15. TRISOSCAL (TRIadic Similarities Ordinal SCALing)

1. OVERVIEW

Concisely: TRISOSCAL (TRIadic Similarities Ordinal SCALing) provides internal analysis of a set of triadic (dis)similarity measures by a Minkowski distance model using a monotone transformation of the data.

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

Data:	Model:
One-mode	Minkowski distance
Polyadic (triadic)	One set of points
Ordinal	One space
Triad-conditional	Internal
Incomplete	
Replications allowed	

1.1 ORIGIN, VERSIONS AND ACRONYMS

The present program is a revised version of the TRISOSCAL program developed by M.J. Prentice at the University of Edinburgh, which was developed as a generalisation of MINITRI, a program in E.E. Roskam's (University of Nijmegen) MINI series. The original Roskam MINITRI approach is included in the present version as an option (v.i).

1.2 TRISOSCAL IN BRIEF

In a triadic comparison exercise, subjects are presented with sets of 3 objects drawn from a larger collection and asked to judge the relative (dis)similarity of the objects involved. The TRISOSCAL program seeks to represent these dissimilarities as distances between the objects considered as points in a space of minimum dimensionality. The data are considered to be at the ordinal level.

1.3 TRISOSCAL AND OTHER PROGRAMS IN MDS(X)

The TRISOSCAL program makes use of the general MINI-algorithm to deal with triadic data. No other program takes such triadic data but the model involved is the general distance one.

2. DESCRIPTION

2.1 DATA

The fourth quadrant of Coombs's (1964) fourfold typology of data concerns distance information on pairs of pairs. The most obvious method of obtaining directly such data is the so-called method of tetrads in which the subject is presented with all possible combinations of four objects and asked: "which is the most similar/dissimilar pair ?" This method has the disadvantage of requiring a very large number of judgements even on fairly small sets of stimuli. The method of triads while eliciting information on pairs of objects in systematic relation to other objects in the set reduces considerably the number of judgements required of a subject.

2.1.1 The method of triads

Quite simply, the method of triads consists in presenting the subject with all possible triads (but see 2.3.3). (S)He is asked to consider the three possible pairs formed by the triad ABC, namely (A,B), (B,C) and (A,C) and to state either

"which is the most similar pair of these three ?"

or

"which is the most similar pair and which the least similar pair of these three ?"

The first method yields only a partial ordering on each triad in that we know only that for any triad A, B, C only that (A,B) is more similar than (B,C) and than (A,C). The latter case by contrast produces a strict ordering since if the subject chooses (A,B) as the most similar and (B,C) as the least similar, then the order of the three pairs in terms of similarity is necessarily (A,B) (A,C) (B,C).

If the first method has been used in obtaining the data then the user should specify ORDER (0) on the PARAMETERS card. If the method producing a strict ordering has been used then ORDER (1) should be specified.

2.1.1.1 Presentation of the data

The number of objects to be positioned as points in the space is specified on the N OF STIMULI card, the number of actual triads presented to the program on the N OF TRIADS card.

Each object is labelled by a number and thus each triad consists of three numbers, say (5, 2, 4) which are interpreted in the following way.

2.1.1.1.1 When ORDER (0)

The pair which is chosen as the most similar is designated by the first pair of numbers of the three. Thus in our example the pair (5,2) is that chosen.

If the subject has been asked which pair is the most dissimilar then the pair chosen should again be the pair defined by the first two numbers, but in this case the parameter DATA TYPE should be given the value 1 on the PARAMETERS card.

2.1.1.1.2 When ORDER (1)

When the subject has been asked to choose both the most similar and the least similar pair, then the triad is interpreted in the following way.

The first pair of numbers defines the pair chosen as the most similar. The pair consisting of the first and last number is that chosen as the least similar the pair consisting of the second and third numbers is thus the "middle" pair. Thus for the triad 5,2,4 the pair (5,2) is the most similar, the pair (2,4) the next most similar and the pair (5,4) the least similar.

By specifying DATA TYPE (1) on the PARAMETERS card the data are interpreted as dissimilarities rather than similarities.

