Nonmetric MDS based on Kruskal's Least Squares Monotonic Transformation

1. Kruskal defined the fit measure he called Stress, and proposed an algorithm for minimizing S as a function of:

   a : the coordinates X from which the distances $d_{ij}[x_i,x_j]$ are computed; *and*

   b : monotonic transformations t of the data $t[o_{ij}]$.

   Stress is denoted S and defined as:

$$S = \sqrt{\frac{\sum_i \sum_j (d_{ij} - t[o_{ij}])^2}{\sum_i \sum_j t[o_{ij}]^2}} \quad .$$

3. Kruskal proposed an algorithm for minimizing S. In other notes we have presented Kruskal's steepest descent algorithm for minimizing S over the distances $d_{ij}[x_i,x_j]$ for a given scaling of the data (which, in those notes, was simply denoted o_{ij}). In these notes we discuss minimizing S over monotonic transformations $t[o_{ij}]$ of the data.

4. We begin by noting that the denominator is the sum of squared transformed data. In transforming these data we impose the requirement that this sum remain constant over whatever transformations t we compute. We restate the problem as optimizing:

$$S^2 = \frac{1}{T} \sum_{i<j} (d_{ij} - t[o_{ij}])^2 \quad .$$

   over monotonic transformations t, where

$$T = \sum_{i<j} t[o_{ij}]^2 \quad .$$

5. Since the specific cell that a datum appears in plays no role here, we simplify notation. Specifically, we reshape the dissimilarity data *O* to be a column vector *o*. This vector has *n(n-1)/2* values taken from the upper (or lower, but not both) triangle of *O*. The values are placed into this vector in ascending order, the smallest values first, the largest last.

   We also “reshape” the distances to be a column vector *d* with elements permuted to be in order corresponding to the values in *o*.

6. The monotonic transformation imposes the restraint that, for pairs of observed dissimilarities which are not tied:

$$(o_p < o_q) \rightarrow (t[o_p] \le t[o_q]) \ .$$

   Kruskal proposed two least squares monotonic transformations which differ in how ties are treated. He called them the “primary” and “secondary” least squares monotonic transformations. They involve:

$$(o_p = o_q) \rightarrow \left(\begin{array}{ll} \text{no equality restriction} & \text{(primary)} \\ (t[o_p] = t[o_q]) & \text{(secondary)} \end{array} \right. \ .$$

   Elsewhere in these notes we have called these two assumptions “continuous” (primary) and “discrete” (secondary). Note that for the primary (and secondary) case there are ordinal restrictions imposed by the relations of o_{ij} and o_{kl} with other data.

7. To be “least squares”, this transformation must yield transformed data $t[o_p]$ which minimize Stress for a given set of distances d_p:

$$S^2 = \frac{1}{T}\left[\sum_p (d_p - t[o_p])^2\right] = \frac{1}{T}\|d - t[o]\|$$

8. Kruskal proved that his transformation algorithm yields the monotonic transformation that is a least squares fit to *d*. Young showed that it can be represented by the formula

$$t[o] = U(U'U)^{-1}U'd \ ,$$

where *U* is an indicator matrix with *n(n-1)/2* rows, one for every distance) and *b* columns, one for every "block", where "block" will be explained below. An indicator matrix is a binary matrix with a single one in each row, the one indicating which block the distance is in.

9. "Blocks" are groups of observations which must be tied in order to maintain monotonicity. The purpose of the iterative algorithm is to determine the block structure. The algorithm is as follows:
   - If there are ties and if this is the "primary" transformation, determine *P*, a permutation matrix which sorts *d* into order within ties.
   - For the "primary" transformation, make the initial *U* an identity matrix. For the "secondary" transformation, the initial *U* indicates tie-group membership, with the initial *U* having b columns, where b is equal to the number of unique values in *o*.
   - Compute $t[o] = U(U'U)^{-1}U'd$ (secondary) or $t[o] = U(U'U)^{-1}U'Pd$ (primary, where P is the needed permutation matrix).
   - If there are no monotonicity violations in *t[o]*, exit. If there are violations, merge adjacent values in *t[o]* into blocks, change b to reflect the number of blocks after merger, and do the previous step again.

10. We can now state Stress as:

$$S^2 = \frac{1}{T}\left\|d - U(U'U)^{-1}U'd\right\| = \frac{\left\|d - U(U'U)^{-1}U'd\right\|}{\left\|U(U'U)^{-1}U'd\right\|} \quad .$$

11. Note that *t[o]* is a least squares fit to *d*, meaning that

$$\left\|d - U(U'U)^{-1}U'd\right\|$$

is minimized. However, Stress is not minimized unless

$$T = \left\|U(U'U)^{-1}U'd\right\|$$

is constant, which it isn't. This leads to the "normalization" step given next.

12. Normalization: We make sure that the denominator *T* (the sum of squares of the transformed data) remains constant by normalizing as follows:

$$t[o] \leftarrow t[o]\left[\frac{\|d\|}{\|t[o]\|}\right].$$

13. Example:

Assume that o = [2 2 2 5 5 5 5 7 7]

Assume that d = [3.90 3.23 4.90 5.23 4.23 4.56 5.23 4.90 3.90]

Assume we are doing a “secondary” transformation.

Try 1:

$$U' = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix},$$

$$t[o]' = \begin{bmatrix} 4.01 & 4.01 & 4.01 & 4.81 & 4.81 & 4.81 & 4.81 & 4.40 & 4.40 \end{bmatrix}$$

Try 2:

$$U' = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$$t[o]' = \begin{bmatrix} 4.01 & 4.01 & 4.01 & 4.67 & 4.67 & 4.67 & 4.67 & 4.67 & 4.67 \end{bmatrix}$$

Note that this last *t[o]* is monotonic with *o*, and is a least squares fit to *o* since *t[o]* consists of means of *o*, within blocks.