to variable y is $x^2 + 2yz$, using rules 4 and 8 of Table 8.1 and treating the term z^2x as a "constant" (i.e., as not dependent on y). The derivative to one argument of a function of several variables is called the *partial derivative*. The vector of partial derivatives is called the *gradient* vector.

In the following, we focus on one particular multivariable function that becomes important in much of the remainder of this book, the *trace* function, tr $\mathbf{A} = \sum_{i=1}^n a_{ii}$ discussed earlier in Section 7.2 and Table 7.4. The trace can be used to simplify expressing a multiargument linear function such as $f(x_{11}, \ldots, x_{ik}, \ldots, x_{nn}) = \sum_{k=1}^n \sum_{i=1}^n a_{ki}x_{ik}$, where the a_{ik} terms denote constants and x_{ik} are variables:

$$\sum_{k=1}^{n} \sum_{i=1}^{n} a_{ki}x_{ik} = \text{tr } \mathbf{AX} = f(\mathbf{X}).$$

Here, the constants are collected in the matrix $\mathbf{A}$, the variables in $\mathbf{X}$ (see, e.g., Table 8.2 for an example). Suppose that we want to find the partial

TABLE 8.2. Example of differentiating the linear function tr $\mathbf{AX}$ with respect to an unknown matrix $\mathbf{X}$.

(1)	$\mathbf{AX} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{bmatrix}$
(2)	$f(\mathbf{X}) = \text{tr } (\mathbf{AX}) = a_{11}x_{11} + a_{12}x_{21} + a_{21}x_{12} + a_{22}x_{22}$
(3)	$\partial f(\mathbf{X})/\partial \mathbf{X} = (\partial f(\mathbf{X})/\partial x_{ij})$
(4)	$\begin{bmatrix} \partial f(\mathbf{X})/\partial x_{11} = a_{11} & \partial f(\mathbf{X})/\partial x_{12} = a_{21} \\ \partial f(\mathbf{X})/\partial x_{21} = a_{12} & \partial f(\mathbf{X})/\partial x_{22} = a_{22} \end{bmatrix} = \mathbf{A}'$
(5)	rule: $\partial \text{tr } (\mathbf{AX})/\partial \mathbf{X} = \mathbf{A}'$

derivative of the linear function $f(\mathbf{X})$ with respect to the matrix $\mathbf{X}$. The partial derivative of $f(\mathbf{X})$ with respect to $\mathbf{X}$ is the matrix consisting of the derivatives of $f(\mathbf{X})$ with respect to each element of $\mathbf{X}$ (i.e., the matrix with elements $\partial f(\mathbf{X})/\partial x_{ik}$). The notation $\partial f(\mathbf{X})/\partial x_{ik}$ denotes the partial derivative. It replaces $df(\mathbf{X})/dx_{ik}$ used previously in Section 8.2 to make clear that we are dealing with a multivariable function f rather than with a function of just one variable, as in Section 8.2. All variables except x_{ik} are considered constant in $\partial f(\mathbf{X})/\partial x_{ik}$. The matrix of partial derivatives is also denoted by $\nabla f(\mathbf{X})$, by ∇tr $\mathbf{AX}$, or by ∂tr $\mathbf{AX}/\partial\mathbf{X}$.

To find $\nabla f(\mathbf{X})$, we have to take the first derivative of $f(\mathbf{X})$ with respect to every x_{ik} separately. That is, $\partial\text{tr } \mathbf{AX}/\partial x_{ik} = a_{ki}$, so that $\partial\text{tr } \mathbf{AX}/\partial\mathbf{X} = \mathbf{A}'$. The steps needed to find the partial derivative of tr $\mathbf{AX}$ are illustrated in Table 8.2. (For properties of matrix traces, see Table 7.4.) More rules for differentiating a matrix trace function are presented in Table 8.3.

Matrix traces are also useful for expressing a quadratic function such as

$$\sum_{i=1}^{n}\sum_{k=1}^{m} x_{ik}^2 = \text{tr } \mathbf{X}'\mathbf{X}.$$

Because tr $(\mathbf{XX}')$ is equal to $\sum_k \sum_i x_{ki}^2$, tr $\mathbf{X}'\mathbf{X} =$ tr $\mathbf{XX}'$. Hence, the gradient of tr $\mathbf{X}'\mathbf{X}$ is equal to $2\mathbf{X}$ by rule 4, Table 8.3, setting $\mathbf{A} = \mathbf{I}$.

As another example, assume that we want to minimize

$$\begin{aligned} f(\mathbf{X}) &= \text{tr } (\mathbf{X}-\mathbf{Z})'(\mathbf{X}-\mathbf{Z}) \\ &= \sum_{i=1}^{n}\sum_{k=1}^{m}(x_{ik} - z_{ik})^2 \end{aligned}$$

by an appropriate choice of $\mathbf{X}$. We solve this problem formally by first finding the gradient $\nabla f(\mathbf{X})$ and then setting $\nabla f(\mathbf{X}) = \mathbf{0}$ and solving for

TABLE 8.3. Some rules for differentiating a matrix trace with respect to an unknown matrix $\mathbf{X}$; matrix $\mathbf{A}$ is a constant matrix; matrices $\mathbf{U}$, $\mathbf{V}$, $\mathbf{W}$ are functions of $\mathbf{X}$ (Schönemann, 1985).

(1)	$\partial \text{tr } (\mathbf{A})/\partial \mathbf{X} = \mathbf{0}$
(2)	$\partial \text{tr } (\mathbf{AX})/\partial \mathbf{X} = \mathbf{A}' = \partial \text{tr } [(\mathbf{AX})']/\partial \mathbf{X}$
(3)	$\partial \text{tr } (\mathbf{X}'\mathbf{AX})/\partial \mathbf{X} = (\mathbf{A} + \mathbf{A}')\mathbf{X}$
(4)	$\partial \text{tr } (\mathbf{X}'\mathbf{AX})/\partial \mathbf{X} = 2\mathbf{AX}$ if $\mathbf{A}$ is symmetric
(5)	$\partial \text{tr } (\mathbf{U} + \mathbf{V})/\partial \mathbf{X} = \partial \text{tr } (\mathbf{U})/\partial \mathbf{X} + \partial \text{tr } (\mathbf{V})/\partial \mathbf{X}$
(6)	$\partial \text{tr } (\mathbf{UVW})/\partial \mathbf{X} = \partial \text{tr } (\mathbf{WUV})/\partial \mathbf{X} = \partial \text{tr } (\mathbf{VWU})/\partial \mathbf{X}$ Invariance under "cyclic" permutations
(7)	$\partial \text{tr } (\mathbf{UV})/\partial \mathbf{X} = \partial \text{tr } (\mathbf{U}_c\mathbf{V})/\partial \mathbf{X} + \partial \text{tr } (\mathbf{UV}_c)/\partial \mathbf{X}$ Product rule: $\mathbf{U}_c$ and $\mathbf{V}_c$ is taken as a constant matrix when differentiating

$\mathbf{X}$. The gradient can be obtained as follows. If we expand $f(\mathbf{X})$, we get

$$f(\mathbf{X}) = \text{tr } \mathbf{X}'\mathbf{X} + \text{tr } \mathbf{Z}'\mathbf{Z} - 2\text{tr } \mathbf{X}'\mathbf{Z},$$

and, by using the rules from Table 7.4,

$$\begin{aligned} \nabla f(\mathbf{X}) &= \nabla \text{tr } \mathbf{X}'\mathbf{X} + \nabla \text{tr } \mathbf{Z}'\mathbf{Z} - \nabla 2\text{tr } \mathbf{X}'\mathbf{Z} \\ &= 2\mathbf{X} + \mathbf{0} - 2\mathbf{Z} = 2\mathbf{X} - 2\mathbf{Z}. \end{aligned}$$

To find the minimum of $f(\mathbf{X})$, its gradient $\nabla f(\mathbf{X}) = 2\mathbf{X} - 2\mathbf{Z}$ must be equal to $\mathbf{0}$, so that $\mathbf{X} = \mathbf{Z}$ at the minimum.