2.2 THE MODEL

Roskam (1970) has shown that aggregating triadic data by a simple vote-count procedure (counting the number of times that pair jk is judged more similar than pair ℓm) not only obscures but can positively distort the order information in the data especially when not all triads are presented. Rather than the simple vote-count, he suggests that each point j be assigned a sub-matrix, whose row- and column-elements correspond to pairs in which j occurs. Within these it is possible to use the vote-count method. Each of these matrices is represented as a row of a new rectangular asymmetric matrix whose row-elements correspond to the objects and whose column-elements, although labelled as objects refer to the pair formed by the column-element with the particular row-element.

This matrix forms the basis of the analysis but is treated in two different ways by the two STRESS approaches (v.i.). The "local" approach treats the matrix as row-conditional and continues the analysis as in MINICPA while the "global" approach does not enforce this conditionality.

2.2.1 The Algorithm

1. An initial configuration is generated or one is supplied by the user (see 2.3.2).

2. The distances in the configuration are calculated according to the Minkowski metric chosen (see 2.3.1).

3. The fitting values are calculated (see 2.2.2).

4. STRESS is calculated according to the option chosen (see 2.2.2).

5. A number of tests are performed: e.g.

   Has STRESS reached an acceptable minimum ?

   Has a specified number of iterations been performed ?

   Has the improvement in STRESS over the last few iterations been too small to warrant continuing ?

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

6. The direction in which each point should move in order that STRESS should decrease as well as the estimated optimum size of that movement are calculated.

7. The configuration is moved in accordance with 6 and the program returns to 2.

2.2.2 Fitting-values and STRESS

At each iteration a set of fitting values is calculated which are constrained to being in the same order as the dissimilarities implied in the data. These fitting values are used to calculate the value of STRESS which is an index of how well the particular configuration matches the data. Two methods are available within TRISOSCAL for making this calculation - Roskam's "local" approach and Prentice's "global" approach.

2.2.2.1 The "Local" approach

This is the approach used exclusively in the original Roskam MINITRI program. Fitting values are assigned to pairs of points (stimuli) so that the order of the fitting-values matches the order of dissimilarities within each triad. Each inversion of that order will lead to an increase in the value of STRESS. In this method no account is taken of inversions of order occurring between triads. Consequently, the same datum can be fitted by different fitting values in different triads.

2.2.2.2 The "Global" approach

Consider the following two triads: (ABC) and (BCD). In the "local" approach the program is free to assign to the one pair (B,C) which occurs in both triads two distinct fitting values without affecting the value of STRESS. The "global" approach forces the program to assign the same fitting value. This has the effect of requiring that the order of fitting values be kept across the whole set of stimuli. This option is chosen by specifying STRESS(1) on the PARAMETERS card.

Since this latter method obviously imposes far greater constraints on the solution than the "local" approach, the values of STRESS obtained will be considerably higher. The "local" procedure ignores transitivity between triads and thus it is often advisable to use this option if the data have been collected from a large number of subjects.

2.3 FURTHER FEATURES

2.3.1 Distances in the configuration

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 TRISOSCAL 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).

2.3.2 The initial configuration

It is not possible to generate an initial configuration directly from the triadic data. However, as a vote count matrix is formed (section 2.2), this is used to generate an initial configuration in the same way as the Guttman-Lingoes-Rosham MINI programs. This configuration uses only the ordinal properties of the vote count matrix and has certain desirable properties such as avoiding local minima.

If one user wishes to supply an initial configuration then this is input via the READ CONFIG card and its associated INPUT FORMAT. The configuration must be in the maximum dimensionality to be used in the solution. The parameter MATFORM is used to specify how the input configuration is punched and is detailed in section 3.1.

2.3.3 Balanced incomplete block designs

Even with the method of triads the number of judgements required of subjects, increasing as it does with the cube of the number of stimuli, rapidly becomes unmanageable. Balanced incomplete block designs are designs which reduce this number, while ensuring that certain desirable conditions (such as ensuring that every possible triad is presented at least once) are met. These are described in Burton and Nerlove (1976).

