NUMERICAL CLASSIFICATION SUITE

CLUSTR

C.J. ANDREWS

UNIVERSITY OF QUEENSLAND
COMPUTER CENTRE

A NUMERICAL CLASSIFICATION SUITE

CLUSTR

C.J. ANDREWS

ABSTRACT

This manual describes a suite of programs which are capable of dealing effectively with sets of data which are to be numerically classified. The data represent several entities which are describes by relevant attributes.

The method by which the classification is performed may be controlled in a most flexible manner, by several easily set user options. These options control the following steps in the classification process:

(i) A transformation of the raw data may optionally be carried out in one of several ways.
(ii) A choice of dissimilarity indices may be made.
(iii) A choice of sorting and clustering strategies is available.
(iv) Output optionally available includes printouts of trellis diagrams, two way tables and summaries of the raw data, and plots of derived dendrograms from the sorting strategies.
(v) optional Ordination derived from the methods of Principal Component Analysis and/or Principal Coordinate analysis, may be selected.

The program as outlined performs both normal and inverse analyses of two-dimensional raw data in the form of entities versus attributes. Such data are commonly generated in psychological, taxonomic, and biological studies and also in studies in other social sciences.

CLUSTR can also be used to classify three dimensional data (entity-1 x entity-2 x attribute) as is often required in ecological study, for example in <sites x times x species> analysis. This extension in no way affects the two-dimensional study of data, and is entirely transparent to users of the latter facility.

An extension of the known TAXAN program has been accomplished, and the output options of CLUSTR may be coupled with the ability of TAXAN to handle disordered multistate data. This facility is useful for taxonomists.

ACKNOWLEDGEMENTS

The author gratefully acknowledges the help of several people in the preparation of these programs. Prof. W. Stephenson provided the initial motivation and need for the suite, and over a long period has provided the author with many helpful discussions and suggestions as to its functioning. Dr. H.T. Clifford has contributed to the author's thinking in many areas of classification and is similarly acknowledged.

Dr. Clifford provided access to Dr. E. Burr's program TAXAN which forms the basis of the sorting section of CLUSTR. It has been modified with the help of Mr. R.A. Cook of La Trobe University. Mr. R.D. Nilsson provided certain ideas for flexible core management which the author has subsequently incorporated into CLUSTR.

The University of Queensland Prentice Computer Centre has provided finance for the latest stage of development of this suite and is also gratefully acknowledged.

SUPPORT

Support for this suite is available in two forms.

Users who desire advice on classificatory methodology may consult Dr. H.T. Clifford of the Botany Department, University of Queensland. Such users should first have gained familiarity with the methods, via one of the standard texts (e.g. 1,2)

Those who desire some assistance with the construction and running of CLUSTR data decks may consult the University Computer Centre via its user consultation services. An attempt should first have been made to set up the deck before consultation, as shown in this manual.

Contents

Part 1

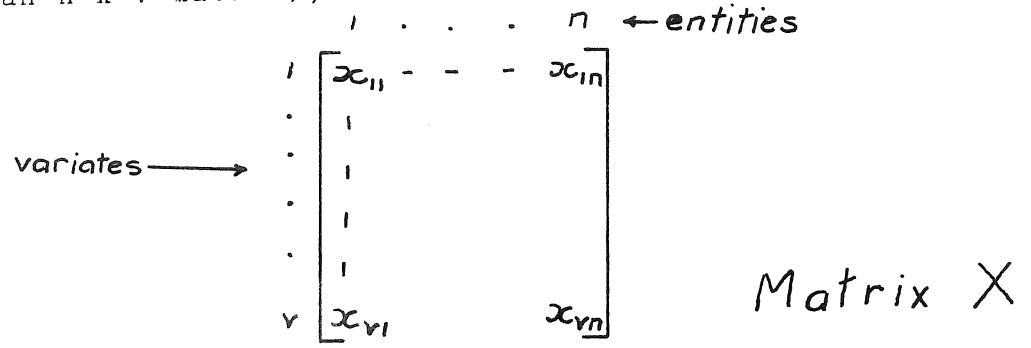
The Analysis of Two Dimensional (2D) Data

1.0 The Classification Process - an overview

Throughout this part it is assumed that a user has a data matrix that he wishes to "have classified" and that he has some basic knowledge of what sort of classification he wishes to have performed on that matrix. Those users who desire more information on the theory of numerical classification itself are referred to one of the standard texts on the subject (e.g. 1,2). It will be the object of this section (section 1.0) to define terminology used with regard to this program, and set the stage for describing the options this program provides at each stage of its execution.

An examination is now made of the processes which are carried out on a set of data in classifying it.

