Augmentation and Majorization Algorithms for Squared Distance Scaling

J. de Leeuw[*] P.J.F. Groenen[†] R. Pietersz [‡]

February 15, 2004

Abstract

The Elegant algorithm for doing squared-distance multidimensional scaling has been proposed by De Leeuw (1975), but has never been published. The algorithm is based on the idea of augmentation. In this paper, we extend the Elegant algorithm to accommodate differential weights. Moreover, we show that the derivation by augmentation leads to the same algorithm as the derivation by iterative majorization.

Keywords: S-Stress, Multidimensional scaling, iterative majorization, augmentation algorithm.

1 Introduction

In this paper, we study the squared-distance multidimensional scaling. This problem is formalized by minimizing the raw-S-Stress loss function, that is,

$$\sigma(\mathbf{X}) = \sum_{i<j} w_{ij}(\delta_{ij} - d_{ij}^2(\mathbf{X}))^2. \tag{1}$$

[*] Department of Statistics, University of California, Los Angeles, CA 90095-1554, USA (e-mail: deleeuw@stat.ucla.edu)

[†] Econometric Institute, Erasmus University Rotterdam, P.O. Box 1738, 3000 DR Rotterdam, The Netherlands (e-mail: groenen@few.eur.nl)

[‡] Econometric Institute, Erasmus University Rotterdam, P.O. Box 1738, 3000 DR Rotterdam, The Netherlands (e-mail: pietersz@few.eur.nl)

over $\mathbf{X}$. Here $\mathbf{X}$ is an $n \times p$ *configuration*, the w_{ij}'s are known non-negative *weights*, the δ_{ij}'s are known *dissimilarities*, and $d_{ij}^2(\mathbf{X})$ is the *squared Euclidean distance* between rows i and j of $\mathbf{X}$. Thus we fit squared distances to the dissimilarities. Note that the summation is done only over the upper triangular elements of the dissimilarity matrix. The weights w_{ij} are assumed to be irreducible, that is, there does not exist two or more subsets of objects such that all weights between objects belonging to different subsets is zero. This assumption avoids the situation where the MDS problem can be split into two or more independent MDS problems.

We need some convenient matrix expressions for the squared distances. If we define $\mathbf{C} = \mathbf{X}\mathbf{X}'$ then we can write

$$d_{ij}^2(\mathbf{X}) = (\mathbf{e}_i - \mathbf{e}_j)'\mathbf{C}(\mathbf{e}_i - \mathbf{e}_j) = \text{tr}\, \mathbf{C}\mathbf{A}_{ij}, \tag{2}$$

with $\mathbf{e}_i$ and $\mathbf{e}_j$ columns i and j of the $n \times n$ identity matrix and $\mathbf{A}_{ij}$ the matrix

$$\mathbf{A}_{ij} = (\mathbf{e}_i - \mathbf{e}_j)(\mathbf{e}_i - \mathbf{e}_j)'. \tag{3}$$

Many different algorithms have been proposed to minimize the loss function (1). Foremost of these is perhaps the ALSCAL method (Takane, Young, & Leeuw, 1977), which is of the cyclic coordinate descent type. One ALSCAL iteration consists of a cycle over all np coordinates of $\mathbf{X}$, minimizing loss over one coordinate at a time, while keeping the other coordinates fixed at their current values. Since the loss function is a multivariate quartic in $\mathbf{X}$, the coordinate subproblems can be solved by finding the roots of a univariate cubic (and choosing the one corresponding to the minimum).

Even before ALSCAL, De Leeuw (1975) proposed an augmentation algorithm to minimize (1), in the case in which there are no weights. The paper was never published, but the algorithm has been discussed by Takane (1977) and Browne (1987). They did not include the original derivation and a convergence proof. In this paper, we give this missing derivation and extend the augmentation algorithm to include weights. We show that the augmentation algorithm can also be derived as an iterative majorization algorithm[1].

[1] For the cyclic coordinate ascend, block relaxation, alternating least squares, augmentation, and majorization terminology we refer to the Appendix.