In the sequel, we often make use of trace minimizations. For the difficult problem of minimizing the Stress function, we need an additional minimization method, iterative majorization, which is explained in the next section.

8.4 Minimizing a Function by Iterative Majorization

For finding the minimum of a function $f(x)$, it is not always enough to compute the derivative $f'(x)$, set it equal to zero, and solve for x. Sometimes the derivative is not defined everywhere, or solving the equation $f'(x) = 0$ is simply impossible. For such cases, we have to refer to other mathematical techniques. A useful method consists of trying to get increasingly better estimates of the minimum. We call such a numerical method an *algorithm*. It consists of a set of computational rules that are usually applied repeatedly, where the previous estimate is used as input for the next cycle of computations which outputs a better estimate. An elegant method is called iterative majorization,[2] which is based on the work of De Leeuw (1977). We first present the main principles of iterative majorization. In the next section, we apply it to the Stress function.

Principles of Majorization

One of the main features of iterative majorization (IM) is that it generates a monotonically nonincreasing sequence of function values. If the function is bounded from below, we usually end up in a stationary point that is a local minimum. An early reference to majorization in the context of line search can be found in Ortega and Rheinboldt (1970, pp. 253–255). Majorization has become increasingly popular as a minimization method; see, for example, Kiers (1990), Bijleveld and De Leeuw (1991), Verboon and Heiser (1992), and Van der Lans (1992). In the field of multidimensional scaling, it has been applied in a variety of settings by, among others, De Leeuw (1977, 1988), De Leeuw and Heiser (1977, 1980), Meulman (1986, 1992), Groenen (1993), Groenen, Mathar, and Heiser (1995), and Groenen, Heiser, and Meulman (1999). Some general papers on iterative majorization are De Leeuw (1994), Heiser (1995), Lange, Hunter, and Yang (2000), Kiers (2002), and Hunter and Lange (2004). Below, we provide an introduction to iterative majorization.

The central idea of the majorization method is to replace iteratively the original complicated function $f(x)$ by an auxiliary function $g(x,z)$, where z in $g(x,z)$ is some fixed value. The function g has to meet the following requirements to call $g(x,z)$ a *majorizing function* of $f(x)$.

- The auxiliary function $g(x,z)$ should be simpler to minimize than $f(x)$. For example, if $g(x,z)$ is a quadratic function in x, then the minimum of $g(x,z)$ over x can be computed in one step (see Section 8.2).
- The original function must always be smaller than or at most equal to the auxiliary function; that is, $f(x) \leq g(x,z)$.
- The auxiliary function should touch the surface at the so-called *supporting point* z; that is, $f(z) = g(z,z)$.

To understand the principle of minimizing a function by majorization, consider the following. Let the minimum of $g(x,z)$ over x be attained at

[2] The term iterative majorization and its abbreviation (IM) was coined by Heiser (1995). Before, the method was called simply majorization. In MDS the method goes back to the work of De Leeuw (1977).

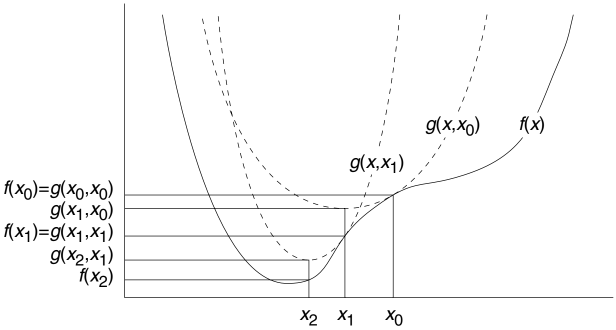


FIGURE 8.4. Illustration of two iterations of the iterative majorization method. The first iteration starts by finding the auxiliary function $g(x, x_0)$, which is located above the original function $f(x)$ and touches at the supporting point x_0. The minimum of the auxiliary function $g(x, x_0)$ is attained at x_1, where $f(x_1)$ can never be larger than $g(x_1, x_0)$. This completes one iteration. The second iteration is analogous to the first iteration.

x^*. The last two requirements of the majorizing function imply the chain of inequalities

$$f(x^*) \leq g(x^*, z) \leq g(z, z) = f(z). \tag{8.11}$$

This chain of inequalities is named the *sandwich* inequality by De Leeuw (1993), because the minimum of the majorizing function $g(x^*, z)$ is squeezed between $f(x^*)$ and $f(z)$. A graphical representation of these inequalities is presented in Figure 8.4 for two subsequent iterations of iterative majorization of the function $f(x)$. The iterative majorization algorithm is given by

1. Set $z = z_0$, where z_0 is a starting value.
2. Find update x^u for which $g(x^u, z) \leq g(z, z)$.
3. If $f(z) - f(x^u) < \varepsilon$, then stop. ($\varepsilon$ is a small positive constant.)
4. Set $z = x^u$ and go to 2.

Obviously, by (8.11) the majorization algorithm yields a nonincreasing sequence of function values, which is an attractive aspect of iterative majorization. If the function $f(x)$ is not bounded from below, and if there are no sufficient restrictions on x, then the stop criterion in step 3 may never be met. In the sequel, this situation does not arise. Although the function value never increases, the majorization principle does not say how fast the function values converge to a minimum. In most applications, an algorithm based on iterative majorization is not very fast. As shown in Section 8.2, a necessary condition for a minimum at point x^* is that the derivative of $f(x)$ at x^* is 0. Using the inequalities of (8.11), this also implies that x^*

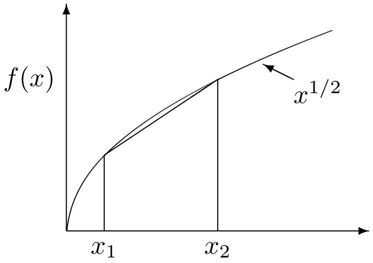


FIGURE 8.5. Graph of the concave function $x^{1/2}$.

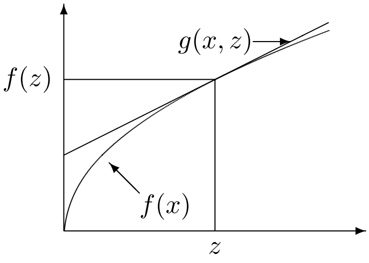


FIGURE 8.6. An example of linear majorization of the concave function $f(x) = x^{1/2}$ by the linear majorizing function $g(x, z)$.

minimizes $g(x, x^*)$ over x, with $g(x^*, x^*)$ as the minimum. Thus, the necessary condition of a zero derivative at a local minimum may be replaced by the weaker condition that $g(x^u, y) = f(y)$ and $x^u = y$. In general, the majorization algorithm can stop at any stationary point, not necessarily at a local minimum. However, Fletcher (1987) notes that, for algorithms that reduce the function value on every iteration, it usually holds that "the stationary point turns out to be a local minimizer, except in rather rare circumstances" (p. 19).