It is assumed that n entities are described by the values of v variates. The data for each of the entities can be set out in a matrix X (an n x v matrix), viz:

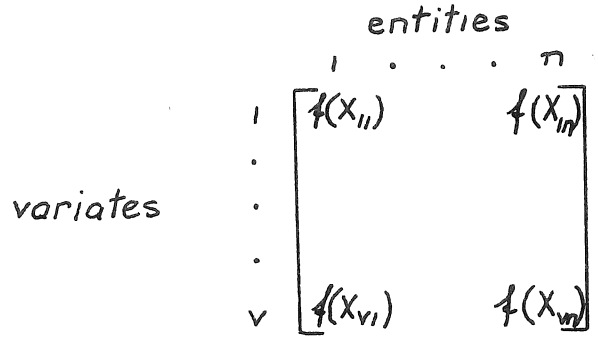


In this matrix Xij represents the value of the ith attribute(variate) of the jth entity.

Taking into account all v variates for each entity it is possible to produce (n x n) indices of dissimilarity between all the possible pairs of entities. It may not however always be desirable to produce these indices from the raw data and so a new matrix F (n x v) can first be produced such that each element of F is a TRANSFORMED version of the elements of X (i.e. as in matrix F below)

* * * * * * * * * * *

1. Clifford,H.T. Stephenson, W. "An Introduction to Numerical Classifiction" Ac.Pr. 1975
2. Sneath,P.H.A., and Sokal,R.R., "Numerical Taxonomy", Freeman, 1973.



Matrix F

The transforming function could be defined for example as

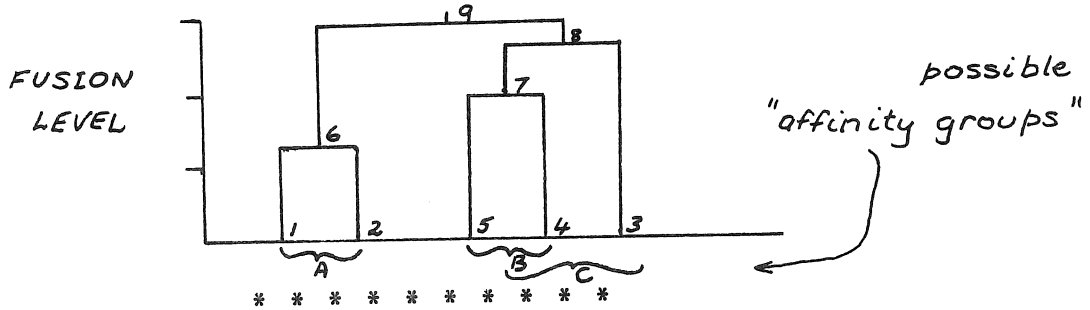
f(Xij)=log(Xij+1)

or f(Xij)=Xij i.e. operate on raw data

From matrix F can now be produced the matrix D, a square nxn matrix whose i,j th element Dij represents the dissimilarity between elements i and j of the F matrix. This matrix has elsewhere been termed the Q matrix (3); Also the term TRELLIS DIAGRAM has been applied.

The trellis diagram can now be "sorted" so that elements of closest affinity can be brought together progressively in clusters. This process has variously been termed sorting, clustering, grouping or fusing. The means of determining which element of D is next fused into an existing group is called the "sorting strategy". Initially each entity is regarded as a group of size 1. The sorting strategy therefore operates on the D matrix, and successive fusions progressively reduce the size of the matrix. What is produced in essence is a re-ordering of the rows (or columns) of the D matrix.

Two factors are noted when entities fuse with existing groups (which may possibly be other single entities). These are, the entities which fuse, and the dissimilarity level at which they fuse. The result of all fusions for one trellis diagram can be expressed in terms of these two factors in a DENDROGRAM e.g.



3. Gower, T.C., "Some Distance Properties of Latent Root and Vector Methods Used in Multivariate Analysis". Biometrika 53, 325-388 1966

In the above case entities 1 and 2 have fused to form group 6, 5 and 4 to give 7, and entity 3 has fused with group 7 to form a larger group 8. Similarly, 6 and 8 fuse to give 9 and the process is complete at this stage since one cluster alone finally exists.

The base line of this dendrogram can be used to give an ordering of the initial entities into "groups of closest affinity" at various levels of affinity (see for example groups B and C above).

It is now possible to rearrange the columns of the original X matrix to reflect this ordering and in so doing the grouping of entities should become visually more obvious in that matrix.

So far this general discussion has centred on classifying entities against each other taking account of all possible variates. It is also posible to perform the so called inverse analysis classifying attributes against each other. In this case it is possible, may be desirable, but is not mandatory to use a different transformation, different dissimilarity index, and different sorting strategy than was used for the "normal" analysis. The end products, however, are the same - a dendrogram specifying in this case attribute correlations, and a resorting this time of the rows of the X matrix.

When both resorting of rows and columns of the X matrix occurs, a so called TWO-WAY table is produced.