2 Augmentation algorithm

Let us analyze (1) more closely and first expand this function as

$$
\begin{aligned}
\sigma(\mathbf{X}) &= \sum_{i<j} w_{ij}\delta_{ij}^2 + \sum_{i<j} w_{ij} d_{ij}^4(\mathbf{X}) - 2\sum_{i<j} w_{ij}\delta_{ij} d_{ij}^4(\mathbf{X}) \\
&= \sum_{i<j} w_{ij}\delta_{ij}^2 + \sum_{i<j} w_{ij}[(\mathbf{e}_i - \mathbf{e}_j)'\mathbf{C}(\mathbf{e}_i - \mathbf{e}_j)]^2 \\
&\quad -2\sum_{i<j} w_{ij}\delta_{ij}(\mathbf{e}_i - \mathbf{e}_j)'\mathbf{C}(\mathbf{e}_i - \mathbf{e}_j) \\
&= \eta_\delta^2 + \eta^2(\mathbf{C}) - 2\rho(\mathbf{C}). \qquad (4)
\end{aligned}
$$

It can be easily recognized that $\eta^2(\mathbf{C})$ is a quadratic function in $\mathbf{C}$ and $\rho(\mathbf{C})$ a linear function in $\mathbf{C}$. Unfortunately, the metric of $\eta^2(\mathbf{C})$ is not simple. What the augmentation algorithm does is to expand $\eta^2(\mathbf{C})$ in such a way that it has a simple metric. The additional terms should not influence the loss function.

We now expand $\eta^2(\mathbf{C})$. We first write

$$
\eta^2(\mathbf{C}) = \sum_{i<j} w_{ij}(\mathbf{e}_i - \mathbf{e}_j)'\mathbf{C}(\mathbf{e}_i - \mathbf{e}_j)(\mathbf{e}_i - \mathbf{e}_j)'\mathbf{C}(\mathbf{e}_i - \mathbf{e}_j).
$$

The difficult part of the metric lies in the fact that the middle metric $(\mathbf{e}_i - \mathbf{e}_j)(\mathbf{e}_i - \mathbf{e}_j)'$ is dependent on i and j. The core of the augmentation algorithm is to replace this middle metric by $(\mathbf{e}_k - \mathbf{e}_l)(\mathbf{e}_k - \mathbf{e}_l)'$ and sum additionally over k and l. This makes the quadratic term in the loss function used in the augmentation algorithm equal to

$$
\begin{aligned}
\eta^2_{\text{aug}}(\mathbf{C}) &= \sum_{i<j}\sum_{k<l} w_{ij}^{1/2} w_{kl}^{1/2}(\mathbf{e}_i - \mathbf{e}_j)'\mathbf{C}(\mathbf{e}_k - \mathbf{e}_l)(\mathbf{e}_k - \mathbf{e}_l)'\mathbf{C}(\mathbf{e}_i - \mathbf{e}_j) \\
&= \sum_{i<j} w_{ij}^{1/2}(\mathbf{e}_i - \mathbf{e}_j)'\mathbf{C}\left[\sum_{k<l} w_{kl}^{1/2}(\mathbf{e}_k - \mathbf{e}_l)(\mathbf{e}_k - \mathbf{e}_l)'\right]\mathbf{C}(\mathbf{e}_i - \mathbf{e}_j) \\
&= \text{tr}\,\mathbf{C}\left[\sum_{k<l} w_{kl}^{1/2}(\mathbf{e}_k - \mathbf{e}_l)(\mathbf{e}_k - \mathbf{e}_l)'\right]\mathbf{C}\left[\sum_{i<j} w_{ij}^{1/2}(\mathbf{e}_i - \mathbf{e}_j)(\mathbf{e}_i - \mathbf{e}_j)'\right] \\
&= \text{tr}\,\mathbf{C}\left[\sum_{k<l} w_{kl}^{1/2}\mathbf{A}_{ij}\right]\mathbf{C}\left[\sum_{i<j} w_{ij}^{1/2}\mathbf{A}_{ij}\right] = \text{tr}\,\mathbf{CSCS}
\end{aligned}
$$

with

$$\mathbf{S} = \sum_{i<j} w_{ij}^{1/2} \mathbf{A}_{ij}. \tag{5}$$

Thus, we see above that if we expand the summation of $\eta^2(\mathbf{C})$ into $\eta^2_{\text{aug}}(\mathbf{C})$, the metric becomes easy. It can be seen that if the summation of k and l is limited to only those elements for which $k = i$ and $l = j$, then the summation above would be equal to $\eta^2(\mathbf{C})$.

