Carroll & Chang's INDSCAL algorithm for 3-way Metric Weighted MDS.

1. Carroll & Chang's developments are appropriate to 3-way, 2-mode dissimilarity data. They begin with the same first step that was used by Torgerson, except that each dissimilarity matrix is double centered to obtain several scalar-product matrices. In matrix algebra this is:

$$B_k = O_k^{(2)} - O_k^{(2)}U(U'U)^{-1} - (U'U)^{-1}U'O_k^{(2)} + (U'U)^{-1}U'O_k^{(2)}U(U'U)^{-1}$$

   where $O_k^{(2)}$ is subject k's (nxn) matrix of squared dissimilarity data (presumably squared Euclidean distances) and U is a column of n ones.

2. We can show that $B_k = XW_kX' + E$, where X is the nxr stimulus matrix and W_k is the rxr diagonal matrix of weights for subject k.

3. Carroll & Chang's INDSCAL algorithm then solves for an initial stimulus coordinate matrix X by averaging all matrices B_k and performing an eigen-decomposition. Set X^R and X^L to equal X.

4. The algorithm than iterates the following three steps:

   i: Estimate weights, using estimates of X.

   - Re-arrange matrices B_k into an (m x n^2) matrix B whose k'th row is formed from B_k by horizontally adjoining rows of B_k.
   - Form an (r x n^2) matrix P having one row for each dimension. Row *a* contains element-wise products of left and right coordinates, in order: $x_{1a}x_{1a}$, $x_{1a}x_{2a}$, … $x_{1a}x_{na}$, $x_{2a}x_{1a}$, … $x_{2a}x_{na}$, … $x_{na}x_{na}$.
   - We can show that $B = WP$, so that we can solve for the least squares estimates of the weights, given the current coordinates, as:

$$W = BP'(PP')^{-1}.$$

Carroll & Chang's INDSCAL algorithm for 3-way WMDS (continued).

ii: Estimate right-coordinates, using current estimates of left-coordinates and weights W:

- Re-arrange matrices B_k into an (mn x n) matrix B by vertically adjoining the m (n x n) matrices B_k.
- Form an (mn x r) matrix L consisting of m vertically adjoined (n x r) matrices $X_k^L = X^L W_k$.
- We can show that $B = LX^{R\prime}$, so that we can solve for the least squares estimates of the right-coordinates by:

$$X^R = (L'L)^{-1}L'B \ .$$

iii: Estimate left-coordinates, using current estimates of right-coordinates and weights W:

- Re-arrange matrices as above.
- Form an (mn x r) matrix R consisting of m vertically adjoined (n x r) matrices $X_k^R = W_k X^{R\prime}$.
- We can show that $B' = X^L R'$, so that we can solve for the least squares estimates of the left-coordinates by:

$$X^L = BR(R'R)^{-1} \ .$$

4. After sufficient iteration fit $\left\| B_k - X^L W_k X^{R\prime} \right\|$ converges. We discard one of the estimates of X.

Takane, Young & de Leeuw's ALSCAL algorithm for 3-way metric or nonmetric, weighted or unweighted MDS

1. ALSCAL optimizes the fit of the *squared* distances to the *squared* data. The optimization formula, called S-Stress, is:

$$S = \sqrt{\frac{1}{m}\sum_{k=1}^{m}\left[\frac{\sum_i\sum_j (d_{ijk}^2 - o_{ijk}^2)^2}{\sum_i\sum_j o_{ijk}^4}\right]},$$

   where the squared distances are weighted Euclidean distances:

$$d_{ijk}^2 = \sum_{a=1}^{r} w_{ka}(x_{ia} - x_{ja})^2 .$$

2. An initial configuration of stimulus points is obtained via the Togerson approach. This involves estimating the additive constant, averaging data across matrices and double centering. Then an eigen problem is solved. This gives the initial X.

Takane, Young & de Leeuw's ALSCAL algorithm for 3-way WMDS (continued)

3. We iterate the following two steps:

   i: Estimate weights, using current estimates of X.