Once a trellis diagram has been obtained, it is possible to perform a principal co-ordinate analysis of this diagram to obtain a representation of the n entities or v variates in a space which can be interpreted geometrically. The clusters may then be viewed visually. The method by which this can be done is elsewhere described and is quite complex. For details of its derivation the user is referred to references (3) and (4).

Principal Component Analysis may also be performed, and by this alternate method the n entities may be represented in a v-dimensional space. The latter analysis operates on the original F matrix, and is equivalent to a rotation of the original v axes against which the entities of F are represented, to new orthogonal positions which maximise attribute variance along their length.

The two Principal Analyses are members of a group of analyses known under the general title of Ordination.

* * * * * * * * * *

4. Gower, J.C., "Multivariate Analysis and Multidimensional Geometry", Statistician, 17,13-28 1967

The CLUSTR program provides users with all the features outlined above, i.e. transformation, dissimilarity calculation, sorting, production of two-way tables, and ordination, for both normal and inverse analyses, with possible differences in strategy for each.

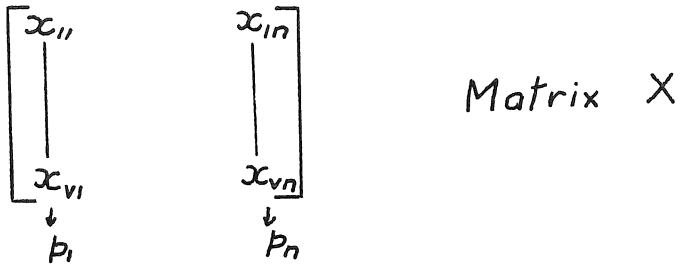
The various different strategies available will be defined in Section 2.0 and how they may be specified in running the program will be shown in Section 3.0 of this manual.

2.0 CLUSTR operations for 2D data

The various options available to a user at each level of the clustering process are now outlined. It is assumed at this stage that the raw data matrix has been read from an input medium. The method by which this is done will be given in Section 3.0 and 8.0.

2.1 Summary of Raw Data

An initial summmary of the raw data is printed in 2D analysis. For an X matrix of the following form,



a quantity Pi is derived which may be either
(a) the total of all variate values for that entity.

$$Pi = \sum_{j=1}^{v} Xji \qquad \text{[later termed S]}$$

or (b) the average variate value for that entity

$$Pi = \frac{\sum_{j=1}^{v} Xji}{v} \qquad \text{[later termed A]}$$

2.2 Transformation Option

The transformation option available for each element of the X matrix (F(Xij) of 1.0) may be:

either (i) F(Xij)=log(Xij) log transform [L]

or(ii) F(Xij)= Xij no tranformation [N]

$$\text{or(iii)}\quad F(Xij) = \frac{Xij}{\sum_{k=1}^{n} Xik}$$ (standardize by row total for normal analysis) [S]

$$F(Xij) = \frac{Xij}{\sum_{k=1}^{v} Xkj}$$ (standardize by column total for normal analysis) [S]

or(iv) F(Xij) = Xij ** (1/n) where n is integral [P]

or(v) F(Xij) = (Xij-m)/s [V]
where m and s are the mean and standard deviation in rows or columns.

2.3 Dissimilarity Index Option

The current dissimilarity indices allowed are:

(i) Bray-Curtis [B]

$$dij = \frac{\sum_{k=1}^{v} |Fki-Fkj|}{\sum_{k=1}^{v} |Fki+Fkj|}$$ (For normal analysis)

$$dij = \frac{\sum_{k=1}^{n} |Fik-Fjk|}{\sum_{k=1}^{n} |Fik+Fjk|}$$ (For inverse analysis)

where Fij is the element found in the transformation step as F(Xij).

(ii) Manhattan Metric [M]

$$dij = \sum_{k=1}^{v} |Fki-Fkj|$$

or $$dij = \sum_{k=1}^{n} |Fik-Fjk|$$ for analyses as before

(iii) Canberra Metric [C]

$$dij = \frac{1}{v} \sum_{k=1}^{v} \frac{|Fki-Fkj|}{|Fki+Fkj|}$$

or $$dij = \frac{1}{n} \sum_{k=1}^{n} \frac{|Fik-Fjk|}{|Fik+Fjk|}$$

for analyses as before

(iv) D-squared Euclidean Distance [D]

$$dij = \sum_{k=1}^{v} (Fki-Fkj)^2$$

or $$dij = \sum_{k=1}^{n} (Fik-Fjk)^2$$

for analyses as before

(v) Matching Coefficient [A]

$$dij = \frac{\sum_{k=1}^{v} (Wij)_k}{v}$$

or $$dij = \frac{\sum_{k=1}^{n} (Wij)_k}{n}$$

where (Wij) =1 if elements i and j have equal values for Fik and Fjk. Thus both negative and positive matches are considered.