These results are used in the augmented loss function

$$\lambda(\mathbf{X}, \Gamma) = \sum_{i<j} \sum_{k<l} w_{ij}^{1/2} w_{kl}^{1/2} (\gamma_{ijkl} - d_{ijkl}(\mathbf{X}))^2, \tag{6}$$

where $d_{ijkl}(\mathbf{X}) = (\mathbf{e}_i - \mathbf{e}_j)'\mathbf{C}(\mathbf{e}_k - \mathbf{e}_l)$. Now minimize the augmented loss function over $\mathbf{X}$ and over Γ constrained by $\gamma_{ijij} = \delta_{ij}$. Thus the “diagonal” elements of Γ are constrained to be equal to the corresponding elements of Δ, and all other elements are free. Write these constraints as $\Gamma \in \mathcal{G}$.

Suppose that we have a known estimate $\mathbf{C}_0$ for $\mathbf{C}$. Then, the best values $\tilde{\gamma}_{ijkl}$ that minimize $\lambda(\mathbf{X}, \Gamma)$ over $\Gamma \in \mathcal{G}$ are

$$\tilde{\gamma}_{ijkl} = \begin{cases} \delta_{ij} & \text{if } k = i \text{ and } l = j, \\ d_{ijkl}(\mathbf{C}_0) & \text{if } k \neq i \text{ and } l \neq j. \end{cases} \tag{7}$$

Thus,

$$\min_{\Gamma \in \mathcal{G}} \lambda(\mathbf{X}, \Gamma) = \sigma(\mathbf{X})$$

and also

$$\min_{\mathbf{X}} \sigma(\mathbf{X}) = \min_{\mathbf{X}} \min_{\Gamma \in \mathcal{G}} \lambda(\mathbf{X}, \Gamma).$$

The augmentation algorithm is defined by using block relaxation on λ, that is, we iteratively alternate minimization over $\mathbf{X}$ for fixed Γ and minimization over $\Gamma \in \mathcal{G}$ for fixed $\mathbf{X}$. Or, in other words, we apply *alternating least squares* to λ.

For the optimal $\tilde{\gamma}_{ijkl}$, the “off-diagonal” elements always have zero error and will not contribute to the loss of $\lambda(\mathbf{X}, \Gamma)$. Therefore, the augmentation has the advantage of a much simpler metric, while still minimizing S-Stress.

Convergence of the algorithm to a stationary point follows from the general theory of block relaxation algorithms. We produce a decreasing sequence of loss function values using a continuous update mapping, and we can thus apply the theory in Zangwill (1969).

To obtain an update for **C** and thus for **X**, we first need to expand $\lambda(\mathbf{X},\tilde{\Gamma})$, that is,

$$\begin{aligned}\lambda(\mathbf{X},\tilde{\Gamma}) &= \sum_{i<j}\sum_{k<l} w_{ij}^{1/2}w_{kl}^{1/2}\tilde{\gamma}_{ijkl}^2 + \sum_{i<j}\sum_{k<l} w_{ij}^{1/2}w_{kl}^{1/2}d_{ijkl}^2(\mathbf{X}) \\ &\quad -2\sum_{i<j}\sum_{k<l} w_{ij}^{1/2}w_{kl}^{1/2}\tilde{\gamma}_{ijkl}d_{ijkl}(\mathbf{X})) \\ &= \eta_{\tilde{\gamma}}^2 + \eta_{\text{aug}}^2(\mathbf{C}) - 2\rho_{\text{aug}}(\mathbf{C}). \end{aligned} \tag{8}$$

