8. MINISSA (Michigan-Israel-Nijmegen Integrated Smallest Space Analysis)

1. OVERVIEW

Concisely: MINISSA (Michigan-Israel-Nijmegen Integrated Smallest Space Analysis) provides internal analysis of a two-way symmetric matrix of (dis)similarities by means of an Euclidean distance model using a monotone transformation of the data.

Following the categorisation developed by Carroll and Arabie (1979) the program may be fully described as:

Data:	Model:
One mode	Minkowski metric (restricted)
Two-way	One set of points
Dyadic	One space
Ordinal	Internal
Unconditional	
Complete	
One replication	

1.1 ORIGIN AND VERSIONS OF MINISSA

Two versions of the MINISSA program are available in the MDS(X) series:

- MINISSA(N): a fast, efficient version of the basic Guttman-Lingoes MINI-SSA program, but having a limited number of user options. This version emanates from Nijmegen and is part of Roskam's recently released KUNST library of MDS programs. The program is referred to simply as MINISSA in the MDS(X) library.

- MINISSA(M): based upon the original SSA program in the Michigan (Guttman-Lingoes) series, and contains a large number of user options. It is, consequently, more bulky and computationally more expensive than MINISSA(N). It is referred to as SSA(M) in the MDS(X) series.

The basic form of the two programs is identical, therefore this document describes the shorter MINISSA(N) version. The additional options available in MINISSA(M) are described in Appendix 3.

1.2 BRIEF DESCRIPTION OF MINISSA (UG 3.5)

MINISSA performs what is known as the basic model of MDS by taking (the lower triangle of) a square symmetric matrix whose elements are to be transformed to give the distances of the solution. This transformation will preserve the rank order of the input data. The model is formally equivalent to that developed by Kruskal (1964) although MINISSA uses a hybrid computational approach to the minimization problem, involving techniques originated by both Kruskal and Guttman. This approach is efficient and succeeds better than other programs in avoiding suboptimal solutions (Lingoes and Roskam 1973).

1.3 RELATION TO OTHER PROGRAMS IN THE LIBRARY

The MINISSA method and algorithm form the basis of both MRSCAL and MINICPA programs. In MRSCAL it is assumed that there is a linear or power relation between the data and the solution distances, while in MINICPA the data are treated as asymmetric, (as, for example, when each row of the square data matrix consists of conditional probabilities). In Carroll and Arabie's terminology MINICPA accepts two-mode, two-way data, MINISSA, two-way, one-mode data. Output from MINISSA may be used as input for PINDIS.

2. DESCRIPTION OF THE PROGRAM (UG 2.3)

Since MINISSA is described in some detail in the Users' Guide chapter references are placed at the head of each section of the description.

2.1 DATA

MINISSA accepts as input the lower triangle (without diagonal) of a square symmetric data matrix. Each entry of this input matrix is a measure of (dis)similarity between the row-element and the column element. Commonly these are pairwise ratings of similarity, but any symmetric measure may be used, including correlations, covariances and co-occurrences.

The aim of the algorithm is to position the elements as points in a space of minimum dimensionality so that a measure of departure from perfect fit between the (monotonically) rescaled data and the distances of the solution (STRESS) is minimised. Perfect fit occurs if a monotone transformation of the data can be found which forms a set of actual distances.

2.1.1 Example

Benjamin (1958) collected data on the social mobility of some 2600 subjects using thirteen occupational categories. Macdonald, who is investigating the notion of social distance, uses the index devised by Blau and Duncan (1967, p.43) to measure the dissimilarity in mobility between occupational groups. (For a fuller description of this index see section 2.3.3.4 of the Users' Guide). The measure, writes Macdonald (1972, pp.213-14) may be interpreted as "the percentage of the sons of (group) A that would have to be reallocated jobwise for the sons of A to match the sons of B". He assembles the index values into a lower diagonal matrix, and these are included in the example described in section 4. The scaling solution is discussed at length in Macdonald's article.

2.2 THE ALGORITHM

1. An initial configuration is input by the user, or one is generated by the program (see 2.3.2 below).

2. This configuration is normalised (see 2.2.2 below).

3. The distances between the points are calculated according to the Minkowski metric chosen (see 2.3.3 below).

4. The disparities or fitting-values are calculated (see 2.2.1).

5. STRESS, the index of badness-of-fit between the disparities and the distances, is calculated.

6. A number of tests are performed to determine whether the iterative process should continue, e.g.

   Is STRESS sufficiently low ?

   Has the improvement of STRESS over the last few iterations been so small as to be not worth continuing ?

   Has a specified maximum number of iterations been performed ?

   If the answer to any of these is YES, then the configuration is output as solution. If not, then

7. For each point on each dimension the direction in which it would have to move for STRESS to be minimized is calculated as is the optimal size of the move (the 'step-size').

8. The configuration is moved in accordance with 7 and the program returns to step 2.

2.2.1 Minimization, fitting values (UG 3.5.2)

In MINISSA there are two methods of finding the minimum STRESS value. These are known in Guttman's (1968) terminology as soft and hard squeeze methods. The program begins by using the soft squeeze which minimizes raw STRESS and when this has reached a minimum switches to the hard squeeze and minimizes STRESS1. By convention different fitting values (step 4) are used in the different phases.

2.2.1.1 Soft squeeze

Soft squeeze derives from a technique of Guttman's (1968). It is particularly efficient at quickly reducing STRESS. Fitting values are calculated using a procedure known as rank-image permutation. These fitting values are known as d^* (DSTARS) and have the property of being strongly monotone with the data. That is to say that unequal data values must be matched with unequal fitting values (formally if $\delta_{ij} > \delta_{k\ell}$ then $d^*_{ij} > d^*_{k\ell}$).

2.2.1.2 Hard squeeze

When a minimum has been reached using the soft squeeze the program switches to the so-called hard squeeze, which is a simpler, more well-behaved method. Fitting values are now calculated using a procedure known as monotone regression and are known as $\hat{d}$ (DHATS). These have the property of being weakly monotone with the data in that unequal data may be matched with equal fitting values if in so doing STRESS is reduced (formally, if $\delta_{ij} > \delta_{k\ell}$ then $\hat{d}_{ij} \geq \hat{d}_{k\ell}$).

To summarise:

	SOFT SQUEEZE (initial method)	HARD SQUEEZE (second method)
Minimises:	Raw Stress	$STRESS_1$
Using:	d^* (DSTAR)	$\hat{d}$ (DHAT)
Relation to data:	strongly monotone	weakly monotone

Users who wish to vary the combination of fitting values with methods are referred to SSA(M).

2.2.2 STRESS and normalization (UG 3.3)

In the so-called 'soft-squeeze' the program minimizes raw STRESS (otherwise known as raw phi, or STRESS_o) which is simply the sum of the squared differences between the distances in the configuration and the DSTAR's, i.e. $\sum_{ij} (d_{ij} - d^*_{ij})^2$. Since this index might be minimized by successive scaling down of the overall size of the configuration, the configuration is normalised after each iteration.

In the so-called 'hard-squeeze' however, STRESS_1 is calculated and minimized. STRESS_1 is simply a normalized form of raw STRESS: the normalizing factor being the sum of the squared distances in the configuration. This removes the dependence of the original index on the size of the configuration. Values for STRESS of both flavours are output by the program.

2.2.2.1 Step-size and angle factor

At step 7, the algorithm computes the direction in which each point should be moved in order to reduce STRESS. This is done by calculating the partial derivation of STRESS with respect to each point - the negative gradient. It is also important however correctly to compute the optimal amount of movement in that direction. This is the so-called 'step-size'. This step-size may be changed at each iteration. These changes are monitored by the 'angle factor', which is in effect the cosine of the angle between successive gradients, i.e. the correlation between them. This ensures that, as the program moves towards convergence, and the gradient becomes less steep the step-size will decrease, so as to minimize the possibility of overshooting a minimum STRESS value. MINISSA prints out at termination the final angle factor. At this stage the value ought to be very small. If it is large, then more iterations should be attempted.

2.3 FURTHER OPTIONS IN MINISSA

2.3.1 Ties in the data

It is possible to treat ties in the data in two ways when calculating STRESS. These are known as the primary and secondary approaches and are chosen by the user, by means of the parameter TIES on the PARAMETERS card.

2.3.1.1 The primary approach (TIES (1))

The primary approach allows that if two data elements are equal then the assigned fitting values may be unequal. The tie is broken if, in so doing, STRESS is reduced. Substantively this approach regards ties in the data as relatively unimportant. It is, of course, possible for the program to capitalise on this approach to produce a 'good', though degenerate configuration. If data contain a lot of ties and the program is using the primary approach then long horizontal lines will appear in the Shepard diagram. A number of such horizontal lines is a sign of possible degeneracy in the solution.

2.3.1.2 The secondary approach (TIES (2))

On the other hand, the secondary approach regards the equality of data elements as important information and requires that the fitting values be equal for equal data. This constraint is more stringent than the primary approach and will normally result in higher STRESS values.

2.3.1.3 The parameter EPSILON

A further approach to tied data is given by means of EPSILON on the PARAMETERS card. Each pair of data values will be compared and, if the difference between them is less than this value they will be regarded as tied. This approach is recommended if the user wishes to place little emphasis on the smaller variations in the data.

For a full description of options regarding ties and the preservation of order information, see the Users' Guide section 3.2.3. The user wishing to combine a particular approach to ties with a particular type of fitting value is referred to the options available in SSA(M) described in Appendix 4 below.

2.3.2 The initial configuration (UG 3.5.1 & A3.2)

The values of a 'good' starting point for the iterative process include saving on machine time and avoidance of local minima. Two options exist within MINISSA for the choice of initial configuration.