(vi) ratio coefficient [R]

$$dij = v - \sum_{k=1}^{v} \left(\frac{(Fik)}{(Fjk)} \right)^{g}$$

or

$$dij = n - \sum_{k=1}^{n} \left(\frac{(Fki)}{(Fkj)} \right)^{g}$$

where g is chosen to be either +1 or -1 to make the fraction less than or equal to unity.

General Note on Dissimilarity Measures

It will be seen later that missing data may be specified in the CLUSTR data input deck. If any of the elements of F specified above are missing for particular items in the summations then those summation items contribute nothing to the summations.

2.4 Sorting Strategies

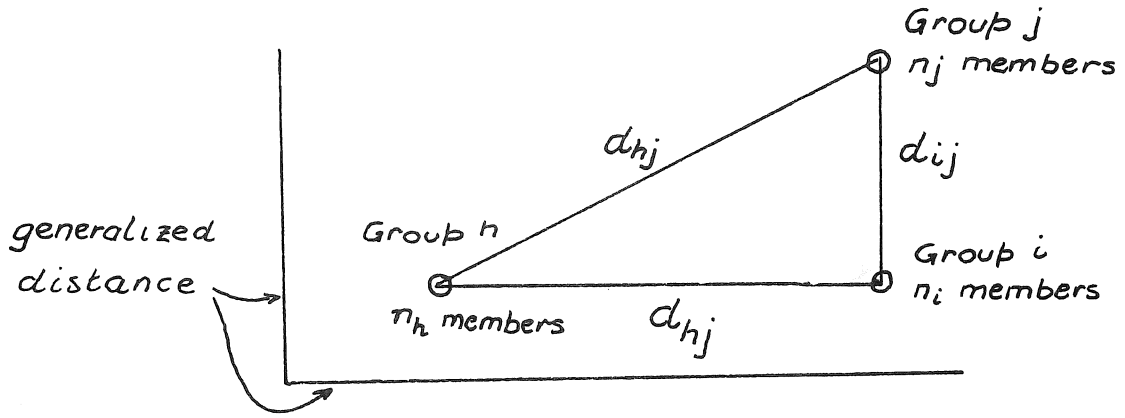
Several commonly used sorting strategies are available in this program suite. The point at which a sorting strategy becomes identifiable is that point at which the dissimilarity index is recalculated between a new group and all other groups after the former group has received a new member. Lance and Williams (5) have shown that some common strategies (at least most of those described here) could be expressed in terms of the following general form:

$$d_{hk} = \alpha d_{hi} + \alpha d_{hj} + \beta d_{ij} + \gamma |d_{hi} - d_{hj}|$$

where the choice of $\alpha_i, \alpha_j, \beta$ and γ determine the strategy. The following diagram defines the other terms in the expression.

* * * * * * * * * *

5. Lance, G.N. and Williams, W.T., "A Generalised Sorting Strategy for Computer Classifiations" Natre (Lond.) 212,218 1966



The strategies allowed in CLUSTR are defined in terms of these parameters as follows:

(i) Nearest Neighbour

$$\alpha_i = \alpha_j = \frac{1}{2} \qquad \beta = 0 \qquad \gamma = -\frac{1}{2}$$

(ii) Furthest Neighbour

$$\alpha_i = \alpha_j = \frac{1}{2} \qquad \beta = 0 \qquad \gamma = +\frac{1}{2}$$

(iii) Group Average

$$\alpha_i = \frac{n_i}{n_h} \qquad \alpha_j = \frac{n_j}{n_h} \qquad \beta = \gamma = 0$$

(iv) Simple Average

$$\alpha_i = \alpha_j = \frac{1}{2} \qquad \beta = \gamma = 0$$

(v) Centroid

$$\alpha_i = \frac{n_i}{n_h} \qquad \alpha_j = \frac{n_j}{n_h} \qquad \beta = -\alpha_i \alpha_j \qquad \gamma = 0$$

(vi) Incremental Sum Of Squares
Not of the Lance and Williams type previously mentioned. The user is referred to Burr(6).

* * * * * * * * * *

(6)Burr, E.J., 'Cluster Sorting with Mixed Character Types. II Fusion Strategies' Aust. Comp. Jour. 2,98-103 1970.

(vii) Variance
Again not of the Lance and Williams type; see(6).

(viii) Flexible

$$\alpha_i = \alpha_j = .625 \quad \beta = -.25 \quad \gamma = 0$$

2.5 Other Options

At each stage of the classification process, the user may optionally select output of various results. These include trellis diagrams, dendrograms and two way tables.

2.6 Ordination

Ordination is treated as a slightly separate topic as it may proceed fairly independently of other analyses. Once trellis diagrams have been obtained, Principal Coordinate analysis may proceed, and once transformations have been carried out (i.e. matrix F options have been specified) Principal Components Analysis may be performed.