Linear and Quadratic Majorization

We distinguish two particularly useful classes of majorization: linear and quadratic (De Leeuw, 1993). The first one is majorization of a function that is *concave.* A concave function $f(x)$ is characterized by the inequality $f(\alpha x + (1-\alpha)z) \geq \alpha f(x) + (1-\alpha)f(z)$ for $0 \leq \alpha \leq 1$. Thus, the line that connects the function values at $f(x)$ and $f(z)$ remains below the graph of a concave function. An example of the concave function $f(x) = x^{1/2}$ is given in Figure 8.5. But for such a function $f(x)$, it is always possible to have a straight line defined by $g(x, z) = ax + b$ (with a and b dependent on z) such that $g(x, z)$ touches the function $f(x)$ at $x = z$, and elsewhere the line defined by $g(x, z)$ is above the graph of $f(x)$. Clearly, $g(x, z) = ax + b$ is a linear function in x. Therefore, we call this type of majorization *linear majorization.* Any concave function $f(x)$ can be majorized by a linear function $g(x, z)$ at any point z. Thus, $g(x, z)$ satisfies all three requirements of a majorizing function. An example of a linear majorizing function $g(x, z)$ with supporting point z of the concave function $f(x) = x^{1/2}$ is given in Figure 8.6.

The second class of functions that can be easily majorized is characterized by a bounded second derivative. For a function $f(x)$ with a bounded second derivative, there exists a quadratic function that has, compared to $f(x)$, a larger second derivative at any point x. This means that $f(x)$ does not have

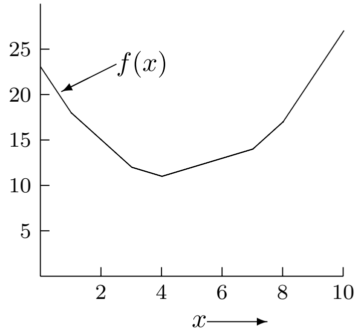


FIGURE 8.7. Graph of the function $f(x) = |x-1| + |x-3| + |x-4| + |x-7| + |x-8|$.

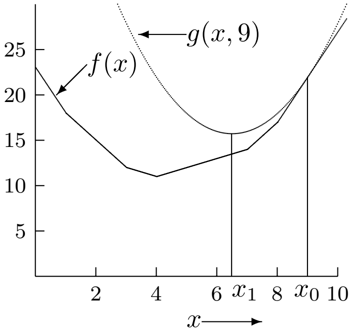


FIGURE 8.8. A quadratic majorizing function $g(x, x_0)$ of $f(x)$ with supporting point $x_0 = 9$.

very steep parts, because there always exists a quadratic function that is steeper. This type of majorization can be applied if the function $f(x)$ can be majorized by $g(x,z) = a(z)x^2 - b(z)x + c(z)$, with $a(z) > 0$, and $a(z)$, $b(z)$, and $c(z)$ functions of z, but not of x. We call this type of majorization *quadratic majorization.*

Example: Majorizing the Median

Heiser (1995) gives an illustrative example of iterative majorization for computing the *median.* The median of the numbers $x_1, x_2, \ldots, x_n$ is the number for which $f(x) = \sum_{i=1}^n |x - x_i|$ is a minimum. For example, the median of the numbers $x_1 = 1$, $x_2 = 3$, $x_3 = 4$, $x_4 = 7$, and $x_5 = 8$ is 4. Thus, the median is the value for which 50% of all observations is smaller. The function $f(x)$ is shown in Figure 8.7.

How can we majorize $f(x)$? We begin by noting that $g(x,z) = |z|/2 + x^2/|2z|$ majorizes $|x|$ (Heiser, 1988a). The three majorization requirements are fulfilled by this $g(x,z)$. First, $g(x,z)$ is a simple function because it is quadratic in x. Second, we have $f(x) \leq g(x,z)$ for all x and fixed z. This can be seen by using the inequality $(|x| - |z|)^2 \geq 0$, which always holds, because squares are always nonnegative. Developing this inequality gives

$$\begin{aligned} x^2 + z^2 - 2|x||z| &\geq 0 \\ 2|x||z| &\leq x^2 + z^2 \\ |x| &\leq \frac{1}{2}\frac{x^2}{|z|} + \frac{1}{2}|z|, \end{aligned} \tag{8.12}$$

which proves $|x| \leq g(x,z)$. The third requirement of a majorizing function is that there must be equality in the supporting point; that is, $f(z) = g(z,z)$.

If we substitute $x = z$ in (8.12), we obtain

$$\frac{1}{2}\frac{z^2}{|z|} + \frac{1}{2}|z| = \frac{1}{2}|z| + \frac{1}{2}|z| = |z|,$$

which shows that all three requirements for a majorizing function hold.

$f(x)$ is majorized by replacing x and z in (8.12) by the separate terms in $f(x)$. This means that $|x-1|$ is majorized by $g_1(x,z) \leq |z-1|/2 + (x-1)^2/|2(z-1)|$. Similarly, the second term $|x-3|$ of $f(x)$ is majorized by $g_2(x,z) \leq |z-3|/2 + (x-3)^2/|2(z-3)|$, and so on. Summing the majorization functions for each term in $f(x)$ yields the majorizing function of $f(x)$; that is,

$$\begin{aligned} g(x,z) &= g_1(x,z) + g_2(x,z) + g_3(x,z) + g_4(x,z) + g_5(x,z) \\ &= \frac{1}{2}|z-1| + \frac{(x-1)^2}{|2(z-1)|} + \frac{1}{2}|z-3| + \frac{(x-3)^2}{|2(z-3)|} \\ &\quad + \frac{1}{2}|z-4| + \frac{(x-4)^2}{|2(z-4)|} + \frac{1}{2}|z-7| + \frac{(x-7)^2}{|2(z-7)|} \\ &\quad + \frac{1}{2}|z-8| + \frac{(x-8)^2}{|2(z-8)|}. \end{aligned} \tag{8.13}$$

To start the iterative majorization algorithm, choose the initial value to be $x_0 = 9$, although any other value would be equally valid. This implies that the first supporting point x_0 in the IM algorithm is $z = x_0 = 9$. After substitution of $z = 9$ into (8.13) and simplification, we obtain

$$\begin{aligned} g(x,9) &= \frac{1}{2}|9-1| + \frac{(x-1)^2}{|2(9-1)|} + \frac{1}{2}|9-3| + \frac{(x-3)^2}{|2(9-3)|} + \frac{1}{2}|9-4| \\ &\quad + \frac{(x-4)^2}{|2(9-4)|} + \frac{1}{2}|9-7| + \frac{(x-7)^2}{|2(9-7)|} + \frac{1}{2}|9-8| + \frac{(x-8)^2}{|2(9-8)|} \\ &= \frac{8}{2} + \frac{(x-1)^2}{16} + \frac{6}{2} + \frac{(x-3)^2}{12} + \frac{5}{2} \\ &\quad + \frac{(x-4)^2}{10} + \frac{2}{2} + \frac{(x-7)^2}{4} + \frac{1}{2} + \frac{(x-8)^2}{2} \\ &= \frac{8}{2} + \frac{(x-1)^2}{16} + \frac{6}{2} + \frac{(x-3)^2}{12} + \frac{5}{2} \\ &\quad + \frac{(x-4)^2}{10} + \frac{2}{2} + \frac{(x-7)^2}{4} + \frac{1}{2} + \frac{(x-8)^2}{2} \\ &= \frac{239}{240}x^2 - \frac{517}{40}x + \frac{4613}{80}. \end{aligned} \tag{8.14}$$

This example of quadratic majorization is illustrated in Figure 8.8. Because $g(x, x_0)$ is quadratic in x, its minimum can be easily obtained by setting the derivative equal to zero (see Section 8.2). The minimum of $g(x, x_0)$ is attained at $x_1 \approx 6.49$. Due to the majorization inequality, we must have that $f(x_1) \leq g(x_1, x_0) \leq g(x_0, x_0) = f(x_0)$. Thus, we have found an x_1 with a lower function value $f(x)$. The next step in the majorization algorithm is to declare x_1 to be the next supporting point, to compute $g(x, x_1)$ and find its minimum x_2, and so on. After some iterations, we find that 4 is the minimum for $f(x)$; hence 4 is the median.