The user may supply a starting configuration. This may be a guess at the solution, an a priori configuration or a solution to a previous metric scaling. The matrix of coordinates is preceded by a READ CONFIG control card, which will have associated with it an INPUT FORMAT card to read real (F-type) values. The configuration may be input either stimuli (rows) by dimensions (columns) or dimensions (rows) by stimuli (columns). (In this latter case, the parameter MATFORM should be given the value (1) on the PARAMETERS card).

Alternatively, the program will generate a starting configuration with desirable numerical properties. This configuration is the usual one in the Guttman-Lingoes-Roskam MINI programs and uses only the ordinal properties of the data. It has been found to be particularly useful in avoiding problems with local minima. Further details justifying this choice of initial configuration will be found in Lingoes and Roskam (1973, pp.17-19), and Roskam (1975, pp.37-44).

2.3.3 Distances in the configuration

The user may choose how the distances between the points in the configuration are to be computed by the MINKOWSKI parameter. The default of 2.0 gives the ordinary Euclidean metric and 1.0 gives a 'city-block' metric but any positive number may be used. It is however unwise to use large values as there is then a risk of overflow.

2.3.4 The final configuration

When the iterative process is terminated, the current configuration is output as the solution. If the metric is euclidean (i.e. MINKOWSKI(2)) then the configuration is rotated to principal axes. It should be noted that these axes are arbitrary from the point of view of interpretation, but have certain desirable geometric properties. In particular the coordinates of the points on the axes are uncorrelated. Furthermore it is often helpful in deciding on the 'correct' dimensionality of the solution to notice how much variation is associated with each axis. This variation is given in the output by the value SIGMA which is the standard deviation of the coordinates on each axis.

2.3.5 STRESS and dimensionality (UG 3.7)

The estimation of the appropriate dimensionality of an MDS solution is central to the analysis. Three methods are commonly used with MINISSA in addition to that involving SIGMA alluded to above.

The first guideline asserts that the ratio between the number of data elements and the number of latent parameters (i.e. coordinates) should be at least two. This compression ratio should serve as a useful guide when choosing the dimensionalities for a run of the program.

The second is a heuristic device analogous to the familiar "scree test" of factor analysis. STRESS should decrease with increasing dimensionality until in n-2 dimensions a perfect (though trivial) fit will be achieved. If a graph is drawn of STRESS against dimensionality it is a common occurrence to find an 'elbow' - a sharp decrease in STRESS between dimensions occurring at some relatively low dimensionality. At this value, to add dimensions will not significantly improve the fit of data to solution so it is reasonable to attempt interpretation of this solution.

If however 10 and 60 points are being used and the dimensionality is less than or equal to 5 the program will print a value of Stress 1 based on an approximation to random data as detailed in Spence (1979).

2.3.6 Local minima (UG 3.5.4)

For a given set of data each configuration will have an associated STRESS value. The MINISSA procedure finds the 'best' configuration, by finding the partial derivatives of STRESS (with respect to the coordinates). It is possible that a given STRESS value, although locally the minimum attainable, may not be the real 'global' minimum.

As mentioned earlier both a good initial configuration and a hybrid algorithm (such as MINISSA) tend to decrease the possibility of local minima occurring. Relatively high STRESS values may be a sign of local minima as would a decrease in STRESS in decreasing dimensionality. If the user suspects local minima, then it is suggested (s)he try a number of different starting configurations.

3. INPUT PARAMETERS

All parameter keywords may be shortened to the first four letters. All subsequent mis-spellings are ignored.

3.1 LIST OF PARAMETERS

Keyword	Default Value	Function
DATA TYPE	0	0: The data are similarities (high values mean high similarities between points). 1: The data are dissimilarities (high values mean high dissimilarities between points).
MINIMUM ITERATIONS	6	Sets the minimum number of iterations to be performed before the convergence test.
EPSILON	0.0	Data are to be considered tied if difference between them is less than EPSILON.
MATFORM	0	(Only relevant when 'READ CONFIG' is used. 0: The input configuration is punched stimuli (rows) by dimensions (columns). 1: The input configuration is punched dimensions (rows) by stimuli (columns).
TIES	1	1: Primary approach to ties in the data. 2: Secondary approach to ties in the data.
MINKOWSKI	2.0	1: Distances in the configuration are measured by 'city-block' metric. 2: Distances are measured by a Euclidean metric. Any positive number may be used.

3.2 NOTES

1. The control card $\left\{\begin{matrix}\#\\ \text{N}\\ \text{NO}\end{matrix}\right\}$ OF STIMULI may be replaced by the card $\left\{\begin{matrix}\#\\ \text{N}\\ \text{NO}\end{matrix}\right\}$ OF POINTS.

2. The following card is not valid:

   $\left\{\begin{matrix}\#\\ \text{N}\\ \text{NO}\end{matrix}\right\}$ OF SUBJECTS

3. Note that the program expects real (F-type) numbers. The data should be input as the lower half of a matrix without diagonal. The INPUT FORMAT statement should read the longest row of this matrix (i.e. n-1 values when there are n stimuli).

4. Note that MINISSA will not accept negative values.

5. Program limits:

   Maximum number of stimuli = 80
   Maximum number of dimensions = 8

3.3 PRINT, PLOT AND PUNCH OPTIONS

The general format for printing, plotting and punching output is described in the Overview. In the case of MINISSA, the available options are as follows:

3.3.1 PRINT options (output to line printer)

Option	Form	Description
INITIAL	p × r matrix	Initial configuration, either generated by the program or printed by the user (p = no. of stimuli).
FINAL	p × r matrix	Final configuration, rotated to principal components.
DISTANCES	lower triangular, with diagonal	Solution distances between points, calculated according to MINKOWSKI parameter.
FITTING	lower triangular, with diagonal	Fitting values: the disparities (DHAT) values.
RESIDUALS	lower triangular, with diagonal	The difference between the distances and the disparities.
HISTORY		An iteration by iteration history of STRESS and values.

By default only the final configuration and the final STRESS values are printed.

3.3.2 PLOT options (output to line printer)

Option	Description
INITIAL	Up to r(r-1)/2 plots of the initial configuration. (r = no. of dimensions).
FINAL	Up to r(r-1)/2 plots of final configuration (r = no. of dimensions).
SHEPARD	The Shepard diagram of distances plotted against data. Fitting values are shown by *, actual data/distance pairs by 0.
STRESS	Plot of STRESS values by iteration.
POINT	Histogram of point contributions to STRESS.
RESIDUALS	Histogram of residual values.

By default, the Shepard diagram and the final configuration will be plotted. Configuration plots are calibrated both from 0 to 100 and from 0 to the maximum coordinate value.

3.3.3 PUNCH options

Option	Description
SPSS	Outputs I (Row index), J (Column index) and corresponding DATA, DISPARITIES, DISTANCES, RESIDUALS values in the format: (2I3, 4F12.0).
FINAL	Outputs final configuration as stimulus (row) by dimension (column) matrix. Each row is prefaced by the stimulus number. Format: (I4, rF10.0) where r is the number of dimensions.
STRESS	Outputs STRESS value by iteration.

By default, none of these options is produced.

4. EXAMPLES

4.1 TEST RUNS

```
col 1                 col 16

RUN NAME              8 POINT ZERO STRESS DATA
TASK NAME             AS MADE FAMOUS BY USERS GUIDE
N OF STIMULI          8
DIMENSIONS            2
INPUT FORMAT          (7F4.0)
PARAMETERS            TIES(2), DATA(1)
READ MATRIX
  <data>
PRINT                 ALL
PLOT                  SHEP(2)
COMPUTE
FINISH
```

```
col 1          col 16

RUN NAME       OCCUPATIONAL DISSIMILARITY DATA
TASK NAME      AS IN SEC. 2.1.1
N OF STIMULI   13
DIMENSIONS     5 TO 1
PARAMETERS     DATA(1)
INPUT FORMAT   (12F5.0)
COMMENT        THE GROUPS ARE:
               1.  FARMERS
               2.  AGRICULTURAL WORKERS
               3.  HIGHER ADMIN ETC.
               4.  OTHER ADMIN ETC.
               5.  SHOPKEEPERS
               6.  CLERICAL WORKERS
               7.  SHOP ASSISTANTS
               8.  PERSONAL SERVICE
               9.  FOREMEN
               10. SKILLED WORKERS
               11. SEMI-SKILLED WORKERS
               12. UNSKILLED WORKERS
               13. ARMED FORCES (OR)
READ MATRIX
51.1
71.4  75.8
63.0  52.7  36.9
58.6  57.7  40.8  32.3
67.0  55.6  38.6  17.7  38.2
63.4  52.3  39.4  13.4  27.8  27.3
54.5  43.3  55.5  29.3  41.1  35.0  23.5
71.2  47.5  56.5  26.2  41.0  35.6  21.1  36.1
65.2  44.3  62.3  33.0  45.1  42.1  27.4  32.0  14.7
65.7  43.0  68.2  39.0  50.8  47.3  33.3  36.0  15.7   8.4
60.1  34.2  69.4  39.8  51.9  47.2  35.5  30.4  23.9  21.1  19.3
66.7  41.9  62.7  36.1  44.6  42.7  29.0  35.9  21.2  20.7  18.4  18.9
PLOT                                 SHEP(2)
COMPUTE
FINISH
```

BIBLIOGRAPHY

Andrews, D. F. (1972) Plots of high-dimensional data, *Biometrics*, *28*, 125-136.

Bailey, K. D. (1974) Interpreting smallest space analysis, *Sociol. Meth. and Res.*, *3*, 3-29.

Barlow, R. E., D.J. Bartholomew, J. M. Brenner and H. D. Brunk (1972) *Statistical inference under order restrictions*, New York: Wiley.

Benjamin, P. (1957) Intergenerational differences in occupation, *Population Studies*, *11*, 262-8.