2.6.1 Method of Principal Coordinate Analysis

The method used for Principal Coordinate Analysis is that given by Gower(4) as modified by Williams and Dale(7).

Given a dissimilarity matrix D of elements Dij the procedure is as follows:

1 (a) For Bray Curtis trellises form the similarity matrix Sij=1 - Dij
(b) For Manhattan Metric trellises form the matrix Sij=(-1/2)*Dij**2

2. "Transform" the elements by Sij <= Sij - S.j - Si. + S.. where . represents the mean over the appropriate row or column

3. Find the eigenvectors and eigenvalues of the matrix (Sij) and standardise each vector so that the sum of squares of its elements is equal to the corresponding eigenvalue.

4. The k th elements of each of the n vectors then represents the co-ordinates of the kth point with respect to axes defined by the n vectors. The relative magnitude of each eigenvalue gives the

* * * * * * * * * *

7. Dale, M.B. Personal Communication

relative "importance" of each eigenvector axis.

2.6.2 Principal Component Analysis

The method used for Principal Component Analysis is that outlined by various authors e.g. Seal(8),Blackith and Reyment(9), and Sneath and Sokal(2).

Given a matrix F the method is

1. Form the matrix, R, of variance and covariance between entities or attributes.
2. The eigenvectors of R give the Principal Components we are seeking.
3. Standardize the eigenvectors so that they are of length 1, giving matrix V.
4. The matrix of new coordinate points P of the entities represented by F is given by $P=V'F$.

2.7 Conclusion

Section 2 has given the formal definition of each of the facilities available in CLUSTR. How to specify these to the running program will be examined in Section 3.

* * * * * * * * * *

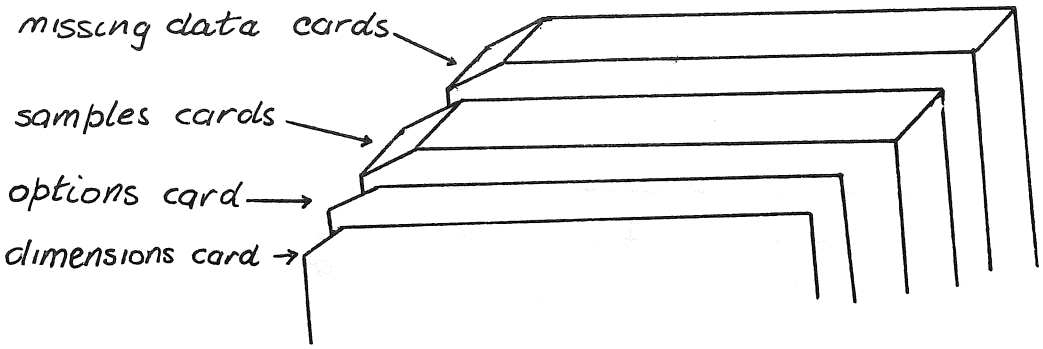
(8) Seal, H.L., "Multivariate Statistical Analysis For Biologists", Methuen, London, 1964.
(9) Blackith,R.E., and Reyment, R.A., (eds), "Multivariate Morphometrics" , Ac. Pr., N.Y., 1971.

3.0 Data Input Format For 3D Data

To run the CLUSTR program, a user codes his data onto cards and forms a data deck. The data deck contains four types of card in the manner of 3.1.

3.1 Data Deck Structure

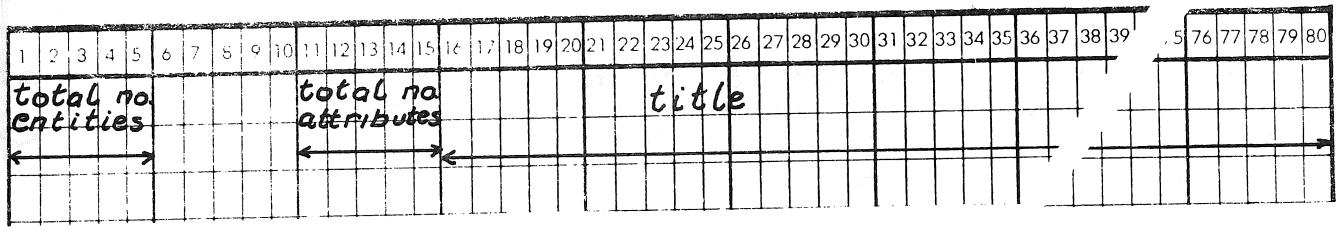
The data deck consists of four types of cards placed in the following order:



Each of these cards will be discussed in the following sections. The samples card and dimensions card will be outlined first as they are least complex. The options card will next be detailed, and the section concluded with details of the missing data cards.