3. INPUT PARAMETERS

3.1 LIST OF PARAMETERS

Keyword	Default Value	Function
DATA TYPE	0	0: Input data are similarities 1: Input data are dissimilarities.
MINKOWSKI	2.0	(Any positive number) sets the Minkowski parameter for determination of distances in the configuration.
ORDER	0	0: Partial order is input 1: Full order is input (section 2.1)
STRESS	0	0: STRESS calculated using "local" approach. 1: STRESS calculated using "global" approach (see 2.2.2).
MATFORM	0	0: Input configuration is input dimensions (rows) by stimuli (columns). 1: Input configuration is input stimuli (rows) by dimensions (columns) (only if READ CONFIG is used).

3.2 NOTES

1. The N OF TRIADS card, having the same form as the N OF STIMULI card, is mandatory in TRISOSCAL.

2. N OF TRIADS may be replaced by N OF SUBJECTS.

3. Program Limits:

Maximum number of stimuli allowed by the program is	50
Maximum number of triads allowed by the program is	3333
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 TRISOSCAL, the available options are as follows:

3.3.1 PRINT options

Option	Form	Description
INITIAL	p × r	The co-ordinates of the points in the initial configuration are printed.
FINAL	p × r	The solution matrix, the co-ordinates of the stimulus points in the final configuration are printed.
DISTANCES	p × p (lower triangle only)	The matrix of interpoint distances in the final configuration is printed.
FITTING	p × p (lower triangle only)	The matrix of fitting values is printed.
RESIDUALS	p × p (lower triangle only)	The matrix of residuals (distances-fitting values) is printed.
HISTORY		A detailed history of the iterative process is printed.
COUNT	p × p (lower triangle only)	The vote-count matrix as derived from the triadic comparisons is printed.
GRADIENT	p × r	The matrix at gradients as applied to the final configuration is printed.

By default only the final configuration is printed.

3.3.2 PLOT options

Option	Description
INITIAL	The initial configuration is plotted as r(r-1)/2 two-way plots.
FINAL	The solution is plotted as r(r-1)/2 two-way plots.
SHEPARD	The Shepard diagram of data against distances is plotted.
POINT	A histogram of the contribution to STRESS of each point is plotted.
RESIDUALS	A histogram of residual values is produced.
STRESS	A histogram of the STRESS values at each iteration is produced.

By default only the Shepard diagram and the FINAL configuration are plotted.

3.3.3 PUNCH options

Option	Description
FINAL	The solution configuration is output, indexed in a fixed format.
SPSS	The following are output in a fixed form I = row index J = column index VOTE = entry in votecount matrix correspondin to I,J DIST = the corresponding distance FITTING = the corresponding fitting value RESID = the corresponding residual value
STRESS	An iteration by iteration history of STRESS values is punched according to a fixed format.

4. EXAMPLES

```
col 1                col 16
-------------------------------------------------------------
RUN NAME             SOME DATA FOR TRISOSCAL
N OF STIMULI         10
N OF TRIADS          120
DIMENSIONS           2 TO 3
PARAMETERS           MINKOW(1), ORDER(1), STRESS(1)
INPUT FORMAT         (12I3)
READ MATRIX

  <data follow here>

PRINT                COUNT
PLOT                 SHEPARD, POINT(3)
COMPUTE
FINISH
-------------------------------------------------------------
```

BIBLIOGRAPHY

Burton, M.L. and S.B. Nerlove (1976) Balanced designs for triads tests: two examples from English, Soc.Sci.Res., 5, 247-67.

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

Coombs, C.H. (1964) A theory of data, New York: Wiley.

Prentice, M.J. (1973) On Roskam's nonmetric multidimensional scaling algorithm for triads, Edinburgh, Project on Occupational Cognition, mimeo.

Roskam, E.E. (1969) Data theory and algorithms for non-metric scaling, Department of Psychology, University of Nijmegen, mimeo.

Roskam, E.E. (1970) The method of triads for nonmetric multidimensional scaling, Nederlands Tijdschrift voor de Psychologie, 25, 404-7.

Roskam, E.E. (1975) Non-metric data analysis, Department of Psychology, University of Nijmegen, Report 75-MA-13.

APPENDIX 1:

There are no other programs widely available for the analysis of triadic data.