Blau, P.M. and O. D. Duncan (1967) *The American Occupational Structure*, New York: Wiley.

Carroll, J. D. and P. Arabie (1980) Multidimensional scaling, in M. R. Rozenzweig and L. W. Porter (eds.) *Annual review of psychology*, Palo Alto, Ca., Annual Reviews.

Coxon, A. P. M. and P. M. Davies (eds.)(1980) *Readings in multidimensional scaling*, London: Macmillan.

Everitt, B.S. and P. Nicholls (1974) Visual techniques for representing multivariate data, *The Statistician*, *24*, 37-49.

Guttman, L. (1968) A general technique for finding the smallest coordinate space for a configuration of points, *Psychometrika*, *33*, 469-506.

Hubert, L. (1974) Some applications of graph theory and related non-metric techniques to problems of approximate seriation: the case of symmetric proximity measures, *Br. J. Math. and Stat. Psych.*, *27*, 133-153.

Kendall, D. G. (1971b) Seriation from abundance matrices, in Hodson *et al* 1971 (op cit), reprinted in Coxon and Davies *op. cit.*

Kruskal, J. B. (1964) Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis, *Psychometrika*, *29*, 1-27, reprinted in Coxon and Davies *op. cit.*

Kruskal, J. B. (1964) Nonmetric multidimensional scaling: a numerical method, *Psychometrika*, *29*, 115-129, reprinted in Coxon and Davies, *op. cit.*

Kruskal, J.B. and J.D. Carroll (1969) Geometric models of badness-of-fit functions, in P.R. Krishnaiah (ed.) Multivariate Analysis II, New York: Academic Press.

Lingoes, J.C. (1977) Identifying directions/regions in the space for interpretation.

Lingoes, J.C. and I. Borg (1979) Identifying spatial manifolds for interpretation, in J.C. Lingoes et al, 1979.

Lingoes, J.C. and E.E. Roskam (1973) A mathematical and empirical study of two multidimensional scaling algorithms, Psychometrika, 38, (supplement), reprinted in Lingoes, Roskam and Borg (eds.) op. cit.

Lingoes, J.C., E.E. Roskam and I. Borg (1979) Geometrical representations of directional data, Ann Arbor: Mathesis Press.

MacDonald, K.I. (1972) MDSCAL and distances between socio-economic groups, in K. Hope (ed.), The Analysis of Social Mobility, Oxford: Clarendon Press.

Rabinowitz, G.B. (1975) An introduction to nonmetric multidimensional scaling, Am. Journ. Pol. Sci., 19, 343-390.

Roskam, E.E. (1975) Non-metric data analysis: general methodology and techniques, The Netherlands: University of Nijmegen Report 75-MA-13.

Shepard, R.N. (1962) The analysis of proximities: multidimensional scaling with an unknown distance function (parts 1 and 2), Psychometrika, 27, 125-246.

Spence, I. (1979)
Mult. Beh. Res., 14

Wagenaar, W.A. and P. Padmos (1971) Quantitative interpretation of stress in Kruskal's multidimensional scaling technique, Brit. J. Math. Statist. Psychol., 24. 101-110, reprinted in Coxon and Davies op. cit.

APPENDIX 1: RELATION OF MINISSA TO OTHER PROGRAMS

The MINISSA program merges the two main traditions of basic non metric MDS: the Shepard-Kruskal approach (using monotone regression, weak monotonicity and minimising STRESS$_1$) and the Guttman-Lingoes approach (using rank images, strong monotonicity and minimising raw STRESS). The former was implemented in the original MDSCAL program, and the latter in the G-L SSA-1 program. Both of these programs are now outdated and have been withdrawn.

The basic model is now implemented as the default option by a number of general purpose programs: KYST (the successor to MDSCAL), TORSCA (for Torgerson Scaling) and ALSCAL-4 (the successor to POLYCON). The chief advantages of MINISSA are its small size and speed of computation and its resistance to suboptimal solutions.

APPENDIX 2:

This appendix relies heavily on Roskam (1975a), which is used with permission.

The MINISSA algorithm

Let the stimuli be indexed $i,j = 1,\dots,n$

Let the coordinates be $x_{ia}, a = 1,\dots,r$

The distance between two points i,j be

$$d_{ij} = \sqrt[u]{\sum | x_{ia} - x_{ja} |^u} \qquad (1)$$

where $u = 1$ (city block)

or $u = 2$ (euclidean)

Let the dissimilarities be δ_{ij} regarded as a lower triangle matrix with diagonal δ_{ii} undefined.

The set of coordinates $\underset{\sim}{X} = \{x_{ia}\}$ we refer to as the configuration.

The STRESS_1 of the configuration is

$$S_1 = \sqrt{\frac{\sum_{ij} (d_{ij} - d^o_{ij})^2}{\sum_{ij} d^2_{ij}}} \quad (i > j) \qquad (2)$$

where d^o_{ij} is

either d^*_{ij} in which case $\delta_{ij} > \delta_{k\ell} \Rightarrow d^*_{ij} > d^*_{k\ell}$

or $\hat{d}_{ij}$ in which case $\delta_{ij} > \delta_{k\ell} \Rightarrow \hat{d}_{ij} \geq \hat{d}_{k\ell}$

Alternatively we may, following Guttman, define the coefficient of alienation (a product moment form), derived as follows. Consider 'raw stress', which is equivalent to 'raw phi'

$$\phi_o = S_o = \sum_{ij} (d_{ij} - d^o_{ij})^2 \qquad (3)$$

The coefficient of alienation, is then

$$K = \sqrt{1 - \frac{(\sum_{ij} d_{ij} \times d^o_{ij})^2}{\sum_{ij} d^2_{ij} \times \sum_{ij} (d^o_{ij})^2}} \qquad (4)$$

Note that $d^o_{ij} = \hat{d}_{ij} \rightarrow K = S$

$d^o_{ij} = \overset{*}{d}_{ij} \rightarrow K = S \sqrt{1 - (\frac{1}{2}S)^2}$

Since $S_i > S_j \Longleftrightarrow K_i > K_j$

we may minimize either K or S. By convention

S is used with $\hat{d}$

and K is used with d^*

The iterative procedure

The iterative process is indexed by a superscript s. Thus $\underset{\sim}{X}^s$ is the configuration at iteration s. At each iteration the first phase consists in finding an improved configuration X^{s+1} which fits best the monotone values $(d^o)^s$. The second phase then finds an improved set of monotone values $(d^o)^{s+1}$ which fit the distances d^{s+1} obtained $\underset{\sim}{X}^{s+1}$. After the second phase we compute stress S^{s+1}.

The calculation of the monotonic transformation is detailed in the Users' Guide 3.2.

The first phase is itself iterative nested within the s iterations. These iterations are counted by t. The STRESS is minimized by a process known as steepest descent. Specifically,

$$x_{ia}^{t+1} = x_{ia}^{t} - \alpha_t \left(\frac{\delta S}{\delta x_{ia}} \right)^{t} \qquad (5)$$

where α is the optimal step size. This method calls for calculation of the partial derivatives of S with respect to x. Two methods are used: one with the so-called soft-squeeze, the other with the hard squeeze.

The soft squeeze approach

We find the derivatives of S_o (raw stress) with respect to x_{ia} by considering

$$\frac{\partial S_o}{\partial d_{ij}} = 2(d_{ij} - d^{*}_{ij}) \qquad (6)$$

Since $$\frac{\partial d_{ij}}{\partial d^{u}_{ij}} = \frac{d_{ij}}{u} \left[\sum_{a} | x_{ia} - x_{ja} |^{u-1} \right] = \frac{1}{u} \left(\frac{1}{d_{ij}} \right)^{u-1}$$

where u is the Minkowski metric

$$\frac{\partial d^{u}_{ij}}{\partial x_{ka}} = \sum_{ij} (\delta^{ki} - \delta^{kj})^{u} \, | x_{ia} - x_{ja} |^{u-2} (x_{ia} - x_{ja}) \qquad (7)$$

(where $\delta^{ki} = 1$ if k = I and $\delta^{kj} = 1$ if k = j and = 0 otherwise)

For convenience, we define

$$w_{ija} = \left\{ \frac{|x_{ia} - x_{ja}|}{d_{ij}} \right\} \qquad (8)$$

with $w_{ija} \equiv 1$ if $d_{ij} = 0$.

Thus: $$\frac{\partial S_o}{\partial x_{ka}} = 2 \sum_{ij} (\delta^{ki} - \delta^{ki}) (d_{ij} - d^*_{ij}) \frac{1}{d_{ij}} w_{ija} (x_{ia} - x_{ja}) \qquad (9)$$

From this we may derive

$$\frac{\partial S}{\partial x_{ka}} = 4 \sum_j \left(1 - \frac{\bar{d}^*_{kja}}{d_{kj}}\right) w_{kja} (x_{ka} - x_{ja}) \qquad (10)$$

Now let $\underset{\sim}{C}$ be the correction matrix

$$c_{kja} \equiv \left(1 - \frac{\bar{d}^*_{kj}}{d_{kj}}\right) w_{kja} \quad (\text{if} \quad k \neq j) \qquad (11)$$

(if $k = j$ c_{kja} is arbitrary though undefined)

Putting (10) to zero and using (11) we get

$$\phi = \sum_j c_{kja} (x_{ka} - x_{ja}) = x_{ka} \sum_j c_{kja} - \sum_j c_{kja} x_{ja} \qquad (12a)$$

$$x_{ka} = \left| \frac{1}{\sum_j c_{kja}} \right| \times \left(\sum_j c_{kja} \; x_{ja}\right) \qquad (12b)$$

Note that this equation is stationary if $\{x\}$ minimizes S.