The key problem in quadratic majorization is to find a majorizing inequality such as (8.12). Unlike concave functions, which can always be linearly majorized, it is an art to find quadratic majorizing functions. Note that linear and quadratic majorization can be combined without any problem as long as the majorization conditions hold.

8.5 Visualizing the Majorization Algorithm for MDS

To get an idea what the iterative majorization algorithm does in MDS, we consider a mini example from the data of Exercise 3.3. These data contain the correlations among the returns of 13 stock markets. To analyze these data, we converted the correlations into dissimilarities by (6.1), so that $\delta_{ij} = (2-2r_{ij})^{1/2}$. Then we performed ratio MDS by the Smacof algorithm (see the next section). The resulting configuration is given in Figure 8.9. We see, for example, that the Dow Jones (dj) and Standard & Poors (sp) indices correlate highly, because they are very close together. We also see that the European indices (brus, dax, vec, cbs, ftse, milan, and madrid) are reasonably similar because they are located together. The Asian markets (hs, nikkei, taiwan, and sing) do not seem to correlate highly among one another as they are lying at quite some distance from one another.

To see how the iterative majorization algorithm for MDS works, consider the situation where the coordinates of all stock indices are kept fixed at the positions of Figure 8.9 except for the point nikkei. To minimize raw Stress, we can only vary the two coordinates x_{i1} and x_{i2} of nikkei. This simplification allows us to visualize the raw Stress function as a surface in 3D with x_{i1} and x_{i2} in the xy plane and the raw Stress value on the z-axis. Figure 8.10 shows the raw Stress surface in both panels. The ground area shows the position of all the fixed points and, for reference, also the optimal position of nikkei. It is clear that in this situation, the coordinates for nikkei where raw Stress finds its global minimum are indeed located at the point with label nikkei. However, a computer is "blind" and cannot "see" where these optimal coordinates of nikkei with the lowest raw Stress function is found. Therefore, it needs an optimization algorithm such as iterative majorization to compute the location of minimal raw Stress.

Iterative majorization for MDS works in this example as follows. Suppose that the initial guess for the coordinates of nikkei is the origin. Then,

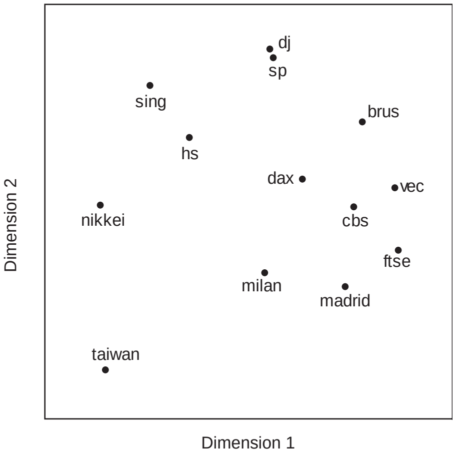


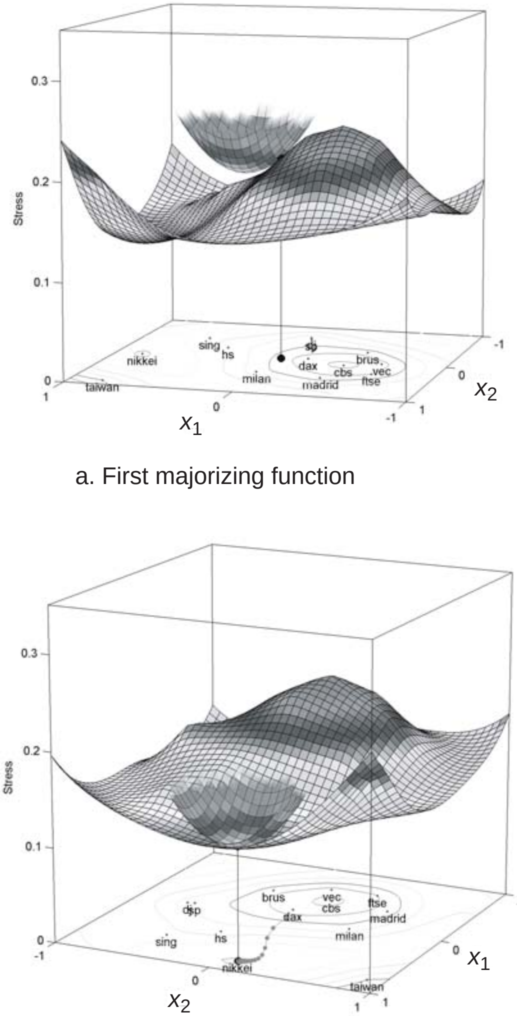
FIGURE 8.9. Ratio MDS solution of correlations between returns of 13 stock markets. The data are given in Exercise 3.3.

the majorizing function must touch the raw Stress function at the origin (with coordinates $x_{i1} = 0$ and $x_{i2} = 0$) and must be located above it (or touch it) at other locations. The parabola in Figure 8.10a satisfies these restrictions and is therefore a valid majorizing function. At the location of the minimum of this majorizing function, the raw Stress function is lower. Thus, choosing this location as the next estimate of the coordinates for nikkei reduces the raw Stress. At this location, a new majorizing function can be found that again touches the raw Stress function at this location and is otherwise located above the raw Stress function. The minimum of this new majorizing function can be determined and will again decrease raw Stress. This process is iterated until the improvement in raw Stress is considered small enough. This final situation is shown in Figure 8.10b with the last majorizing function. We note that the majorizing algorithm has correctly identified the best local minimum possible. The estimates for the location of point nikkei in the different iterations is shown by the trail of points in the xy plane between the origin and the final location of nikkei.

Here, we focused on the special case that only two coordinates need to be estimated and all others are kept fixed. The next section explains how the iterative majorization algorithm works when all coordinates need to be found simultaneously.

8.6 Majorizing Stress

So far, we have discussed the principle of iterative majorization for functions of one variable x only. The same idea can be applied to functions that



a. First majorizing function

b. Final majorizing function

FIGURE 8.10. Visualization of the raw Stress function for the Stock market data where all coordinates are kept fixed except those of nikkei. For reference, the optimal position of nikkei is also shown. The upper panel shows the majorizing function with the origin as current estimate for the location of nikkei. The lower panel shows the final majorizing function and a trail of points in the xy-plane showing the positions of point nikkei in the different iterations.

have several variables. As long as the majorizing inequalities (8.11) hold, iterative majorization can be used to minimize a function of many variables.