3.2 Dimensions Card

The dimensions card contains the maximum dimensions of the matrix to be input. The format is:



The title may be up to 65 characters in length and serves to identify all subsequent output.

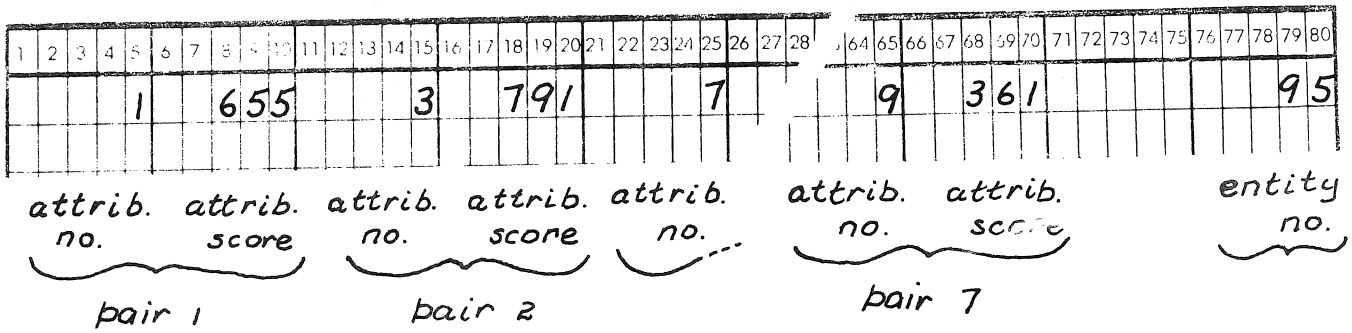
3.3 Samples Cards

The samples cards contain the data for the n x v attribute matrix X.

There are two different forms for the input which are selectable depending on which the user finds most convenient. The forms are either to input the data by entity, or by attribute, i.e. row-wise or column-wise in the X matrix.

Entity format:

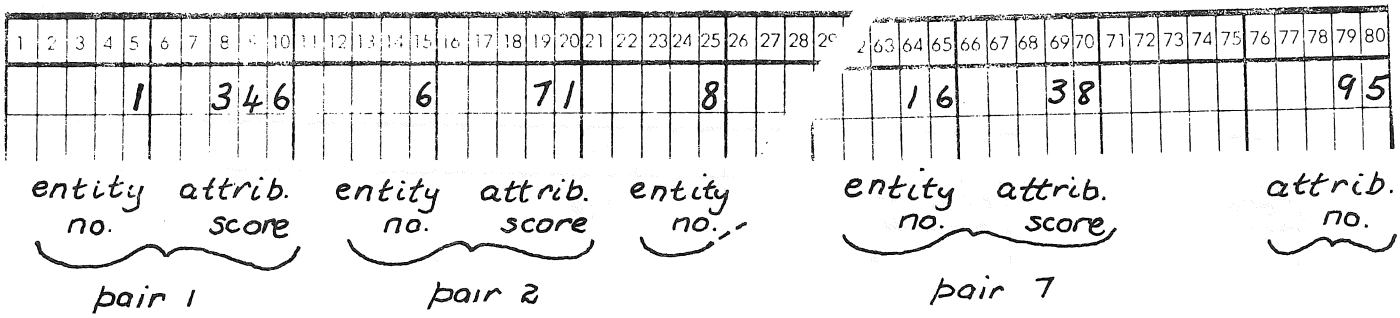
A user specifies the various attributes for a given entity together with an attribute score. The form is



More than one card for a given entity is permissable.

Attribute Format:

A user specifies the scores for several entities for one given attribute per card. The form is

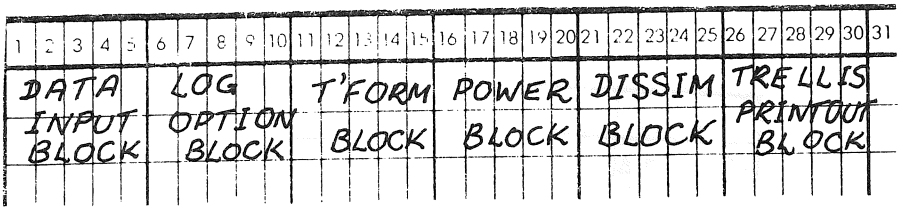


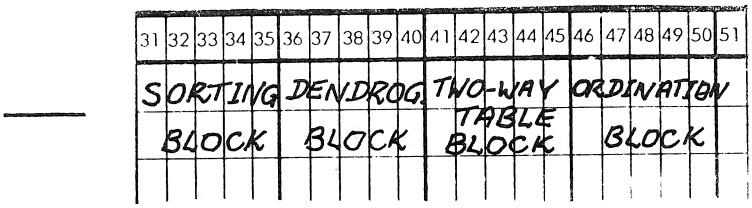
More than one card per attribute is permissable.