Alternatively, in matrix notation:

Let $\underset{\sim}{C}_a \equiv \{c_{kja}\}$

$\underset{\sim}{Q}_a \equiv \{q_{ka} \equiv \sum_j c_{kja}$, in diagonal form$\}$

$\underline{x}_a \equiv \{x_{ja}\}$ with $k_{ij} = 1,\ldots,m$

Then $\underline{x}_a = \underset{\sim}{Q}_a^{-1} \underset{\sim}{C}_a \underline{x}_a$

By an arbitrary definition of the diagonal elements of $\underset{\sim}{C}$ which makes

$$q_{ka} \equiv \sum_j c_{kja} = \sum_j w_{kja} = m \text{ if } u = 2$$

we may rewrite 12b as

$$x_{ka} = x_{ka} - \left| \frac{1}{4 \sum_j w_{kja}} \right| \times \left| \frac{\partial S}{\partial x_{ka}} \right| \qquad (13)$$

The soft-squeeze assumes that the matrix is symmetrical and S is evaluated for $i > j$. The equation 13 is used as a steepest descent equation which is, however, monitored by an additional parameter β. Letting t indicate the first phase iteration

$$x_{ka}^{(t+1)} = x_{ka}^{(t)} - \left| \frac{\beta}{2 \sum w_{kja}} \right|^{(t)} \times \left| \frac{\partial S}{\partial x_{ka}} \right|^{(t)} (t=0,1,\ldots,t_{max}) \qquad (14)$$

In actual calculation we obtain the product term by

$$\frac{\beta}{\sum_j w_{kja}} \sum_{kj} (1 - \bar{d}^{*}_{kj} / d_{kj}) \times w_{kja} (x_{ka} - x_{ja}) \qquad (15)$$

<u>The hard squeeze</u>

Alternatively, we can define S_1 (STRESSFORM$_1$, STRESS$_1$)

$$S_1 = \left| \frac{\sum_{ij} (d_{ij} - \hat{d}_{ij})^2}{\sum_{ij} d_{ij}^2} \right| \qquad i > j.$$

Let $S_o = \sum_{ij} (d_{ij} - \hat{d}_{ij})^2$

and $NF1 = \sum_{ij} d^2_{ij}$

i.e. $S_1 = \frac{S_o}{NF1}$

The partial derivative of stress w.r.t. the coordinates is

$$\frac{\partial S_1}{\partial x_{ka}} = \frac{1}{NF1S_1} \sum_j \gamma_{kja} (x_{ka} - x_{ja}) \qquad j = 1,\ldots,m \qquad (16)$$

where $\gamma_{kja} = (1 - \frac{\hat{d}_{kj}}{d_{kj}} - S_1^2) \, w_{kja} \qquad (k \neq j)$

$$\sum_j \gamma_{kja} = (1 - bS_1^2) \sum_j w_{kja} \qquad (j = 1,\ldots m;\ b = \frac{M}{M-2})$$

(NB. As with the soft squeeze the diagonal elements of $\underset{\sim}{\Gamma}_a \equiv \{\gamma_{kja}\}$ can be arbitrarily chosen without affecting validity of 16).

Setting 16 to zero we get

$$x_{ka} = \xi^{-1}_{ka} \sum_j \gamma_{kja} \, x_{ja}$$

where $\underset{\sim}{\Xi}_a$ is a diagonal matrix with elements $\xi_{ka} \equiv \{\sum_j \gamma_{kja}\}$

This is the hard squeeze version of 12b, and may be rewritten as a steepest descent equation. In so doing we adjust the diagonal elements of $\underset{\sim}{\Gamma}_a$ which will, of course, modify the column sums. Without changing the definition of $\underset{\sim}{\Gamma}_a$ we can replace ξ^{-1}_{ka} by $\beta \, \xi^{-1}_{ka}$

Thus by a parallel derivation 10 - 16, we obtain

$$x_{ka}^{(t+1)} = x_{ka}^{(t)} - \left| \frac{\beta\, N_1 S_1}{(1-bS_1^2)\sum_j w_{kja}} \right|^{(t)} \times \left| \frac{\partial S_1}{\partial x_{ka}} \right|^{(t)} \quad (t = 0,1,\ldots t_{max}) \qquad (17)$$

In the actual calculation we evaluate the product term as ...

$$\left| \frac{\beta}{(1-bS_1^2)} \right| \times \left| \frac{1}{\sum_j w_{kja}} \right| \times \sum_{kj} \left| (1 - S_1^2 - \hat{d}_{kj}/d_{kj})\, w_{kja}\, (x_{ka} - x_{ja}) \right| \qquad (18)$$

If the metric is euclidean, then $w_{kja} = 1$.

The stepsize factor (angle factor) β

Essentially the first phase iterations are a steepest descent procedure. The gradient at $\alpha = \alpha^{(t)}$ is written $\underset{\sim}{\nabla}_t$ and consists of the values $\frac{\partial S_1}{\partial x_{ka}}$ or $\frac{\partial S_1}{\partial x_{ka}}$ as the case may be.

Let $\underset{\sim}{\nabla}_t \equiv \{g_{ka}^{(t)}\}$

then the steepest descent method may be written:

$$\underset{\sim}{\alpha}^{(t+1)} = \alpha^{(t)} - \alpha_t\, \nabla_t \quad \begin{cases} t = 0,1,\ldots,t_{max} & \text{(soft squeeze)} \\ t = S & \text{(hard squeeze)} \end{cases} \qquad (19)$$

where α_t is the stepsize at iteration t.

The value of β is determined by:

$$\beta = \begin{cases} \beta_t = 1 & (t = 0) \\ \beta_t = (4^{(\theta^3_{t,t-1})} \times 3\sqrt{\beta_{t-1}}\ ) & (t \geq 1) \end{cases} \qquad (20)$$

where

$$\theta_{t,t-1} = \frac{\nabla_t \cdot \nabla_{t-1}}{|| \nabla_t || \times || \nabla_{t-1} ||} \qquad (21)$$

$$\frac{\sum\limits_{k} \sum\limits_{ka} g_{ka}^{(t)} \; g_{ka}^{(t-1)}}{\left\{ \sum\limits_{ka} \sum\limits_{a} (g_{ka}^{(t)})^2 \right\}^{\frac{1}{2}} \left\{ \sum\limits_{ka} \sum\limits_{a} (g_{ka}^{(t-1)})^2 \right\}^{\frac{1}{2}}} \qquad (t \geq 1) \qquad (22)$$

is the cosine of the angle between successive gradients.

carefully at the graphs to see if the fit looks reasonable. When the fit is bad this may be due to a number of causes, but most likely it will be due to the fact that the model used in the Monte Carlo simulation is not appropriate for the data set. In this case the use of MSPACE may be unwarranted. Bad fits may also be signalled by obtaining a minimum fit value greater than about 25. If the fit is bad, this is very often due to the fact that the one dimensional stress is "unusually" low. In this case, MSPACE will probably underestimate the dimensionality. This kind of situation seems to arise most often when the configuration is ellipsoidal, or cigar shaped. Frequently, in this situation, the pattern of fit values over dimensionalities is not concave upward but contains two minima.

9. MRSCAL (MetRic SCALing)

1. OVERVIEW

Concisely: MRSCAL (MetRic SCALing) provides internal analysis of a two-way data matrix by means of a Minkowski distance model using either a linear or a logarithmic transformation of the data.

Following the categorisation developed by Carroll and Arabie (1979) MRSCAL may be described as:

Data: One mode
Two-way
Dyadic
Unconditional
Complete
One replication

Model: Minkowski metric
One set of points
One space
Internal

1.1 ORIGIN AND VERSIONS OF MRSCAL

The MRSCAL program is the basic metric distance scaling program in Roskam's MINI series. The MRSCAL program in the MDS(X) series is based upon the 1971 and KUNST (1977) versions.

1.2 BRIEF DESCRIPTION OF MRSCAL

The MRSCAL algorithm is a metric counterpart to MINISSA. Its aim is to position a set of stimulus objects as a set of points in a space of minimum dimensionality in much the same way as MINISSA, except that the distances in this space will be a linear (or optionally a logarithmic) function of the dissimilarities between the stimuli. In this it has obvious similarities to 'classic' MDS (Richardson 1938, Young and Householder 1938) and to the linear (metric) scaling procedure developed by Messick and Abelson (1956) and made more widely known by Torgerson (1958). The MRSCAL algorithm however, utilises the iterative procedures which Guttman, Lingoes and Roskam (1971) developed and also allows the user additional options, both in the manner by which the distances in the solution space are measured (see Section 2.2.2) and in the form of the transformation function linking data to distances in the solution (see Section 2.2.4) which make it both more general and more robust than the original procedures.

1.3 RELATION OF MRSCAL TO OTHER PROGRAMS IN THE LIBRARY

MRSCAL is an exact metric counterpart to MINISSA, differing from it in that it restricts the field of possible transformation of the data to linear (or power) ones.

Output from MRSCAL may be input to PINDIS.

2. DESCRIPTION

MRSCAL is dealt with in Chapter 5 of the Users' Guide.

2.1 DATA

MRSCAL accepts as input the lower triangle (without diagonal) of a square symmetric data matrix. Each entry of this matrix will be a measure of the (dis)similarity between the row-element and the column element. If the linear transformation option is chosen it should be borne in mind that product moment correlations and covariances are not generally acceptable in that they are only monotonically (and not linearly) related to distance.

The aim of the algorithm is to position these elements as points in a space of minimum dimensionality such that a STRESS-like measure of departure from perfect fit (the coefficient of alienation) between the (linearly) rescaled data and the distances in the solution is minimised. A perfect fit occurs if a linear (or logarithmic) transformation of the data is found which is a set of actual distances.