In (5) we saw that $\eta_{\text{aug}}^2(\mathbf{C}) = \text{tr } \mathbf{CSCS}$. We now focus on $\rho_{\text{aug}}(\mathbf{C})$, that is,

$$\rho_{\text{aug}}(\mathbf{C}) = \sum_{i<j}\sum_{k<l} w_{ij}^{1/2}w_{kl}^{1/2}\tilde{\gamma}_{ijkl}(\mathbf{e}_i-\mathbf{e}_j)'\mathbf{C}(\mathbf{e}_k-\mathbf{e}_l) \tag{9}$$

According to (7), $\tilde{\gamma}_{ijkl}$ for the “off-diagonal” elements equal $(\mathbf{e}_i-\mathbf{e}_j)'\mathbf{C}_0(\mathbf{e}_k-\mathbf{e}_l)$ and for the diagonal values we have $\tilde{\gamma}_{ijij} = \delta_{ij}$. What we shall do is insert for the diagonal values $\tilde{\gamma}_{ijij} = \delta_{ij} - (\mathbf{e}_i-\mathbf{e}_j)'\mathbf{C}_0(\mathbf{e}_i-\mathbf{e}_j) + (\mathbf{e}_i-\mathbf{e}_j)'\mathbf{C}_0(\mathbf{e}_i-\mathbf{e}_j)$. The advantage is that we can use the last term in the summation over $ijkl$ to get a simplification that is similar as in (5) while the first part $\delta_{ij} - (\mathbf{e}_i-\mathbf{e}_j)'\mathbf{C}_0(\mathbf{e}_i-\mathbf{e}_j)$ only sums over i and j. Now we can simplify (9) as

$$\begin{aligned}\rho_{\text{aug}}(\mathbf{C}) &= \sum_{i<j}\sum_{k<l} w_{ij}^{1/2}w_{kl}^{1/2}(\mathbf{e}_i-\mathbf{e}_j)'\mathbf{C}(\mathbf{e}_k-\mathbf{e}_l)(\mathbf{e}_k-\mathbf{e}_l)'\mathbf{C}_0(\mathbf{e}_i-\mathbf{e}_j) \\ &\quad +\sum_{i<j} w_{ij}[\delta_{ij} - (\mathbf{e}_i-\mathbf{e}_j)'\mathbf{C}_0(\mathbf{e}_i-\mathbf{e}_j)](\mathbf{e}_i-\mathbf{e}_j)'\mathbf{C}(\mathbf{e}_i-\mathbf{e}_j) \\ &= \text{tr } \mathbf{C}\left[\sum_{k<l} w_{kl}^{1/2}(\mathbf{e}_k-\mathbf{e}_l)(\mathbf{e}_k-\mathbf{e}_l)'\right]\mathbf{C}_0\left[\sum_{i<j} w_{ij}^{1/2}(\mathbf{e}_i-\mathbf{e}_j)(\mathbf{e}_i-\mathbf{e}_j)'\right] \\ &\quad +\text{tr } \mathbf{C}\left[\sum_{i<j} w_{ij}(\delta_{ij} - d_{ij}^2(\mathbf{C}_0))(\mathbf{e}_i-\mathbf{e}_j)(\mathbf{e}_i-\mathbf{e}_j)'\right] \\ &= \text{tr } \mathbf{CSC}_0\mathbf{S} + \text{tr } \mathbf{C}\left[\sum_{i<j} w_{ij}(\delta_{ij} - d_{ij}^2(\mathbf{C}_0))\mathbf{A}_{ij}\right] \\ &= \text{tr } \mathbf{CSC}_0\mathbf{S} + \text{tr } \mathbf{CV} \end{aligned}$$

where

$$\mathbf{V} = \sum_{i<j} w_{ij}(\delta_{ij} - d_{ij}^2(\mathbf{C}_0))\mathbf{A}_{ij}. \tag{10}$$