APPENDIX 2:

This appendix is based on Roskam (1971) which is used with permission.

Let

$$Q = \{q_j\} \qquad j = 1,\ldots,p$$

be a set of objects and let s_{jk} be the perceived similarity of pair j,k. (We assume $s_{jk} = s_{kj}$ apart from random error.

Let

$$g_h = \{\langle q_i, q_j\rangle, \langle q_j, q_k\rangle, \langle q_i, q_k\rangle\} \quad (i\neq j\neq k)$$

be the set of $\binom{p}{3} = p(p-1)(p-2)/6$ possible triads $(h = 1,\ldots,\binom{p}{3})$.

We let m = the number of triads which may or may not be $\binom{p}{3}$ if replications or omissions occur.

We rank s_{jk} by δ

$$\delta_{jkh} \leq \delta_{j\ell h} \quad \text{if} \quad s_{jkh} \geq s_{j\ell h}$$

using subscript h to indicate s_{jk} at triad h.

Further let Δ_h denote δ_{jkh}, $\delta_{k\ell h}$ and $\delta_{j\ell h}$ such that

$$\Delta_h = \langle j,k,\ell\rangle \rightarrow \delta_{jkh} \leq \delta_{k\ell h} \leq \delta_{j\ell h}$$

We seek

$$D = \{d_{jkh}\}$$

and

$$\underset{\sim}{D}^{o} = \{d^{o}_{jkh}\}$$

such that

$$d_{jkh} = d_{kjh} = \sqrt[u]{\sum_{a} (x_{ja} - x_{ka})^{u}} \qquad \begin{matrix} a = 1,\ldots,r \\ u \geq 1 \end{matrix}$$

and $d_{jkh} = d_{jkf}$ for all $h,f = 1,\ldots,m$

such that

$$\underset{\sim}{X} = \{x_{ja}\}$$

is a matrix of co-ordinates

Also

$$d^{o}_{jkh} \leq d^{o}_{k\ell h} \leq d^{o}_{j\ell h} \quad \text{if } \delta_{jkh} \leq \delta_{k\ell h} \leq \delta_{j\ell h}$$

We define a loss function

$$L = \frac{\sum_{jkh} (d_{jkh} - d^{o}_{jkh})^{2} \ell_{jkh}}{\sum_{jkh} d^{2}_{jkh} \ell_{jkh}}$$

as a measure of badness of fit between data (fitting values) and solution.

A2.1 The Algorithm

The procedure is double-phase and iterative with the stages indexed by superscript s. The first phase is itself iterative and is indexed by superscript t. We aim to minimize L.

The process is essentially the same as that in MINISSA. The general formulation of the iterative solution is

$$x_{ja}^{s+1} = \left[\frac{1}{(1-L^2) \sum w_{jka}\, n_{jk}} \sum \gamma_{jka}\, x_{ka} \right]^s$$

where

$$w_{jka} = \left(\frac{| x_{ja} - x_{ka} |}{d_{jk}} \right)^{u-2}$$

n_{ij} = no. of occurrences of (j,k) in the data

which is equivalent to

$$x_{ja}^{s+1} = x_{ja}^{s} - \alpha_a^s \left(\frac{\partial L}{\partial x_{ja}} \right)^s$$

where

$$\alpha_{ja}^{s} = \left(\frac{\beta \sum_{jk} n_{jk}\, d_{jk}^2\, L}{(1-L^2) \sum_{k} w_{jka}\, n_{jk}} \right) \qquad (\beta = 1)$$

The convergence process may be speeded up by defining

$$\beta = 4(\cos \xi^s)^3 \sqrt[3]{\beta^{s-1}}$$

where

$$\cos \xi^s = \frac{\sum_{ja} g_{ja}^{s}\, g_{ja}^{s-1}}{\sqrt{\sum_{ja} (g_{ja}^2)^s \sum_{ja} (g_{ja}^2)^{s-1}}} \qquad (s \geq 1)$$

The secondary process uses either Guttman's rank-image procedure which will decrease but not minimize L. This is used in the earlier interations. We then switch to Kruskal's monotone regression procedure.