2.1.1 Example

Benjamin (1958) collected data on the social mobility of some 2600 subjects using thirteen occupational categories. Macdonald, who is investigating the notion of social distance, uses the index devised by Blau and Duncan (1967, p.43) to measure the dissimilarity in mobility between occupational groups. (For a fuller description of this index see section 2.3.3.4 of the Users' Guide). The measure, writes Macdonald (1972, pp. 213-14) may be interpreted as "the percentage of the sons of (group) A that would have to be reallocated jobwise for the sons of A to match the sons of B". He assembles the index values into a lower diagonal matrix, and these are included in the examples described in section 4. The scaling solution is discussed at length in Macdonald's article.

2.2 THE ALGORITHM

The program proceeds as follows.

1. An initial configuration is input (or one may be generated by the program (see 2.2.1 below)).
2. The configuration is normalised.
3. The inter-point distances are calculated according to the Minkowski metric chosen by the user (see 2.2.2 below).
4. A set of fitting quantities are computed that are
   i) a linear (or power) transformation of the data; and
   ii) a least-squares best-fit to the distances (for details see Appendix 2.)
5. The coefficient of alienation between the fitting-quantities and the distances is computed.
6. A number of tests is performed to determine whether the iterative process should continue; e.g. Is STRESS sufficiently low? Has the improvement in STRESS over the last few iterations been great enough to warrant continuing ? Has a specified maximum number of iterations been performed ?
7. If not, then the gradient is computed. This gives for each point on each dimension the direction in which that point should be moved on that dimension in order that STRESS be minimized.
8. If the gradient is zero then the configuration is output as solution.
9. If not, then the points are moved in accordance with (7) and the program returns to step 2.

2.2.1 Initial configuration

The user may provide a starting configuration by means of the control card READ CONFIG, with its associated INPUT MEDIUM and INPUT FORMAT cards. In this case a coordinate for each point on each dimension is input. This may be done either by stimuli (rows) by dimensions (columns) or dimensions(rows) by stimuli (columns). In this latter case the parameter MATFORM should be given the value 1 on the PARAMETER 1 card.

If this is not done, however, then the program constructs an initial configuration from the original data by the Lingoes-Roskam procedure which, as has often been shown, is a good initial approximation of a solution and also has certain desirable geometrical properties.

2.2.2 Distances in the configuration (UG App. 2.2)

The user may choose the way in which the distance between the points in the configuration is measured by means of the MINKOWSKI parameter. The default value 2 provides for the ordinary Euclidean metric where the distances between two points will be the length of the line joining them. The user may specify any value for the parameter. Commonly used values, however, include 1, the so-called 'city-block' or 'taxi-cab' metric where the distance between the two points is the sum of the differences between their co-ordinates on the axes of the space, and infinity (in MRSCAL approximated by a large number (>25)) the so-called 'dominance' metric when the largest difference on any one axis will eventually come to dominate all others. (Users are warned that high values of MINKOWSKI are liable to produce program failure due to overflow).

2.2.3 STRESS and the coefficient of alienation

The family of STRESS formulae for the MINI series is based on the sum of the squared differences between the fitting-values and the distances. In MRSCAL, since the fitting-values are at interval level, a product-moment form is applicable, represented by MU which is the correlation between the distances and the fitting-values, and is hence a measure of goodness of fit. In addition, a related badness of fit measure very similar to STRESS is calculated, known as the coefficient of alienation, K. The two measures used in MRSCAL are related by:

$$K = \sqrt{(1-MU^2)}$$

2.2.3.1 Angle factor and step-size

At step 7, the algorithm computes the direction in which each point should be moved in order to reduce STRESS. This is done by calculating the partial derivative of STRESS with respect to each point - the negative gradient. It is also important, however correctly, to compute the optimal amount of movement in that direction. This is the so-called 'step-size'. This step-size may be changed at each iteration. These changes are monitored by the 'angle factor', which is in effect the cosine of the angle between successive gradients, i.e. the correlation between them. This ensures that, as the program moves towards convergence, and the gradient becomes less steep the step-size will decrease, so as to minimize the possibility of overshooting a minimum STRESS value. MRSCAL prints out at termination the final angle factor. At this stage the value ought to be very small if it is large, then more iterations should be attempted.

2.2.4 Linear and logarithmic transformations

The most common use of MRSCAL is to find a linear transformation of the data which best fits a configuration of points in the chosen dimensionality. The program will also, however, perform an analysis using logarithmic transformations of the data values. In this case the Shepard diagram will show a smooth exponential curve. The user must specify which transformation is required. If no PARAMETERS card is read and/or no specification of the transformation made, then no analysis will be performed.

2.3 FURTHER FEATURES

2.3.1 The CRITERION parameter

In step 6 of the algorithm a number of stopping tests are performed. One of these involves calculating the improvement in fit between the present and the previous iteration. If the improvement is less than the value given by CRITERION on the PARAMETERS card, then the process is terminated and the current configuration is output as solution. A large value for CRITERION will have the effect of stopping the iterative process earlier than would otherwise be the case. This allows the user to make more cheaply a number of exploratory analyses.

2.3.2 The final configuration

When the iterative process is terminated, the current configuration is output as the solution. If the metric is euclidean (i.e. MINKOWSKI (2)) then the configuration is rotated to principal axes. It should be noted that these axes are arbitrary from the point of view of interpretation, but have certain desirable geometric properties. In particular the coordinates of the points on the axes are uncorrelated. Furthermore it is often helpful in deciding on the 'correct' dimensionality of the solution to notice how much variation is associated with each axis. This variation is given in the output by the value SIGMA which is the standard deviation of the coordinates on each axis.

2.3.3 Dimensionality

As a general rule solutions should be computed in a number of dimensionalities. Since a perfect fit will be obtained in n-2 dimensions the trial dimensionalities should always be in dimensionalities less the n-3. As a guide to the choice of trial dimensionalities it is recommended that the product of stimuli × dimensions should be less than half the number of data elements.

A further method is one superficially similar to the 'scree' test of factor analysis. This involves examining the plot of stress by dimensionality. Since MU is a measure of goodness of fit the plot will show an ascending function and the elbow test for appropriate dimensionality may be performed. The 'appropriate' dimensionality, i.e. one which interpretation may be attempted, is that at which the graph shows an 'elbow', i.e. where the addition of extra dimensions is otiose.

3. INPUT PARAMETERS

3.1 LIST OF PARAMETERS

Keyword	Default Value	Function
DATA TYPE	0	0: The data are similarities 1: The data are dissimilarities.
LINEAR TRANSFORMATION	0	0: Linear transformation is not performed 1: Linear transformation is performed.
LOG TRANSFORMATION	0	0: Logarithmic transformation is not performed 1: Logarithmic transformation is performed.
CRITERION	0.00001	Sets the criterion value for terminating the iterations.
MINKOWSKI	2	Sets the Minkowski metric for the analysis.
MATFORM	0	(ONLY RELEVANT WHEN 'READ CONFIG' IS USED) 0: The input configuration is punched: stimuli (rows) by dimensions (columns). 1: The input configuration is punched: dimensions (rows) by stimuli (columns).

N.B. Either LINEAR TRANSFORMATION or LOG TRANSFORMATION must be specified

```
col 1                    col 16

RUN NAME                 OCCUPATIONAL DISSIMILARITY DATA
TASK NAME                AS IN SEC. 2.1.1
N OF STIMULI             13
DIMENSIONS               5 to 1
PARAMETERS               LINEAR(1), DATA(1)
INPUT FORMAT             (12F5.0)
COMMENT                  THE GROUPS ARE:
                         1.  FARMERS
                         2.  AGRICULTURAL WORKERS
                         3.  HIGHER ADMIN ETC.
                         4.  OTHER ADMIN ETC.
                         5.  SHOPKEEPERS
                         6.  CLERICAL WORKERS
                         7.  SHOP ASSISTANTS
                         8.  PERSONAL SERVICE
                         9.  FOREMEN
                         10. SKILLED WORKERS
                         11. SEMI-SKILLED WORKERS
                         12. UNSKILLED WORKERS
                         13. ARMED FORCES (OR)
READ MATRIX
51.1
71.4  75.8
63.0  52.7  36.9
58.6  57.7  40.8  32.3
67.0  55.6  38.6  17.7  38.2
63.4  52.3  39.4  13.4  27.8  27.3
54.5  43.3  55.5  29.3  41.1  35.0  23.5
71.2  47.5  56.5  26.2  41.0  35.6  21.1  36.1
65.2  44.3  62.3  33.0  45.1  42.1  27.4  32.0  14.7
65.7  43.0  68.2  39.0  50.8  47.3  33.3  36.0  15.7   8.4
60.1  34.2  69.4  39.8  51.9  47.2  35.5  30.4  23.9  21.1  19.3
66.7  41.9  62.7  36.1  44.6  42.7  29.0  35.9  21.2  20.7  18.4  18.9
PLOT                                  SHEP(1)
COMPUTE
FINISH
```

3.2 NOTES

1. The card $\left\{\begin{matrix}\# \\ \mathrm{N} \\ \mathrm{NO}\end{matrix}\right\}$ OF SUBJECTS is not valid with MRSCAL.

2. The card $\left\{\begin{matrix}\# \\ \mathrm{N} \\ \mathrm{NO}\end{matrix}\right\}$ OF STIMULI may be replaced by

   the card $\left\{\begin{matrix}\# \\ \mathrm{N} \\ \mathrm{NO}\end{matrix}\right\}$ OF POINTS

3. a) The program expects input to be in the form of the lower triangle of a matrix of real (F-type) numbers.

   b) The INPUT FORMAT should read the longest, i.e. last, row of this matrix.