Before we arrive at a compact expression for $\lambda(\mathbf{X}, \tilde{\Gamma})$, define the eigendecomposition of $\mathbf{S}$ be given by $\mathbf{S} = \mathbf{Q}\Phi\mathbf{Q}'$. In the following, we need $\mathbf{S}^{1/2}$ and $\mathbf{S}^{-1/2}$. Therefore, define the power r of a matrix as

$$\mathbf{S}^r = \sum_{i=1}^{n} a_i \mathbf{q}_i \mathbf{q}_i' \tag{11}$$

where $a_i = \phi_{ii}^r$ if $\phi_{ii} \neq 0$ and $a_i = 0$ if $\phi_{ii} = 0$. Now, we can express $\lambda(\mathbf{X}, \tilde{\Gamma})$ as

$$\begin{aligned}
\lambda(\mathbf{X}, \tilde{\Gamma}) &= \eta_{\tilde{\gamma}}^2 + \text{tr}\,\mathbf{CSCS} - 2(\text{tr}\,\mathbf{CSC}_0\mathbf{S} + \text{tr}\,\mathbf{CV}) \\
&= \eta_{\tilde{\gamma}}^2 + \text{tr}\,(\mathbf{S}^{1/2}\mathbf{C}\mathbf{S}^{1/2})(\mathbf{S}^{1/2}\mathbf{C}\mathbf{S}^{1/2}) - 2\text{tr}\,(\mathbf{S}^{1/2}\mathbf{C}\mathbf{S}^{1/2})(\mathbf{S}^{1/2}\mathbf{C}_0\mathbf{S}^{1/2}) \\
&\quad -2\text{tr}\,\mathbf{S}^{1/2}\mathbf{C}\mathbf{S}^{1/2}\mathbf{S}^{-1/2}\mathbf{V}\mathbf{S}^{-1/2} \\
&= \eta_{\tilde{\gamma}}^2 + \|\mathbf{S}^{1/2}\mathbf{C}\mathbf{S}^{1/2} - (\mathbf{S}^{1/2}\mathbf{C}_0\mathbf{S}^{1/2} + \mathbf{S}^{-1/2}\mathbf{V}\mathbf{S}^{-1/2})\|^2 \\
&\quad -\|\mathbf{S}^{1/2}\mathbf{C}_0\mathbf{S}^{1/2} + \mathbf{S}^{-1/2}\mathbf{V}\mathbf{S}^{-1/2}\|^2.
\end{aligned} \tag{12}$$

To simplify notation in (12) even further, let k contain the constant terms not depending on $\mathbf{C}$ (and thus not on $\mathbf{X}$), that is, $k = \eta_{\tilde{\gamma}}^2 - \|\mathbf{S}^{1/2}\mathbf{C}_0\mathbf{S}^{1/2} + \mathbf{S}^{-1/2}\mathbf{V}\mathbf{S}^{-1/2}\|^2$, let $\mathbf{Y} = \mathbf{S}^{1/2}\mathbf{X}$ so that $\mathbf{YY}' = \mathbf{S}^{1/2}\mathbf{C}\mathbf{S}^{1/2}$, and let

$$\mathbf{E} = \mathbf{S}^{1/2}\mathbf{C}_0\mathbf{S}^{1/2} + \mathbf{S}^{-1/2}\mathbf{V}\mathbf{S}^{-1/2}. \tag{13}$$

Then, (12) can be written as

$$\lambda(\mathbf{X}, \tilde{\Gamma}) = k + \|\mathbf{YY}' - \mathbf{E}\|^2. \tag{14}$$

It is not difficult to recognize that (14) can be minimized by computing an eigen decomposition on $\mathbf{E}$ and retaining the p largest eigenvalues and corresponding eigenvectors. Thus, if $\mathbf{E} = \mathbf{Q}\Phi\mathbf{Q}'$, Φ_p is a diagonal $p \times p$ matrix with the largest p positive eigenvalues on the diagonal, and $\mathbf{Q}_p$ contains the corresponding eigenvectors, then the update is given by $\mathbf{Y} = \mathbf{Q}_p\Phi_p^{1/2}$, or, equivalently,