We now apply iterative majorization to the Stress function, which goes back to De Leeuw (1977), De Leeuw and Heiser (1977), and De Leeuw (1988). The acronym SMACOF initially stood for "Scaling by Maximizing a Convex Function," but since the mid-1980s it has stood for "Scaling by Majorizing a Complicated Function." Algorithms other than SMACOF have been derived to minimize Stress. For example, using approaches from convex analysis, the same algorithm for minimizing Stress was obtained by De Leeuw (1977), Mathar (1989), and Mathar and Groenen (1991). Earlier, Stress was minimized by steepest descent algorithms by Kruskal (1964b) and Guttman (1968) that use the gradient of Stress. However, the SMACOF theory is simple and more powerful, because it guarantees monotone convergence of Stress. Hence, we pursue the majorization approach and show how to majorize the raw Stress function, $\sigma_r(\mathbf{X})$, following the SMACOF theory.

Components of the Stress Function

The Stress function (8.4) can be written as

$$\begin{aligned}
\sigma_r(\mathbf{X}) &= \sum_{i<j} w_{ij} \left(\delta_{ij} - d_{ij}(\mathbf{X})\right)^2 \\
&= \sum_{i<j} w_{ij}\delta_{ij}^2 + \sum_{i<j} w_{ij} d_{ij}^2(\mathbf{X}) - 2\sum_{i<j} w_{ij}\delta_{ij} d_{ij}(\mathbf{X}) \\
&= \eta_\delta^2 + \eta^2(\mathbf{X}) - 2\rho(\mathbf{X}), \qquad (8.15)
\end{aligned}$$

where $d_{ij}(\mathbf{X})$ is the Euclidean distance between points i and j; see also (3.3). From (8.15) we see that Stress can be decomposed into three parts. The first part, η_δ^2, is only dependent on the fixed weights w_{ij} and the fixed dissimilarities δ_{ij}, and not dependent on $\mathbf{X}$; so η_δ^2 is constant. The second part, $\eta^2(\mathbf{X})$, is a weighted sum of the squared distances $d_{ij}^2(\mathbf{X})$. The final part, $-2\rho(\mathbf{X})$, is a weighted sum of the "plain" distances $d_{ij}(\mathbf{X})$. Before we go on, we have to make one additional assumption: we assume throughout this book that the weight matrix $\mathbf{W}$ is *irreducible*, that is, there exists no partitioning of objects into disjoint subsets, such that $w_{ij} = 0$ whenever objects i and j are in different subsets. If the weight matrix is reducible, then the problem can be decomposed into separate smaller multidimensional scaling problems, one for each subset. Let us consider $\eta^2(\mathbf{X})$ and $\rho(\mathbf{X})$ separately to obtain our majorization algorithm.

A Compact Expression for the Sum of Squared Distances

We first look at $\eta^2(\mathbf{X})$, which is a sum of the squared distances. For the moment, we consider only one squared distance $d_{ij}^2(\mathbf{X})$. Let $\mathbf{x}_a$ be column

a of the coordinate matrix $\mathbf{X}$. Furthermore, let $\mathbf{e}_i$ be the ith column of the identity matrix $\mathbf{I}$. Thus, if $n = 4$, $i = 1$, and $j = 3$, then $\mathbf{e}_i' = [1\ 0\ 0\ 0]$ and $\mathbf{e}_j' = [0\ 0\ 1\ 0]$, so that $(\mathbf{e}_i - \mathbf{e}_j)' = [1\ 0\ {-1}\ 0]$. But this means that $x_{ia} - x_{ja} = (\mathbf{e}_i - \mathbf{e}_j)'\mathbf{x}_a$, which allows us to express the squared distance $d_{13}^2(\mathbf{X})$ as

$$\begin{aligned} d_{13}^2(\mathbf{X}) &= \sum_{a=1}^m \mathbf{x}_a'(\mathbf{e}_1 - \mathbf{e}_3)(\mathbf{e}_1 - \mathbf{e}_3)'\mathbf{x}_a \\ &= \sum_{a=1}^m \mathbf{x}_a' \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{x}_a = \sum_{a=1}^m \mathbf{x}_a'\mathbf{A}_{13}\mathbf{x}_a \\ &= \text{tr } \mathbf{X}'\mathbf{A}_{13}\mathbf{X}. \end{aligned} \tag{8.16}$$

The matrix $\mathbf{A}_{ij}$ is simply a matrix with $a_{ii} = a_{jj} = 1$, $a_{ij} = a_{ji} = -1$, and all other elements zero. Note that $\mathbf{A}_{ij}$ is row and column centered, so that $\mathbf{A}_{ij}\mathbf{1} = \mathbf{0}$ and $\mathbf{1}'\mathbf{A}_{ij} = \mathbf{0}'$. But $\eta^2(\mathbf{X})$ is a weighted sum of these squared distances. One term of $\eta^2(\mathbf{X})$ is

$$\begin{aligned} w_{ij}d_{ij}^2(\mathbf{X}) &= w_{ij}\text{tr } \mathbf{X}'\mathbf{A}_{ij}\mathbf{X} \\ &= \text{tr } \mathbf{X}'(w_{ij}\mathbf{A}_{ij})\mathbf{X}, \end{aligned}$$

and summing over all $i < j$ terms gives

$$\begin{aligned} \eta^2(\mathbf{X}) &= \sum_{i<j} w_{ij}d_{ij}^2(\mathbf{X}) = \text{tr } \mathbf{X}'\left(\sum_{i<j} w_{ij}\mathbf{A}_{ij}\right)\mathbf{X} \\ &= \text{tr } \mathbf{X}'\mathbf{V}\mathbf{X}. \end{aligned} \tag{8.17}$$

In a 3 × 3 example, the matrix $\mathbf{V}$ defined in (8.17) becomes

$$\begin{aligned} \mathbf{V} &= \sum_{i<j} w_{ij}\mathbf{A}_{ij} \\ &= \begin{bmatrix} w_{12} + w_{13} & -w_{12} & -w_{13} \\ -w_{12} & w_{12} + w_{23} & -w_{23} \\ -w_{13} & -w_{23} & w_{13} + w_{23} \end{bmatrix}, \end{aligned} \tag{8.18}$$

or, in general, $v_{ij} = -w_{ij}$ if $i \neq j$ and $v_{ii} = \sum_{j=1, j\neq i}^n w_{ij}$ for the diagonal elements of $\mathbf{V}$. By (8.17) we have obtained a compact matrix expression for $\eta^2(\mathbf{X})$. Furthermore, $\eta^2(\mathbf{X})$ is a quadratic function in $\mathbf{X}$, which is easy to handle. Because $\mathbf{V}$ is the weighted sum of row and column centered matrices $\mathbf{A}_{ij}$, it is row and column centered itself, too. Because of our assumption that the weights are irreducible, the rank of $\mathbf{V}$ is $n - 1$, the zero eigenvalue corresponding to the eigenvector $n^{-1/2}\mathbf{1}$.