4. Maximum no. of stimuli = 80
   Maximum no. of dimensions = 8

3.3 PRINT, PLOT AND PUNCH OPTIONS

The general format for printing, plotting and punching output is described in the Overview. In the case of MRSCAL, the available options are as follows:

3.3.1 PRINT options (output to line printer)

Option	Form	Description
INITIAL	p × r matrix	Initial configuration, either generated by the program or printed by the user (p = no. of stimuli, r = no. of dimensions).
FINAL	p × r matrix	Final configuration, rotated to principal components.
DISTANCES	lower triangular, with diagonal	Solution distances between points, calculated according to MINKOWSKI parameter.
FITTING	lower triangular, with diagonal	Fitting values: the disparities (DHAT) values.
RESIDUALS	lower triangular, with diagonal	The difference between the distances and the disparities.

By default only the final configuration and the final STRESS values are printed.

3.3.2 PLOT options (output to line printer)

Option	Description
INITIAL	Up to r(r-1)/2 plots of the initial configuration. (r = no. of dimensions).
FINAL	Up to r(r-1)/2 plots of final configuration (r = no. of dimensions).
SHEPARD	The Shepard diagram of distances plotted against data. Fitting values are shown by *, actual data/distance pairs by 0.
STRESS	Plot of STRESS by iteration.
POINT	Histogram of point contributions to STRESS.
RESIDUALS	Histogram of residual values (logged).

By default, only the Shepard diagram and the final configuration will be plotted. Configuration plots are calibrated both from 0 to 100 and from 0 to the maximum coordinate value.

3.3.3 PUNCH options

Option	Description
SPSS	Outputs I (Row index), J (Column index) and corresponding DATA, DISPARITIES, DISTANCES, RESIDUALS values in the format: (2I4, 4F10.0).
FINAL	Outputs final configuration as stimulus (row) by dimension (column) matrix. Each row is prefaced by the stimulus number. Format: (I4,rF9.6) where r is the number of dimensions.
STRESS	Outputs STRESS value by iteration.

By default, none of these options is produced.

4. EXAMPLES

4.1 TEST RUNS

```
col 1          col 16

RUN NAME       8 POINT ZERO STRESS DATA
TASK NAME      AS MADE FAMOUS BY USERS' GUIDE
N OF STIMULI   8
DIMENSIONS     2
INPUT FORMAT   (7F4.0)
PARAMETERS     LINE(1), DATA(1)
READ MATRIX

  <data>

PRINT          ALL
PLOT           SHEP(2)
COMPUTE
FINISH
```

APPENDIX 1: RELATION OF MRSCAL TO SIMILAR PROGRAMS OUTSIDE MDS(X)

The earliest work in MDS assumed that the data dissimilarities were direct estimates of Euclidean distances, and solved for the coordinates of the space that generated them. This so-called "classic MDS" thus assumes the distances are at the ratio level of measurement. Later developments (Messick and Abelson, 1956) assumed that the data were "relative" distances - i.e. a linear function of the solution distances, thus implying interval level of measurement - and therefore had to solve additionally for the "additive constant" necessary to turn the data into distance estimates (see Appendix 3). A surprisingly robust procedure for implementing such "linear" or metric scaling is described in detail in Torgerson 1958.

Similar procedures to those provided by MRSCAL are implemented in the following package and programs:

(1) KYST (the successor to the original general purpose package known as MDSCAL) provides options for specifying linear and power transformations relating data to the solution distances, and thus implement linear and logarithmic scaling respectively.

(2) ALSCAL-4 (the successor to POLYCON and TORSCA) also allows the user to specify ratio or interval levels of measurement, which also implement classical and linear scaling respectively. There is an additional facility for the user to specify a polynomial in degree 1 to 4 as the nearest equivalent to a logarithmic transformation.

APPENDIX 2: THE MRSCAL ALGORITHMS

This appendix is based on Roskam 1972 which is used with permission.

Let: $\underset{\sim}{X} \equiv \{x_{ia}\} \qquad i = 1,\ldots,n \quad a = 1,\ldots,r$

be a matrix of coordinates of n points in r dimensions.
The Minkowski distance between two points is

$$d_{ij} = \sqrt[u]{\sum_a | x_{ia} - x_{ja} |^u} \tag{1}$$

where u is the Minkowski parameter.

Let $\underset{\sim}{\Delta} \equiv \{\delta_{ij}\}$ be a matrix of observed dissimilarity values.
The matrix is symmetrical and may not contain missing values.
(Similarly $\underset{\sim}{\Sigma}$ denotes a matrix of similarities).

Let $\underset{\sim}{D}^{o} \equiv \{d^{o}_{ij}\}$ be a matrix of values which is obtained by applying any admissible transformation to $\underset{\sim}{\Delta}$

We choose $\underset{\sim}{D}^{o}$ to minimize

$$\sum_{ij} (d_{ij} - d^{o}_{ij})^2 \tag{2}$$

The minimizing d^{o} is denoted $\hat{d}$.

The STRESS of a configuration S1 is

$$\text{STRESS}_1 = \sqrt{\frac{\sum_{ij} (d_{ij} - \hat{d}_{ij})^2}{\sum_{ij} d^2_{ij}}} \tag{3}$$

If the admissible transformations include a free choice of unit, the $\hat{d}$ has the property

$$\sum_{ij} (d_{ij} - \hat{d}_{ij}) \hat{d}_{ij} = 0 \tag{4}$$

Hence STRESS_1 is equal to the coefficient of alienation

$$K = \sqrt{\frac{(\sum_{ij} d_{ij} \hat{d}_{ij})^2}{\sum_{ij} d_{ij}^2 \sum (\hat{d}_{ij})^2}} \tag{5}$$

which is the sine of the angle between $\underset{\sim}{D}$ and $\underset{\sim}{D}^o$ when considered as vectors in a $n(n-1)/2$ dimensional space.

Alternatively MU, the coefficient of monotonicity is

$$\mu = \sqrt{1 - K^2} = \sqrt{1 - S^2}$$

The MRSCAL procedure is double-phase

The first phase improves a configuration $\underset{\sim}{X}$ so as to minimize K for a given $\underset{\sim}{D}^o$.

The second phase finds, for a given $\underset{\sim}{D}$ the transformation $\underset{\sim}{D}^o$ which minimizes K.

The phases are repeated iteratively. K is evaluated after the second phase.

The first phase is itself iterative, at the end of which MU is evaluated.

Iterations are indexed superscript (s): the iterations within the first phase by t.

The first phase

In the first phase we seek to minimize S. The formula from which $X^{(s)}$ is obtained by finding the derivatives of K w.r.t. $\underset{\sim}{X}$ and setting these to zero.

Let x^t be chosen so that

$$\sum_{ij} \left\{ d_{ij} - \alpha^t \hat{d}_{ij}^{s-1} \right\} \alpha^t \hat{d}_{ij}^{s-1} = 0 \qquad (6)$$

$$d_{ij}^{t=0} = d_{ij}^{s-1}$$

$$x_{ia}^{t=0} = x_{ia}^{s-1}$$

$$\alpha^{t=0} = 1$$

At the end of the first phase - i.e. at or short of convergence of the t iterations we have

$$d_{ij}^{s} = d^{t_{max}}$$

$$x_{ia}^{s} = x_{ia}^{t_{max}}$$

Now let

$$K^t = \sqrt{1 - \frac{(\sum_{ij} d_{ij}\ \alpha\hat{d}_{ij})^2}{\sum_{ij} d_{ij}^2 \sum \alpha d_{ij})^2}} \qquad (7)$$

where

$$d_{ij} = d_{ij}^{t} \text{ and } d_{ij}^{o} = \hat{d}_{ij}^{s-1}$$

Using (6) we find $K^t = S^t$

$$K^t = \text{STRESS}_1^t = \sqrt{\frac{\sum_{ij} (d_{ij} - \alpha\hat{d}_{ij})^2}{\sum_{ij} d_{ij}}} \tag{8}$$

where

$$d_{ij} = d_{ij}^t; \quad \hat{d} = \hat{d}^{s-1}$$

Note the $\text{STRESS}_1^t = \text{STRESS}$ iff $t = 0$

Since D^o is invariant under multiplication by a constant α and since $\tilde{K}^t$ is independent of α we may minimize K^t using (8). Note that $K^t = S^t$ if α is chosen so that (6) is satisfied.

Differentiating K^t (omitting the t index for simplicity)

$$\frac{\partial K}{\partial d_{ij}} = \left(\frac{1}{L} (d_{ij} - \alpha\hat{d}_{ij}) - \frac{1}{N} d_{ij} \right) \tag{9}$$

where

$$L = \sum_{ij} (d_{ij} - \alpha\hat{d}_{ij})^2$$

$$N = \sum_{ij} d_{ij}^2$$

i.e. $K = \frac{L}{N}$

Differentiating d_{ij} w.r.t. x_{ka}

$$\frac{\partial d_{ij}}{\partial x_{ka}} = \left(\delta^{ki} - \delta^{kj} \right) \left(\left| \frac{x_{ia} - x_{ja}}{d_{ij}} \right| \right)^{u-1} \text{sign } (x_{ia} - x_{ja}) \tag{10}$$

where

$\delta^{ki} = 1$ if $k = i$ and $\delta^{ki} = 0$ otherwise.