$$\mathbf{X}^+ = \mathbf{S}^{-1/2}\mathbf{Q}_p\Phi_p^{1/2}. \tag{15}$$

3 The Elegant Algorithm

Let us summarize the results above and describe the augmentation algorithm. The algorithm itself is named Elegant and described below:

```
$t = 0$;
Set $\mathbf{X}_0$ to a random initial configuration;
Compute $\mathbf{C}_0 = \mathbf{X}_0\mathbf{X}_0'$;
Compute $\mathbf{S}$ according to (5);
Compute $\mathbf{S}^{1/2}$ and $\mathbf{S}^{-1/2}$ according to (11);
Set $\sigma_{-1} = \sigma(\mathbf{X}_0) + 2\epsilon$;
while $\sigma_{t-1} - \sigma(\mathbf{X}_t) > \epsilon$ do
    $t := t + 1$;
    $\sigma_{t-1} = \sigma(\mathbf{X}_{t-1})$;
    Compute $\mathbf{V}$ by (10) and $\mathbf{E}$ by (13);
    Compute the eigen decomposition of $\mathbf{E}$;
    Compute the update $\mathbf{X}_t$ by (15);
    Compute $\mathbf{C}_0 = \mathbf{X}_t\mathbf{X}_t'$;
end
```

The Elegant algorithm has the property that in each iteration S-Stress is guaranteed not to increase. In practical situations, this property implies that the algorithm will stop at a local minimum.

4 Iterative Majorization

The majorization algorithm is based on the fact that the loss function is quadratic in the elements of $\mathbf{C}$. By choosing a smart majorizing function for the square of the squared distances, we can again reduce the iteration to computing an optimal rank p approximation.

Consider the basic inequality

$$w_{ij}^{1/2}(\mathbf{e}_i - \mathbf{e}_j)'(\mathbf{C} - \mathbf{C}_0)[w_{ij}^{1/2}(\mathbf{e}_i - \mathbf{e}_j)(\mathbf{e}_i - \mathbf{e}_j) - \mathbf{Z}]'(\mathbf{C} - \mathbf{C}_0)(\mathbf{e}_i - \mathbf{e}_j) \leq 0 \tag{16}$$

which holds for any $\mathbf{Z}$ such that $w_{ij}^{1/2}(\mathbf{e}_i - \mathbf{e}_j)(\mathbf{e}_i - \mathbf{e}_j) - \mathbf{Z}$ is negative definite. Now, choose $\mathbf{Z}$ as

$$\mathbf{S} = \sum_{k<l} w_{kl}^{1/2}(\mathbf{e}_k - \mathbf{e}_l)(\mathbf{e}_k - \mathbf{e}_l)' = \sum_{k<l} w_{kl}^{1/2}\mathbf{A}_{kl} \tag{17}$$

which is a positive semi-definite matrix because the weights w_{kl} are nonnegative, $(\mathbf{e}_k - \mathbf{e}_l)(\mathbf{e}_k - \mathbf{e}_l)'$ is by definition positive semi-definite, and a sum of positive semi-definite matrices is again positive semi-definite. One combination kl will be equal to ij. Therefore, $w_{ij}^{1/2}(\mathbf{e}_i - \mathbf{e}_j)(\mathbf{e}_i - \mathbf{e}_j) - \mathbf{S}$ is equal to $-\sum_{k<l,kl\neq ij} w_{kl}^{1/2}\mathbf{A}_{kl}$. Therefore, $w_{ij}^{1/2}(\mathbf{e}_i - \mathbf{e}_j)(\mathbf{e}_i - \mathbf{e}_j) - \mathbf{S}$ is negative semi-definite.