Majorizing Minus a Weighted Sum of Distances

We now switch to $-\rho(\mathbf{X})$, which is minus a weighted sum of the distances; that is,

$$-\rho(\mathbf{X}) = -\sum_{i<j}(w_{ij}\delta_{ij})d_{ij}(\mathbf{X}).$$

For the moment, we focus on minus the distance. To obtain a majorizing inequality for $-d_{ij}(\mathbf{X})$, we use the *Cauchy–Schwarz* inequality,

$$\sum_{a=1}^{m} p_a q_a \leq \left(\sum_{a=1}^{m} p_a^2\right)^{1/2} \left(\sum_{a=1}^{m} q_a^2\right)^{1/2}. \tag{8.19}$$

Equality of (8.19) occurs if $q_a = cp_a$. If we substitute p_a by $(x_{ia} - x_{ja})$ and q_a by $(z_{ia} - z_{ja})$ in (8.19), we obtain

$$\begin{aligned}\sum_{a=1}^{m}(x_{ia} - x_{ja})(z_{ia} - z_{ja}) &\leq \left(\sum_{a=1}^{m}(x_{ia} - x_{ja})^2\right)^{1/2} \left(\sum_{a=1}^{m}(z_{ia} - z_{ja})^2\right)^{1/2} \\ &= d_{ij}(\mathbf{X})d_{ij}(\mathbf{Z}), \qquad (8.20)\end{aligned}$$

with equality if $\mathbf{Z} = \mathbf{X}$. Dividing both sides by $d_{ij}(\mathbf{Z})$ and multiplying by -1 gives

$$-d_{ij}(\mathbf{X}) \leq -\frac{\sum_{a=1}^{m}(x_{ia} - x_{ja})(z_{ia} - z_{ja})}{d_{ij}(\mathbf{Z})}. \tag{8.21}$$

If points i and j have zero distance in configuration matrix $\mathbf{Z}$, then (8.21) becomes undefined, but because of the positivity of $d_{ij}(\mathbf{X})$ it is still true that $-d_{ij}(\mathbf{X}) \leq 0$. Proceeding as in (8.16)–(8.18), a simple matrix expression is obtained:

$$\sum_{a=1}^{m}(x_{ia} - x_{ja})(z_{ia} - z_{ja}) = \text{tr } \mathbf{X}'\mathbf{A}_{ij}\mathbf{Z}. \tag{8.22}$$

Combining (8.21) and (8.22), multiplying by $w_{ij}\delta_{ij}$, and summing over $i < j$ gives

$$\begin{aligned}-\rho(\mathbf{X}) &= -\sum_{i<j}(w_{ij}\delta_{ij})d_{ij}(\mathbf{X}) \\ &\leq -\text{tr } \mathbf{X}'\left(\sum_{i<j} b_{ij}\mathbf{A}_{ij}\right)\mathbf{Z} \\ &= -\text{tr } \mathbf{X}'\mathbf{B}(\mathbf{Z})\mathbf{Z}, \qquad (8.23)\end{aligned}$$

where $\mathbf{B}(\mathbf{Z})$ has elements

$$\begin{aligned} b_{ij} &= \begin{cases} -\dfrac{w_{ij}\delta_{ij}}{d_{ij}(\mathbf{Z})} & \text{for } i \neq j \text{ and } d_{ij}(\mathbf{Z}) \neq 0 \\ 0 & \text{for } i \neq j \text{ and } d_{ij}(\mathbf{Z}) = 0 \end{cases} \\ b_{ii} &= -\sum_{j=1, j\neq i}^{n} b_{ij}. \end{aligned} \tag{8.24}$$

Because equality occurs if $\mathbf{Z} = \mathbf{X}$, we have obtained the majorization inequality

$$-\rho(\mathbf{X}) \quad = \quad -\text{tr } \mathbf{X}'\mathbf{B}(\mathbf{X})\mathbf{X} \leq -\text{tr } \mathbf{X}'\mathbf{B}(\mathbf{Z})\mathbf{Z}.$$

Thus, $-\rho(\mathbf{X})$ can be majorized by the function $-\text{tr } \mathbf{X}'\mathbf{B}(\mathbf{Z})\mathbf{Z}$, which is a linear function in $\mathbf{X}$.

Consider an example for the computation of $\mathbf{B}(\mathbf{Z})$. Let all $w_{ij} = 1$, the dissimilarities be equal to

$$\boldsymbol{\Delta} = \begin{bmatrix} 0 & 5 & 3 & 4 \\ 5 & 0 & 2 & 2 \\ 3 & 2 & 0 & 1 \\ 4 & 2 & 1 & 0 \end{bmatrix}, \tag{8.25}$$

and the matrix of coordinates $\mathbf{Z}$ and their distances be

$$\mathbf{Z} = \begin{bmatrix} -.266 & -.539 \\ .451 & .252 \\ .016 & -.238 \\ -.200 & .524 \end{bmatrix} \text{ and } \mathbf{D}(\mathbf{Z}) = \begin{bmatrix} .000 & 1.068 & .412 & 1.065 \\ 1.068 & .000 & .655 & .706 \\ .412 & .655 & .000 & .792 \\ 1.065 & .706 & .792 & .000 \end{bmatrix}. \tag{8.26}$$

The elements of the first row $\mathbf{B}(\mathbf{Z})$ are given by

$$\begin{aligned} b_{12} &= -w_{12}\delta_{12}/d_{12}(\mathbf{Z}) = -5/1.068 = -4.682 \\ b_{13} &= -w_{13}\delta_{13}/d_{13}(\mathbf{Z}) = -3/0.412 = -7.273 \\ b_{14} &= -w_{14}\delta_{14}/d_{14}(\mathbf{Z}) = -4/1.065 = -3.756 \\ b_{11} &= -(b_{12} + b_{13} + b_{14}) = -(-4.682 - 7.273 - 3.756) = 15.712. \end{aligned}$$

In the same way, all elements of $\mathbf{B}(\mathbf{Z})$ can be computed, yielding

$$\mathbf{B}(\mathbf{Z}) = \begin{bmatrix} 15.712 & -4.682 & -7.273 & -3.756 \\ -4.682 & 10.570 & -3.052 & -2.835 \\ -7.273 & -3.052 & 11.588 & -1.263 \\ -3.756 & -2.835 & -1.263 & 7.853 \end{bmatrix}.$$

The Smacof *Algorithm for Majorizing Stress*

Combining (8.17) and (8.25) gives us the majorization inequality for the Stress function; that is,

$$\begin{aligned} \sigma_r(\mathbf{X}) &= \eta_\delta^2 + \text{tr } \mathbf{X}'\mathbf{V}\mathbf{X} - 2\text{tr } \mathbf{X}'\mathbf{B}(\mathbf{X})\mathbf{X} \\ &\leq \eta_\delta^2 + \text{tr } \mathbf{X}'\mathbf{V}\mathbf{X} - 2\text{tr } \mathbf{X}'\mathbf{B}(\mathbf{Z})\mathbf{Z} = \tau(\mathbf{X}, \mathbf{Z}). \end{aligned} \tag{8.27}$$