3.4 Options Card

The options card specifies the manner in which the analysis is to be performed. It contains codes to indicate to the program which of the particular facilities in section 2.0 a user wishes to specify at each stage of his analysis.

It is organized into 10 blocks each of which contains 5 columns, as follows:





Each block declares the optins required for an identifiable stage in the analysis. The form of the blocks will next be given.

3.4.1 DATA INPUT BLOCK

The data input block contains three fields which signify the form of the input deck following.

(a) Samples Card Format
Column 1 contains an option which specifies which format the samples cards will appear in:
E - entity format
A - attribute format

(b) Divide Option

Columns 2-4 contain a numeric constant which will be used to divide into each incoming sampled number. If left blank no division is performed. This allows the user to

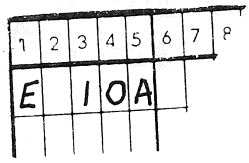
scale his input deck for example 10 or 100.

(c) Listing Option

The program initially lists summary totals or averages as previously defined. Column 5 contains one of the following:

S - summed values
A - average values

A sample form is:

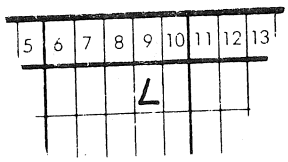


3.4.2 Logarithm Option Block

While the transformation of log (Xij + 1) may more logically appear to be included in the transformation block (3.4.3) it is included in a separate block as this will provide greater flexibility for 3D analysis extension later described.

If a log (Xij + 1) transformation of the complete X matrix is required then the L option is specified in column 9 of the logarithm option block. Further transformation may optionally be applied as previously defined by using options in the transformation block.

A sample form for this block is:

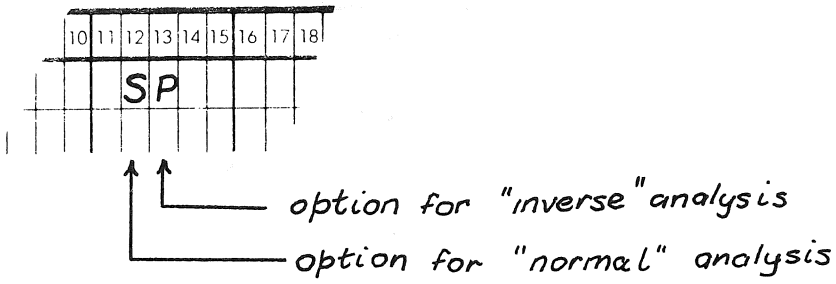


3.4.3 Transformation Block

The transformations of the X matrix (log'ed or not) are specified in columns 11-15 of the options card. The options are:

(i) standardise (by row or column) - S option
(ii) power option - - P option
the power to which each data element is raised is specified in 3.4.4. This option is used for taking the nth root of an element.
(iii) do nothing to the element - - N option
(iv) express as std deviation from mean - V option

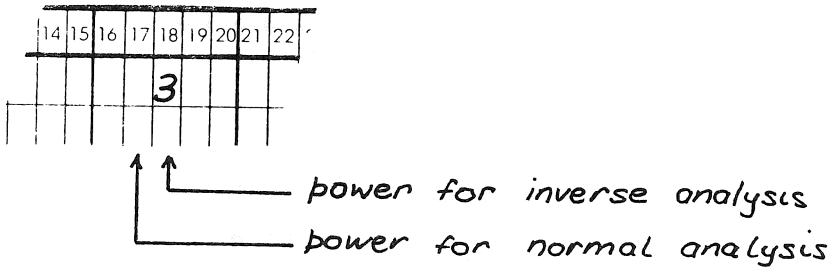
The form of the Transformation block is:



3.4.4 Power Block

If the P option has been used in 3.4.3 the power block specifies the power to which each element is raised. If 3 is placed in the appropriate column each element is raised to the 1/3 power and so on. In general if n is specified the elements are raised to the 1/n power.

The form is:



If the P option is not used, these columns may be left blank.

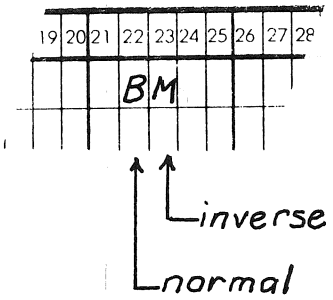
3.4.5 Dissimilarity Block

This block indicates the type of dissimilarity measure to be used in creating the trellis diagrams for later sorting. The

options are:

(i) "B" option - Bray-Curtis measure
(ii) "M" option - Manhattan Metric
(iii)"C" option - Canberra Metric
(iv) "D" option - Euclidean distanc squared
(v) "A" option - Matching Co-efficient
(vi) "R" option - Ratio Measure

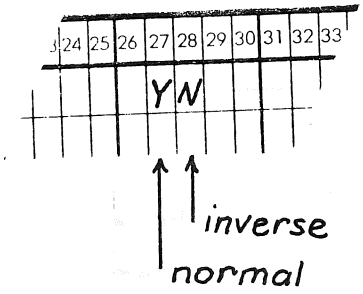
This field may be blank if dissimilarity calculation is not required. The form is e.g.:



3.4.6 Printout Block

This block is a simple Yes/No option block to control the printout of each triangular trellis diagram. The options are:

(i) Y - Yes, print it out
(ii) N - don't print it out.