Expanding (16) for $\mathbf{Z} = \mathbf{S}$ gives

$$\begin{aligned}
& w_{ij}\text{tr}\,\mathbf{C}\mathbf{A}_{ij}\mathbf{C}\mathbf{A}_{ij} \leq && (18)\\
& \quad \text{tr}\,\mathbf{CSC}[w_{ij}^{1/2}\mathbf{A}_{ij}] - 2\text{tr}\,\mathbf{CSC}_0[w_{ij}^{1/2}\mathbf{A}_{ij}] \\
& \quad +2\text{tr}\,\mathbf{C}[w_{ij}d_{ij}^2(\mathbf{C}_0)\mathbf{A}_{ij}] + \text{tr}\,\mathbf{C}_0\mathbf{SC}_0[w_{ij}^{1/2}\mathbf{A}_{ij}] - w_{ij}d_{ij}^4(\mathbf{C}_0) && (19)
\end{aligned}$$

and summing both sides over $i < j$

$$\begin{aligned}
\eta_{\text{aug}}^2(\mathbf{C}) &= \sum_{i<j} w_{ij}\text{tr}\,\mathbf{C}\mathbf{A}_{ij}\mathbf{C}\mathbf{A}_{ij} \\
&\leq \text{tr}\,\mathbf{CSCS} - 2\text{tr}\,\mathbf{CSC}_0\mathbf{S} + 2\text{tr}\,\mathbf{C}[\sum_{i<j} w_{ij}d_{ij}^2(\mathbf{C}_0)\mathbf{A}_{ij}] \\
&\quad +\text{tr}\,\mathbf{C}_0\mathbf{SC}_0\mathbf{S} - \eta_{\text{aug}}^2(\mathbf{C}_0) \qquad (20)
\end{aligned}$$

The cross product term of S-Stress can be expressed as

$$\rho(\mathbf{C}) = (\mathbf{e}_i - \mathbf{e}_j)'\mathbf{C}(\mathbf{e}_i - \mathbf{e}_j) = \text{tr}\,\mathbf{C}[\sum_{i<j} w_{ij}\delta_{ij}\mathbf{A}_{ij}]. \qquad (21)$$

Combining the majorizing inequality of (20) with (22) results in

$$\begin{aligned}
\sigma(\mathbf{X}) &= \eta_\delta^2 + \eta^2(\mathbf{C}) - 2\rho(\mathbf{C}) \\
&\leq \eta_\delta^2 + \text{tr}\,\mathbf{CSCS} - 2\text{tr}\,\mathbf{CSC}_0\mathbf{S} - 2\text{tr}\,\mathbf{C}[\sum_{i<j} w_{ij}(\delta_{ij} - d_{ij}^2(\mathbf{C}_0))\mathbf{A}_{ij}] \\
&\quad +\text{tr}\,\mathbf{C}_0\mathbf{SC}_0\mathbf{S} - \eta_{\text{aug}}^2(\mathbf{C}_0) \\
&= \eta_\delta^2 + \text{tr}\,\mathbf{CSCS} - 2\text{tr}\,\mathbf{CSC}_0\mathbf{S} - 2\text{tr}\,\mathbf{CV} + \text{tr}\,\mathbf{C}_0\mathbf{SC}_0\mathbf{S} - \eta_{\text{aug}}^2(\mathbf{C}_0) \\
&= \eta_\delta^2 + \|\mathbf{S}^{1/2}\mathbf{CS}^{1/2} - [\mathbf{S}^{1/2}\mathbf{C}_0\mathbf{S}^{1/2} + \mathbf{S}^{-1/2}\mathbf{VS}^{-1/2}]\|^2 \\
&\quad -\|\mathbf{S}^{-1/2}\mathbf{VS}^{-1/2}\|^2 - 2\text{tr}\,\mathbf{C}_0\mathbf{V} - \eta_{\text{aug}}^2(\mathbf{C}_0) = \mu(\mathbf{X}). \qquad (22)
\end{aligned}$$

It can be verified that the quadratic term in $\mu(\mathbf{X})$ is exactly equal to the one obtained by the augmentation algorithm in (12). Therefore, the solution to minimize $\mu(\mathbf{X})$ is exactly the same as the update derived by the augmentation algorithm. The conclusion is that iterative majorization is exactly the same as the augmentation algorithm.