   - The method is similar to the INDSCAL method for estimating weights, but is based on squared distances rather than scalar products.
   - Note that we can rewrite squared distances as

$$d_{ijk}^2 = \sum_{a=1}^{r} w_{ka} p_{ija} = \underline{w}_k \underline{p}_{ij}',$$

   where $\underline{w}_k$ is an r-element row-vector of the k'th subject's weights, and where $\underline{p}_{ij}$ is an r-element row-vector whose elements are the squared difference in coordinates for points i and j. That is

$$p_{ija} = (x_{ia} - x_{ja})^2 .$$

   - We can re-arrange the matrices $O_k^{(2)}$ of squared data into an ($m \times n^2$) matrix $O^{(2)}$ whose k'th row is formed by horizontally adjoining rows of $O_k^{(2)}$ (the rows contain squared elements).
   - We form an ($r \times n^2$) matrix P having one row for each dimension. Row *a* has elements $\underline{p}_{11}, \underline{p}_{12}, \underline{p}_{13}, \cdots \underline{p}_{1n}, \underline{p}_{21}, \underline{p}_{22} \cdots \underline{p}_{nn}$.
   - We can show that $O = WP$, which can be solved for W, the (mxr) matrix of weights, each row being $\underline{w}_k$, by calculating:

$$W = OP'(PP')^{-1} .$$

Takane, Young & de Leeuw's ALSCAL algorithm for 3-way WMDS (continued)

ii: Estimate coordinates X, using new estimates of weights W.

- This is done by deriving the derivatives of S-Stress, relative to a single coordinate x_{he}.
- The derivation is similar to the corresponding derivation for Kruskal's algorithm.
- There is the major difference, however: Once the formula for the derivatives is obtained, it can be algebraically solved for x_{he}, giving a closed formula for calculating the optimal value of a single coordinate x_{he}, given all current estimates of other coordinates X, and the current estimates of weights W.
- In outline:

  a : Note that the denominator is a constant. Denote it c_k.

  b : Note that S-Stress is a sum of separable functions, denoted S_k,

$$S = \left[\frac{1}{m}\sum_{k=1}^{m} c_k \sum_i \sum_j (d_{ijk}^2 - o_{ijk}^2)^2\right]^{\frac{1}{2}} = \left[\frac{1}{m}\sum_{k=1}^{m} c_k S_k\right]^{\frac{1}{2}} ,$$

$$S_k = \sum_i \sum_j (d_{ijk}^2 - o_{ijk}^2)^2 .$$

  Thus, S-Stress can be separated into a sum of functions S_k, each of which can be optimized with respect to x_{he}.

Takane, Young & de Leeuw's ALSCAL algorithm for 3-way WMDS (continued)

c : Algebra alows us to isolate constant terms from variable ones:

$$S_k = \sum_i \sum_j \left[\left(\sum_{a \neq e} w_{ka}(x_{ia} - x_{ja})^2 \right) + w_{ke}(x_{ie} - x_{je})^2 - o_{ijk}^2 \right]^2 .$$

Then we define

$$a_{ijke}^2 = \frac{1}{w_{ke}} \left[o_{ijk}^2 - \sum_{a \neq e} w_{ka}(x_{ia} - x_{ja})^2 \right],$$

which allows us to re-formulate

$$S_k = w_{ke}^{1/2} \sum_i \sum_j [a_{ijke}^2 - (x_{ie} - x_{je})^2]^2 .$$

Expanding

$$S_k = w_{ke}^{1/2} \sum_i \sum_j [a_{ijke}^2 - x_{ie}^2 + 2x_{ie}x_{je} - x_{je}^2]^2 .$$

Defining the constant, with respect to x_{he} (assumed to be included in the summation over i), $b_{ijke}^2 = (a_{ijke}^2 - x_{je}^2)$, then

$$S_k = w_{ke}^{1/2} \sum_i \sum_j [b_{ijke}^2 - x_{ie}^2 + 2x_{ie}x_{je}]^2 .$$

We have now isolated the constant terms in b_{ijke}^2, from the variable terms involving x_{he}^2 .

Takane, Young & de Leeuw's ALSCAL algorithm for 3-way WMDS (continued)

d : We can show that the dervatives are:

$$\frac{\partial S_k}{\partial x_{he}} = 4w_{ke}^{1/2}\left[\sum_j x_{he}^3 - 3x_{he}^2 x_{je} + 2x_{he}x_{je}^2 - b_{hjke}^2(x_{he} - x_{je})\right] .$$

Substitution gives us:

$$\frac{\partial S}{\partial x_{he}} = \frac{1}{m}\sum_k\left[\frac{4w_{ke}^{1/2}}{\sum_i\sum_j o_{ijk}^4}\left[\sum_j x_{he}^3 - 3x_{he}^2 x_{je} + 2x_{he}x_{je}^2 - b_{hjke}^2(x_{he} - x_{je})\right]\right]$$

e : This is a quartic equation in one unknown (all weights and all coordinates except x_{he} are known) which can be set to zero and solved by standard techniques.

f : After solving for x_{he} we replace the old value for x_{he} with the new value and proceed to estimate the next coordinate for point h, repetedly cycling through all coordinates for point h until they converge.

g : Then we move to another point, using the new values for the previous point.