Thus $\tau(\mathbf{X}, \mathbf{Z})$ is a simple majorizing function of Stress that is quadratic in $\mathbf{X}$. Its minimum can be obtained analytically by setting the derivative of $\tau(\mathbf{X}, \mathbf{Z})$ equal to zero; that is,

$$\nabla\tau(\mathbf{X}, \mathbf{Z}) = 2\mathbf{V}\mathbf{X} - 2\mathbf{B}(\mathbf{Z})\mathbf{Z} = \mathbf{0},$$

so that $\mathbf{V}\mathbf{X} = \mathbf{B}(\mathbf{Z})\mathbf{Z}$. To solve this system of linear equations for $\mathbf{X}$, we would usually premultiply both sides by $\mathbf{V}^{-1}$. However, the inverse $\mathbf{V}^{-1}$ does not exist, because $\mathbf{V}$ is not of full rank. Therefore, we revert to the Moore–Penrose[3] inverse. The Moore–Penrose inverse of $\mathbf{V}$ is given by $\mathbf{V}^{+} = (\mathbf{V} + \mathbf{1}\mathbf{1}')^{-1} - n^{-2}\mathbf{1}\mathbf{1}'$. The last term, $-n^{-2}\mathbf{1}\mathbf{1}'$, is irrelevant in Smacof as $\mathbf{V}^{+}$ is subsequently multiplied by a matrix orthogonal to $\mathbf{1}$, because $\mathbf{B}(\mathbf{Z})$ also has eigenvector $\mathbf{1}$ with eigenvalue zero. This leads us to the update formula of the Smacof algorithm,

$$\mathbf{X}^{u} = \mathbf{V}^{+}\mathbf{B}(\mathbf{Z})\mathbf{Z}. \tag{8.28}$$

If all $w_{ij} = 1$, then $\mathbf{V}^{+} = n^{-1}\mathbf{J}$ with $\mathbf{J}$ the *centering matrix* $\mathbf{I} - n^{-1}\mathbf{1}\mathbf{1}'$, so that the update simplifies to

$$\mathbf{X}^{u} = n^{-1}\mathbf{B}(\mathbf{Z})\mathbf{Z}. \tag{8.29}$$

De Leeuw and Heiser (1980) call (8.28) the *Guttman transform*, in recognition of Guttman (1968).

The majorization algorithm guarantees a series of nonincreasing Stress values. When the algorithm stops, the stationary condition $\mathbf{X} = \mathbf{V}^{+}\mathbf{B}(\mathbf{X})\mathbf{X}$ holds. Note that after one step of the algorithm $\mathbf{X}$ is column centered, even if $\mathbf{Z}$ is not column centered.

The Smacof algorithm for MDS can be summarized by

```
1. Set $\mathbf{Z} = \mathbf{X}^{[0]}$, where $\mathbf{X}^{[0]}$ is some (non)random start configuration.
   Set $k = 0$. Set $\varepsilon$ to a small positive constant.
2. Compute $\sigma_r^{[0]} = \sigma_r(\mathbf{X}^{[0]})$. Set $\sigma_r^{[-1]} = \sigma_r^{[0]}$.
3. While $k = 0$ or ($\sigma_r^{[k-1]} - \sigma_r^{[k]} > \varepsilon$ and $k \leq$ maximum iterations) do
4.     Increase iteration counter $k$ by one.
5.     Compute the Guttman transform $\mathbf{X}^{[k]}$ by (8.29) if all $w_{ij} = 1$,
       or by (8.28) otherwise.
6.     Compute $\sigma_r^{[k]} = \sigma_r(\mathbf{X}^{[k]})$.
```

[3] Gower and Groenen (1991) report some computationally very efficient Moore–Penrose inverses for some special weight matrices, such as those of a cyclic design and a block design (see Table 6.1).

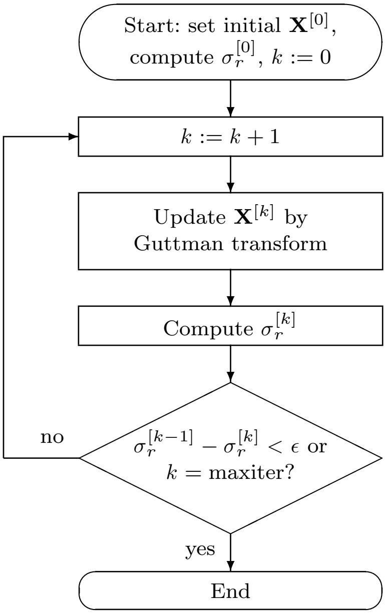

Start: set initial $\mathbf{X}^{[0]}$,
compute $\sigma_r^{[0]}$, $k := 0$
$k := k + 1$
Update $\mathbf{X}^{[k]}$ by
Guttman transform
Compute $\sigma_r^{[k]}$
$\sigma_r^{[k-1]} - \sigma_r^{[k]} < \epsilon$ or
$k =$ maxiter?
no
yes
End

FIGURE 8.11. The flow of the majorization algorithm (SMACOF) for doing MDS.

```
7.      Set $\mathbf{Z} = \mathbf{X}^{[k]}$.
8.  End while
```

A flowchart of the SMACOF algorithm is given in Figure 8.11.

An Illustration of Majorizing Stress

To illustrate the SMACOF algorithm, consider the following example. We assume that all $w_{ij} = 1$, that the dissimilarities $\mathbf{\Delta}$ are those in (8.25), and the starting configuration $\mathbf{X}^{[0]} = \mathbf{Z}$ by (8.26). The first step is to compute $\sigma_r(\mathbf{X}^{[0]})$, which is 34.29899413. Then, we compute the first update $\mathbf{X}^u$ by the Guttman transform (8.29),

$$\begin{aligned}\mathbf{X}^u &= n^{-1}\mathbf{B}(\mathbf{Z})\mathbf{Z} \\ &= \frac{1}{4}\begin{bmatrix} 15.712 & -4.683 & -7.273 & -3.756 \\ -4.683 & 10.570 & -3.052 & -2.835 \\ -7.273 & -3.052 & 11.588 & -1.263 \\ -3.756 & -2.835 & -1.263 & 7.853 \end{bmatrix}\begin{bmatrix} -.266 & -.539 \\ .451 & .252 \\ .016 & -.238 \\ -.200 & .524 \end{bmatrix},\end{aligned}$$

$$\mathbf{X}^u = \begin{bmatrix} -1.415 & -2.471 \\ 1.633 & 1.107 \\ .249 & -.067 \\ -.468 & 1.431 \end{bmatrix} \text{ with } \mathbf{D}(\mathbf{X}^u) = \begin{bmatrix} .000 & 4.700 & 2.923 & 4.016 \\ 4.700 & .000 & 1.815 & 2.126 \\ 2.923 & 1.815 & .000 & 1.661 \\ 4.016 & 2.126 & 1.661 & .000 \end{bmatrix}.$$

TABLE 8.4. The Stress values and the difference between two iterations k of the Smacof algorithm.

k	$\sigma_r^{[k]}$	$\sigma_r^{[k-1]} - \sigma_r^{[k]}$
0	34.29899413	
1	.58367883	33.71531530
2	.12738894	.45628988
3	.04728335	.08010560
4	.02869511	.01858823
5	.02290353	.00579158
6	.02059574	.00230779
7	.01950236	.00109338
8	.01890539	.00059698
9	.01853588	.00036951
10	.01828296	.00025292
11	.01809735	.00018561
12	.01795518	.00014217
13	.01784363	.00011155
14	.01775498	.00008866
15	.01768406	.00007092
16	.01762716	.00005690
17	.01758144	.00004572
18	.01754469	.00003675
19	.01751516	.00002953
20	.01749143	.00002373
21	.01747237	.00001906
22	.01745706	.00001531
23	.01744477	.00001229
24	.01743491	.00000986
25	.01742700	.00000791
26	.01742066	.00000634
27	.01741557	.00000509
28	.01741150	.00000408
29	.01740823	.00000327
30	.01740561	.00000262
31	.01740351	.00000210
32	.01740183	.00000168
33	.01740048	.00000135
34	.01739941	.00000108
35	.01739854	.00000086