Define

$$w_{ija} \equiv \left\{ \left| \frac{x_{ia} - x_{ja}}{d_{ij}} \right| \right\}^{u-2} \qquad (i = j) \tag{11}$$

w_{iip} is any arbitrary value; we choose $w_{iip} = 1$

Combining (9) (10) and (11) and expanding $\delta^{ki}\,\delta^{kj}$ we obtain the negative gradient.

$$-\frac{\partial K}{\partial x_{ia}} = \frac{K}{L} \sum_j \left(1 - S^2 - \frac{\alpha \hat{d}_{ij}}{d_{ij}}\right) w_{ija} \left(x_{ia} - x_{ja}\right) \tag{12}$$

Next define

$$\gamma_{ja} = \frac{1}{\alpha} \left[\delta^{ij} \sum_k \frac{\alpha \hat{d}_{ik}}{d_{ik}} w_{ika} - \frac{\alpha \hat{d}_{ij}}{d_{ij}} w_{ija} + (1 - K^2)\, w_{ija} \right] \tag{13}$$

And

$$U = \sum_{ij} d\, d_{ij} \Big/ \sum d^2_{ij} = \frac{1}{\alpha}\left(1 - K^2\right) \tag{14}$$

So

$$\gamma_{ija} = \delta^{ij} \sum_k \frac{\hat{d}_{ik}}{d_{ik}} w_{ika} - \frac{\hat{d}_{ij}}{d_{ij}} w_{ija} + U\, w_{ija} \tag{15}$$

Using 13 and 15 we may rewrite 12 as

$$-\frac{\partial K}{\partial x_{ia}} = \frac{\alpha K}{L} \sum_j \gamma_{ija} \left(x_{ia} - x_{ja}\right) \tag{16}$$

Note that the diagonal γ_{jja} may be freely chosen. In (16) they have no effect since they are multiplied by $(x_{ia} - x_{ja}) = 0$. Setting to zero, we obtain

$$x_{ia} = \frac{1}{\sum_j \gamma_{ija}} \sum \gamma_{ija}\, x_{ja} \tag{17}$$

which, using (11) and (15) is equal to

$$x_{ia} = \frac{1}{U \sum_j w_{ija}} \sum_j \gamma_{ija} \, x_{ja} \tag{18}$$

Following Guttman we solve iteratively by setting

$$x_{ia}^{t+1} \left\{ \frac{1}{U \sum_j w_{ija}} \sum_j \gamma_{ija} \, x_{ja} \right\}^{s-1,t} \tag{19}$$

where s identifies $\hat{d}_{ij} = \hat{d}_{ij}^{s-1}$ in the formula for γ and U and t identifies $x_{ia} = x_{ia}^{t}$ there and in the formulae for w_{ija} and for d_{ij}. Note that α is a dummy coefficient which drops out when the derivative is put to zero.

For an euclidean metric $w_{ija} = 1$ and (19) simplifies accordingly.

The process in (19) seems to converge, but for unknown reasons.

The second phase

The second phase minimizes

$$\sum (d_{ij}^{s} - \alpha \hat{d}^{s-1} - \beta)^2 \qquad s = 1,... \tag{20}$$

This is a simple regression problem solved by

$$\alpha = \alpha^{s} = \frac{\sum_{ij} (d_{ij} \hat{d}_{ij} - \bar{d} \bar{\hat{d}})}{\sum (\hat{d}_{ij} - \bar{\hat{d}}^{2})} \qquad \begin{array}{l} (\hat{d} = \hat{d}^{s-1} \\ \;d = d^{s}) \end{array} \tag{21}$$

where $\beta = \beta^{s} = \bar{d} - \alpha^{s} \bar{\hat{d}}$

where $(\bar{\ })$ denotes the arithmetic mean over i,j.

In order to begin the process $\underset{\sim}{X}^{s=1}$ (the initial configuration) is defined following Guttman and Lingoes (1971)

$\underset{\sim}{D}^{o^{s=0}}$ is set to $\underset{\sim}{\Delta}$

All d_{ij} are set to max d^o_{ij}

The matrix $\underset{\sim}{C} \equiv$

$$c_{ij} = \delta^{ij} \sum_k \left(\frac{d^o_{ik}}{a} - \frac{d^o_{ij}}{a} \right) \qquad (2$$

The initial configuration consists of the first r dimensions of a principal components rotation of $\underset{\sim}{C}$.

10. MVNDS (Maximum Variance Non-Dimensional Scaling)

1. OVERVIEW

Concisely: MVNDS (Maximum Variance Non-Dimensional Scaling) provides internal analysis of (dis)similarity measure in lower triangle or full format by a non-dimensional model using a monotone transformation of the data.

Alternatively, following the categorisation developed by Carroll and Arabie (1979) the program may be described as:

Data: One mode
Dyadic
Symmetric or
non-symmetric
Complete

Model: Distance (obeying triangle
inequality, but non-dimensional)
One set of points
One "space"

1.1 ORIGIN, VERSIONS & ACRONYMS

MVNDS was developed by Shepard and Cunningham of Stanford University and University of California, San Diego. The present program is based on the 1973 version of the original program.

1.2 MVNDS IN BRIEF

The program takes data in the form of a full square matrix, or lower triangle, of (dis)similarity measures and seeks to transform them monotonically into a set of distances which can be made to satisfy the triangle inequality, subject to having the maximum possible variance in magnitude. The resulting "distances" will constitute a metric but the structure or dimensionality of the spanning space will be undefined.

1.3 RELATION OF MVNDS TO OTHER PROGRAMS IN MDS(X)

The MVNDS program takes data in the same form as MINISSA, MRSCAL and, in effect, produces a monotonic transformation of the data. It differs from other MDS programs in producing distances which are not necessarily capable of a spatial representation, and in allowing the user to vary the importance which monotonicity is to be ascribed in obtaining an acceptable solution.

2. DESCRIPTION

2.1 DATA

The program accepts data in one of two forms. Any (dis)similarity coefficients in lower triangular form such as would be acceptable to MINISSA, MRSCAL (q.v.) etc., are also acceptable to MVNDS.

The user is also allowed the option of inputting a full matrix of (dis)similarity measures. This means that the program is faced with two measures of the (dis)similarity between object j and object k. These it averages by taking the geometric mean (the square root of the product) of the two values. The analysis is then performed with one set of values).

2.2 THE MODEL

The basic aim of the MVNDS program is to generate from a given data set a set of coefficients/numbers which satisfy three conditions. First, they must satisfy the metric axioms of positivity, symmetry and triangle inequality; secondly they should be as close as possible to a perfect monotone transformation of the original data, and thirdly, subject to the previous constraints, they should have the maximum possible variance.

2.2.0.1 The metric axioms

If a set of dissimilarity coefficients satisfy the metric axioms they may be regarded as a set of distances between points in a space of some construction, although the nature of the space is undetermined. Simply the axioms require that all the distances be non-zero and positive; that the distance between point i and point j is the same as that between point j and point i (actually that the matrix is symmetric) and, most importantly that the distance between two points via any third point cannot be shorter than the direct distance between the two. This last property is the so-called triangle inequality.

The prime purpose of the program, therefore, is to generate a set of distances satisfying these axioms from some monotone transformation of the data. The further requirement that these distances have maximum variance is justified in Cunningham and Shepard (1974) as being likely to generate a set of distances representable in a space of low dimensionality.

The user is given the option of weighting the importance assigned by the program to the conditions of metricity, monotonicity and maximum variance. This is done by use of the keywords VARIANCE, MONOTONICITY and TRIANGLES on the PARAMETERS card. These keywords should be given an argument of non-zero real (F-type) values. These will reflect the importance which the user wishes the program to assign to each condition. The default options are VARIANCE(1.0), MONOTONICITY(2.0) and TRIANGLES(100.0) which assign relatively little importance to the variance obtained, slightly more to violations in the order of the distances but virtually forces the program to make the solution obey the triangle inequality by weighting any violations extremely heavily in the badness-of-fit criterion. Since this is the main purpose of the program this weighting factor is increased as the number of iterations increases. This balance, with satisfaction of the triangle inequality heavily weighted is the usual mode of use of the program. The VARIANCE and MONOTONICITY weights should be adjusted to reflect the particular concerns of the user.

In general values of between 50 and 300 for TRIANGLES, and for MONOTONICITY of between 1.5 and 10 are appropriate for most data.

2.3 FURTHER FEATURES

2.3.1 The search procedure

The user is expected to specify the mean value of the solution distances. This is specified by use of the keyword MEAN on the PARAMETERS card. The iterative procedure then seeks the optimum set of distances which satisfy the three constraints but only considers solutions with the specified mean. Thus, if a particular mean value produces a badly-fit solution other mean values should be tried.

2.3.1.1 Initial values and local minima

Since the possible solution values are constrained in this way it is likely that the solution will be close to the global minimum (Cunningham and Shepard 1974). Thus the initial values are formed from a linear function of the (dis)similarity measures.

2.3.2 Uses of the MVNDS program

The aim of MVNDS is to produce from the original data a set of distances which are "better behaved" in the sense that they obey the triangle inequality property. Whilst this is a necessary condition which other more restrictive measures, such as Euclidean and hierarchical clustering (ultra metric) distances must obey, it is not sufficient to guarantee that they can be thus represented.

Consequently, MVNDS is used both in its own right, when attention is focussed primarily on the shape of the re-scaling function (since it tends to be more regular than that produced from the basic non-metric distance model); and as a first stage to produce a set of distances which can then be submitted to further analysis of representation. Commonly, metric scaling (MRSCAL) and hierarchical clustering (HICLUS) are used for this purpose. Alternatively, graphic analysis techniques (Flament 1963) can be used to analyze the distances.

3. INPUT PARAMETERS

3.1 LIST OF PARAMETERS

Keyword	Default	Function
DATA TYPE	0	0: Input data is a lower triangle of similarities. 1: Input data is a lower triangle of dissimilarities. 2: Input data is a full (a)symmetric matrix of similarities. 3: Input data is a full (a)symmetric matrix of dissimilarities.
MEAN	10	<any positive number> Sets the required mean distance.
TRIANGLES	100.0	<any positive number> Sets the initial weight for triangle violations.
MONOTONICITY	2.0	<any positive number> Sets the weight for violations of monotonicity requirement.
VARIANCE	1.0	<any positive number> Sets the weight for evaluation of variance in the solution.