A Types of algorithms

Suppose f is a function on $X \times Y$. A *block relaxation* algorithm for minimizing f starts with some $x_0 \in X$. In each iteration k we find $y^{(k)} = \text{argmin} y \in Y\ f(x^{(k)}, y)$ and then $x^{(k+1)} = \text{argmin} x \in X\ f(x, y^{(k)})$. Thus we alternate updating x and y. If the function we are minimizing is a least squares loss function, then block relaxation becomes *alternating least squares*. Although we have defined block relaxation for two blocks, it is clear how to generalize it to more than two. If there are more than two blocks it becomes interesting how we cycle through the blocks. If each of the blocks only consists of a single variable, then block relaxation is *cyclic coordinate descend*. Block relation is worthwhile if the subproblems are simple, compared to the original problem. Block relation methods in statistics are discussed in Oberhofer and Kmenta (1974), Jensen, Johansen, and Lauritzen (1991), De Leeuw (1994) and alternating least squares became popular in the ALSOS system summerized by Young (1981).

One important special case of block relaxation is *augmentation*. The problem is to minimize a function g over X, but we assume we can find an *augmentation function* f on $X \times Y$ such that $g(x) = \min_{y \in Y} f(x, y)$. Augmentation algorithms apply block relaxation to the augmentation function. They should be considered if we can find an augmentation function which is simpler to minimize than our original function g. The most familiar examples of augmentation algorithms are in factor analysis, where we augment the reduced correlation matrix by including the diagonal elements, and in unbalanced factorial analysis of variance, where we augment by adding enough elements to each cell to get a balanced design.

Majorization is a special case of augmentation. Again the problem is to minimize $g(x)$ on X. Suppose we can find a *majorization function* f on $X \times X$ such that $g(x) \leq f(x, y)$ for all $x, y \in X$ and $g(x) = f(x, x)$ for all $x \in X$. Then f is an augmentation of g, with the special property that $g(x) = f(x, x) = \min_{y \in X} f(x, y)$ for all x. Again, a *majorization algorithm* applies block relation to the majorization function. Majorization methods are discussed in detail by De Leeuw (1994), Heiser (1995), Lange, Hunter, and Yang (2000) and for quadratic majorization functions by Böhning and Lindsay (1988).

References

Böhning, D., & Lindsay, B. (1988). Monotonicity of quadratic-approximation algorithms. *Annals of the Institute of Statistical Mathematics*, *40*(4), 641-

663.

Browne, M. (1987). The Young-Householder Algorithm and the Least Squares Multdimensional Scaling of Squared Distances. *Journal of Classification*, *4*, 175-190.

De Leeuw, J. (1975). *An Alternating Least Squares Approach to Squared Distance Scaling.*

De Leeuw, J. (1994). Block Relaxation Methods in Statistics. In H. Bock, W. Lenski, & M. Richter (Eds.), *Information systems and data analysis.* Berlin: Springer Verlag.

Heiser, W. (1995). Convergent Computing by Iterative Majorization: Theory and Applications in Multidimensional Data Analysis. In W. Krzanowski (Ed.), *Recent advanmtages in descriptive multivariate analysis.* Oxford: Clarendon Press.

Jensen, S. T., Johansen, S., & Lauritzen, S. L. (1991). Globally convergent algorithms for maximizing a likelihood function. *Biometrika*, *78*, 867–877.

Lange, K., Hunter, D., & Yang, I. (2000). Optimization Transfer Using Surrogate Objective Functions. *Journal of Computational and Graphical Statistics*, *9*, 1–20.

Oberhofer, W., & Kmenta, J. (1974). A general procedure for obtaining maximum likelihood estimates in generalized regression models. *Econometrica*, *42*, 579-590.

Takane, Y. (1977). On the Relations among Four Methods of Multidimensional Scaling. *Behaviormetrika*, *4*, 29–42.

Takane, Y., Young, F., & Leeuw, J. D. (1977). Nonmetric Individual Differences in Multidimensional Scaling: An Alternating Least Squares Method with Optimal Scaling Features. *Psychometrika*, *42*, 7-67.

Young, F. W. (1981). Quantitative analysis of qualitative data. *Psychometrika*, *46*, 357–388.

Zangwill, W. I. (1969). *Nonlinear programming: a unified approach.* Englewood-Cliffs, N.J.: Prentice-Hall.