The next step is to set $\mathbf{X}^{[1]} = \mathbf{X}^u$ and compute $\sigma_r(\mathbf{X}^{[1]}) = 0.58367883$, which concludes the first iteration. The difference of $\sigma_r(\mathbf{X}^{[0]})$ and $\sigma_r(\mathbf{X}^{[1]})$ is large, 33.71531530, so it makes sense to continue the iterations. The second update is

$$\mathbf{X}^{[2]} = \begin{bmatrix} 1.473 & -2.540 \\ 1.686 & 1.199 \\ .154 & .068 \\ -.366 & 1.274 \end{bmatrix},$$

with $\sigma_r(\mathbf{X}^{[2]}) = .12738894$. We continue the iterations until the difference in subsequent Stress values is less than 10^{-6}. With this value, it can be expected that the configuration coordinates are accurate up to the third decimal. The history of iterations is presented in Table 8.4. After 35 iterations, the convergence criterion was reached with configuration

$$\mathbf{X}^{[35]} = \begin{bmatrix} -1.457 & -2.575 \\ 1.730 & 1.230 \\ -0.028 & 0.160 \\ -0.245 & 1.185 \end{bmatrix}.$$

Various nice results can be derived from the SMACOF algorithm. For example, De Leeuw (1988) showed that $\mathbf{X}^{[k]}$ converges linearly to a stationary point. In technical terms, linear convergence means that $||\mathbf{X}^{[\infty]} - \mathbf{X}^{[k-1]}||/||\mathbf{X}^{[\infty]} - \mathbf{X}^{[k]}|| \rightarrow \lambda$, where $0 < \lambda < 1$ is the largest eigenvalue not equal to 1 of the matrix of the second derivatives of the Guttman transform. Another attractive aspect of SMACOF is that zero distances are unproblematic, because of the definition of b_{ij} in (8.24). In gradient-based algorithms, ad hoc strategies have to be applied if zero distances occur. If no zero distances are present, then it can be shown that the Guttman transform is a steepest descent step with a fixed stepsize parameter.

8.7 Exercises

Exercise 8.1 Consider the function $f(x) = 2x^3 - 6x^2 - 18x + 9$.

(a) Tabulate the values of the function $f(x)$, its derivative $f'(x)$, and $f''(x)$ for x equal to $-4, -3, -2, 1, 0, 1, \ldots, 6$.

(b) Plot all three functions in the same diagram.

(c) Find the minima and maxima of $f(x)$ in the interval $[-4, +6]$ through inspection of the function graph and through computation, respectively.

(d) Interpret $f''(x)$.

Exercise 8.2 Find local and absolute maxima and minima of the following functions.

(a) $y = x^2 - 3x$, for $0 \leq x \leq 5$.

(b) $v = 1 + 2t + 0.5t^2$, for $-3 \leq t \leq 3$.

(c) $u = 1/(2v + 3)$, for $1 \leq v \leq 3$.

(d) $y = x^3 - 3x$, for $-3 \leq x \leq 3$.

Exercise 8.3 Repeat Exercise 8.2 for

(a) $f(x, y) = 4xy - x^2 - y^2$.

(b) $f(x, y) = x^2 - y^2$.

(c) $f(x, y) = x^2 + 2xy + 2y^2 - 6y + 2$.

Exercise 8.4 Use a computer program that does function plots.

(a) Plot $f(x, y) = x^2 + xy - y$.

(b) Find the minimum value of $f(x, y)$ by graphical means.

(c) Find the minimum of $f(x, y)$ by differentiation techniques. [Hint: Use partial differentiation with respect of $f(x, y)$ with respect to x and y, respectively, to obtain the x- and y-coordinates of the minimal point of the function.]

Exercise 8.5 Use matrix differentiation to solve the regression problem $\mathbf{y} \approx \mathbf{Xb}$, where $\mathbf{y}$ is the criterion vector, $\mathbf{X}$ is the battery of predictor vectors (columns), and $\mathbf{b}$ is the vector of unknown weights. Find $\mathbf{b}$ such that $||\mathbf{y} - \mathbf{Xb}||^2$ =min. (Hint: Express the norm as a trace function.)

Exercise 8.6 Use the solution from Exercise 8.5 to solve the following problems.

(a) Find the vector $\mathbf{x}_1$ that solves (7.23) on p. 156 in a least-squares sense. That is, minimize $f(\mathbf{x}_1) = ||\mathbf{A}_1\mathbf{x}_1 - \mathbf{b}_1||^2$ by an appropriate $\mathbf{x}_i$.

(b) What is the value of $f(\mathbf{x}_1)$ at the optimal $\mathbf{x}_1$?

(c) Repeat (a) for $\mathbf{A}_2, \mathbf{b}_2$, and $\mathbf{x}_2$ from (7.24).

(d) Repeat (a) for $\mathbf{A}_3, \mathbf{b}_3$, and $\mathbf{x}_3$ from (7.25).

Exercise 8.7 Suppose that we want to approximate the list of values $0, 2, 6, 5$, and 9 by a single value x. One option is to put these values in the vector $\mathbf{z} = [0\ 2\ 6\ 5\ 9]'$ and minimize the least-squares function $f(x) = \|\mathbf{z} - x\mathbf{1}\|^2$ over x.

(a) Derive $f'(x)$ and express the result in the matrix algebra. [Hint: Start by expanding $f(x)$ into separate terms. Then apply the rules for differentiation to the individual terms.]

(b) Equate the derivative to zero. Can an analytic solution for x be obtained?

(c) For what value of x is $f(x)$ at its minimum?

(d) What can you say about the x that minimizes $f(x)$?

Exercise 8.8 Consider the matrix $\mathbf{V}$ in (8.18) on p. 188.

(a) What do you expect to be the outcome of $\mathbf{V1}$ and $\mathbf{1'V}$? Compute the results for the small example of (8.18). Does this result hold for $\mathbf{V}$ being of any size?

(b) Suppose $\mathbf{Y} = \mathbf{Z} + \mathbf{1a}'$. Explain why $\mathbf{VZ} = \mathbf{VY}$.

(c) Show that $\mathbf{V}$ is double centered.

(d) Is the matrix $\mathbf{B}(\mathbf{Z})$ in (8.24) also double centered? Explain why or why not.

(e) As a consequence of (d), how do you expect that a translation of the type $\mathbf{Z} + \mathbf{1a}'$ changes a single iteration (8.29) of the SMACOF algorithm?

Exercise 8.9 In the so-called Median-center problem, the objective is to find a point such that the Euclidean distance to all other points is minimal. Let $\mathbf{Y}$ be the matrix of n given points. The function that needs to be minimized is

$$f(\mathbf{x}) = \sum_{i=1}^{n} d_i(\mathbf{x}),$$

where $d_i(\mathbf{x}) = \|\mathbf{x} - \mathbf{y}_i\|$ and $\mathbf{y}_i$ is row i of $\mathbf{Y}$.

(a) Use the results from the section on majorizing the median to find a majorizing function $g(\mathbf{x}, \mathbf{z})$ that is a weighted sum of $d_i^2(\mathbf{x})$ and where the weights are dependent on the $d_i^2(\mathbf{z})$, where $\mathbf{z}$ is the vector with the previous estimates of $\mathbf{x}$.

(b) Determine the derivative of $g(\mathbf{x}, \mathbf{z})$.

(c) Set the derivative of $g(\mathbf{x}, \mathbf{z})$ equal to zero. Solve this equation for $\mathbf{x}$. (Hint: you will have to use results from Section 7.7.)

(d) Use a program that can do matrix computations and program your majorization algorithm. Choose a random $\mathbf{Y}$ and apply your algorithm to this $\mathbf{y}$. Verify that every subsequent iteration reduces $f(\mathbf{x})$.