3.2 NOTES

1. The data should consist of real (F-type) numbers.

2. The following cards are not valid with MVNDS:

   $\left\{\begin{matrix}\#\\ \text{N}\\ \text{NO}\end{matrix}\right\}$ OF SUBJECTS

   DIMENSIONS

3.3 PROGRAM LIMITS

Maximum number of stimuli = 50

3.4 PRINT, PLOT AND PUNCH OPTIONS

3.4.1 PRINT options

Option	Form	Description
DISTANCES	p × p (lower triangle)	The final solution distances are printed.
FITTING	p × p (lower triangle)	The fitting values are printed.
RESIDUALS	p × p (lower triangle)	The residual values are printed.
VIOLATIONS	p × p (lower triangle)	The matrix is printed, each entry of which records the number of times the particular pair is involved in a violation of the triangle inequality.
HISTORY		An iteration by iteration history of the STRESS values is printed.

3.4.2 PLOT options

Option	Description
SHEPARD	The Shepard diagram is plotted.
RESIDUALS	A histogram of residual values is produced.
POINT	A histogram of the point contributions to the badness-of-fit criterion is produced.
STRESS	A plot of the STRESS values at each iteration is produced.

3.4.3 PUNCH options

Option	Description
SPSS	A card-image file containing the following values is output in a fixed format: $\left.\begin{matrix}I\\J\end{matrix}\right\}$ = stimulus pair index DATA = the data value corresponding to I,J DIST = the solution distance between I,J DISP = the corresponding fitting valı RESID = the corresponding residual value
FINAL	The lower-triangular distance matrix is produced in a fixed format.
STRESS	The STRESS values at each iteration are output in a fixed format.

No punched output is produced by default.

4. EXAMPLES

4.1 Test Run

col 1

```
RUN NAME          MVNDS TEST DATA
TASK NAME         ROTHKOPF MORSE CODE CONFUSION DATA
COMMENT           THE DATA ARE TAKEN FROM ROTHKOPF ANALYSED IN
                  CUNNINGHAM AND SHEPARD (1974)
N OF STIMULI      26
PARAMETERS        DATA(2)
COMMENT           WE ASSUME DEFAULT VALUES FOR THE
                  REMAINING PARAMETERS
INPUT FORMAT      (26F3.0)
READ MATRIX

   <data follow here>

PRINT             VIOLATIONS
PLOT              SHEPARD, POINT
COMPUTE
FINISH
```

BIBLIOGRAPHY

Carroll, J.D. and P. Arabie (1979) Multidimensional scaling, in M.R. Rosenzweig and L.W. Porter (eds) (1980) Annual Review of Psychology, Palo Alto Ca: Annual Reviews.

Cunningham, J.P. (1973) The use of the MVNDS program, Department of Psychology, University of San Diego, mimeo.

Cunningham, J.P. and R.N. Shepard (1974) Monotone mapping of similarities into a general metric space, Journal of Mathematical Psychology, 11, pp 335-363.

Flament, C. (1963) Applications of graph theory to group structure, Englewood Cliffs: Prentice Hall.

Rothkopf, E.Z. (1957) A measure of stimulus similarity and errors in some paired-associate larning tasks, Journal of Experimental Psychology, 53, pp 94-101.

APPENDIX 1:

MVNDS is the original and only program to perform this type of analysis.

APPENDIX 2:

A2.1 The Model

Define

$$\underset{\sim}{\Delta} = \{\delta_{jk}\} \qquad\qquad j,k = 1,\ldots,p$$

as a matrix of similarity coefficients between objects j,k.
We seek:

$$\underset{\sim}{D} = \{d_{jk}\}$$

such that

$$d_{jj} = 0 \tag{1a}$$

$$d_{jk} = d_{kj} \tag{1b}$$

$$d_{jk} \geq 0 \tag{1c}$$

$$d_{jk} + d_{k\ell} \geq d_{j\ell} \tag{1d}$$

Further we require that

$$d_{jk} \leq d_{k\ell} \iff \delta_{jk} \geq \delta_{k\ell} \tag{2}$$

and that subject to 1d and 2 in particular

$$\text{var}\ (\underset{\sim}{D}) = \max\ (\text{var}\ (\underset{\sim}{D})) \tag{3}$$

If we consider $\underset{\sim}{D}$ as a vector $\underset{\sim}{d}_i$ $(i = 1,\ldots,m)$ where $m = p(p-1)/2$ in R^m the m-dimensional space of real numbers then each of the set of $p(p-1)(p-2)/2$ linear equalities of the form

$$d_{jk} + d_{k\ell} \geq d_{j\ell} \tag{1d}$$

defines a closed half-space in *R*. We may arbitrarily define $d_{jj} = 0$ and consider only positive distances to satisfy (1a) and (1c).

If we further require $\underset{\sim}{D}$ to have a user specified, fixed arithmetic mean, $\bar{d}$, then the solution lies on the hyperplane

$$\sum_{i}^{m} d_i = m\bar{d} \tag{4}$$

which intersects the region leaving a closed convex polyhedral region *Q*.

We therefore seek that point in *Q* which satisfies (2) and (3). Any point satisfying (2) satisfies the set of inequalities

$$d_m \geq d_{m-1} \geq d_{m-2} \geq \dots\dots \geq d_2 \geq d_1 \tag{5}$$

which defines m - 1 closed half-spaces

$$d_i - d_{i-1} \geq 0 \tag{6}$$

and the solution lies in the intersection of these with *Q*. If the intersection is empty then we define the departure from monotonicity as

$$L = \frac{\sum_{i} (d_i - \hat{d}_i)^2}{m} \tag{7}$$

where d_i is a best-fitting-monotone function, (Kruskal 1964b, 1971; Miles 1959)being monotonically related to $\underset{\sim}{S}$ and minimizing L.

The optimum point in *Q* subject to (3) is that at which a function f

$$f(\underset{\sim}{d}) = \left[\text{var } \underset{\sim}{d} \right] - w_1 \left[L\ (\underset{\sim}{d}) \right] \tag{8}$$

defined on $\mathcal{R}^m$ is at a maximum.

The function expands to

$$f(\underset{\sim}{d}) = \frac{1}{m}\sum_{i}^{m}(d_i - \bar{d})^2 - \frac{w_1}{m}\sum_{i}^{m}(d_i = \hat{d})^2 \qquad (8a)$$

where w_1 is pre-specified.

A2.2 <u>The Method</u>

Given (8) the problem is to find

$$\max_{d}\ \{\ f(\underset{\sim}{d})\ \mid\ \underset{\sim}{d} \in Q \subset \mathcal{R}^m\ \} \qquad (9)$$

The procedure is iterative with iterations counted as superscript s.

$$\underset{\sim}{d}^{s+1} = \underset{\sim}{d}^{s} + \alpha^{s}\ \underset{\sim}{g}^{s} \qquad (10)$$

where

$$\underset{\sim}{g}^{s} = \{\ g_i^s \mid \frac{\partial f}{\partial d_i^s}\ \} \qquad i = 1,\ldots,m \qquad (11)$$

and

α^s is the step-size

The procedure terminates when the length of $\underset{\sim}{g}^s$ is $\simeq$ zero.

However, we seek f_{max} subject to the constraint that it be within Q.

Thus, at each iteration $\underset{\sim}{g}$ is projected onto $m\bar{d}$ to preserve to prespecified mean.

Also, we have to satisfy the triangle inequality constraint namely (p(p-1)(p-2)/2) inequality constraints of the form

$$d_j + d_k - d_\ell \geq 0 \tag{12}$$

We arrange these in arbitrary order, assigning to each an index q (q=1,...,p(p-1)(p-2)/2.

We define $\underset{\sim}{C}$

$$\underset{\sim}{C} = \left\{ c_{qr} : \left\{ \begin{array}{l} c_{qr} = 1 \ (r=j \text{ or } k) \\ c_{qr} = 0 \ (r \neq j, \ k \text{ or } \ell) \\ c_{qr} = 1 \ (r=1) \end{array} \right\} \right\} \begin{array}{l} q=1,\ldots,p(p-1)(p-2)/2 \\ r=1,\ldots,m \\ \\ \end{array} \tag{13}$$

and

$$c_q = \sum_i^m c_{qi} \, d_i$$

We define a new loss function

$$F(\underset{\sim}{d}) = f(\underset{\sim}{d}) - \frac{w_2}{m} \sum_{c_q < 0} (c_q)^2 \tag{14}$$

where w_2 is user specified

Thus

$$\underset{\sim}{d}^{s+1} = \underset{\sim}{d}^s + \alpha^s \frac{\partial F}{\partial \underset{\sim}{d}^s} \tag{15}$$

which, considered for each element of $\underset{\sim}{d}$,

$$\frac{\partial F}{\partial d_i^s} = \frac{2}{m} (d_i^s - \bar{d}) - \frac{2w_1}{m} (\hat{d}_i^s - d_i^s) - \frac{2w_2}{m} \sum_{c_q < 0} c_{qi} \, c_q \tag{16}$$

In order to ensure convergence two technical requirements are sufficient. We require α be chosen so that

$$F(\underset{\sim}{d}^{s} + \alpha^{s}\,\underset{\sim}{g}^{s}) = \max_{\alpha}\,\{F(\underset{\sim}{d})^{s} + \alpha\underset{\sim}{g}^{s}\} \tag{17}$$

and that $w_2 \rightarrow \infty$ as the number of iterations increases.

A2.3 The initial values

$$d_i^o = b(1 - \frac{s_i}{\bar{s}}) + \bar{d} \tag{18}$$

where

$$0 < b \leq \frac{\bar{s}\;\bar{d}}{\max\,\{s_i\} - \bar{s}}$$

with

$$\bar{s} = \frac{\sum_{i}^{m} s_i}{m}$$