3.4.7 Sorting Block

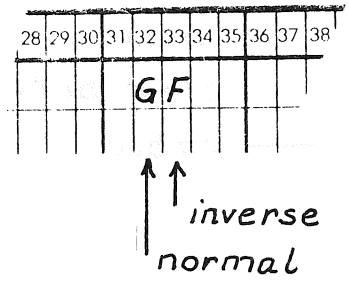
This block indicates the type of sorting algorithm to be used in generating the dendrograms from the triangular matrices. The algorithms are:

Option	
N	nearest neighbour
F	furthest neighbour

G weighted average
A simple average
C centroid
I incremental sum of squares
V variance
X flexible

This field may be left blank if a sort is not required.

A sample form is:

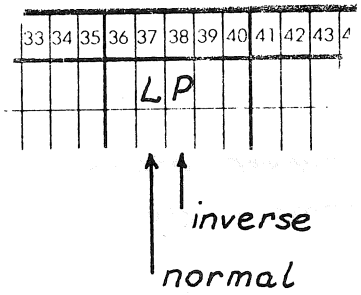


3.4.8 Dendrogram Block

This block controls whether each dendrogram is produced and the form of its output. The options are:

L - produce dendrogram, output is listing of group fusions
P - produce plot of dendrogram
B - produce both plot and listing of dendrogram
N - (or blank column) dendrogram not required

Example:



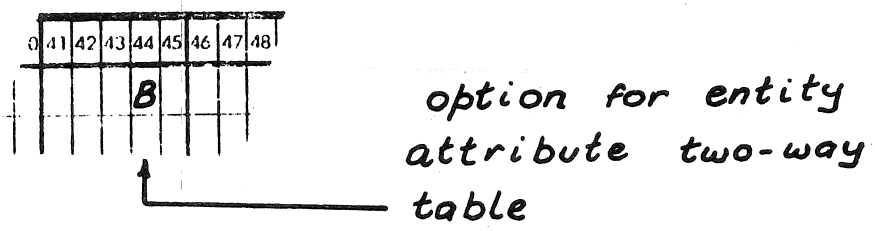
3.4.9 Two Way Table Block

Data which have previously been log-transformed may be de-transformed when printing out a two-way table. The options

which may be specified are:

L - 'leave as is' printout (don't attempt to detransform)
U - 'un-logged' printout (detransform the data)
B - both L and U above
N - (or blank column) not required

Example:

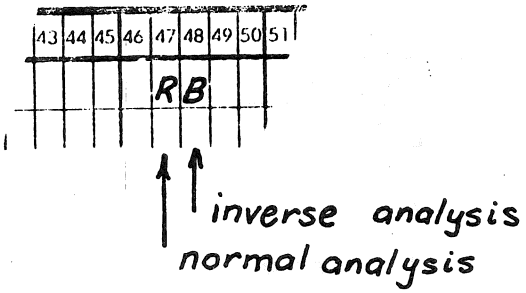


3.4.10 Ordination Block

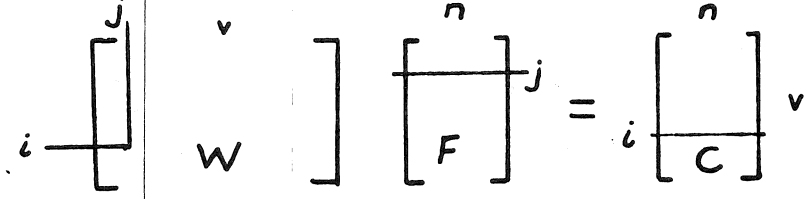
The Ordination block specifies whether or not to produce an ordination analysis of the appropriate trellis diagram, or F matrix. The options available are:

R - produce Principal Coordinate analysis
M - produce Principal Component analysis
B - produce both R and M above
N - (or blank column) ordination not required

Its form is:



In Principal Component analysis the F matrix is multiplied by a weighting matrix, W, to give the matrix of new coordinates, C, i.e.



Using the normal notation of matrix algebra, it will be seen that the element Wij represents the weighting placed on old axis j in the F data, in its contribution to the score on the new axis i in the C data.

The W matrix is automatically printed out in Principal Component analysis.

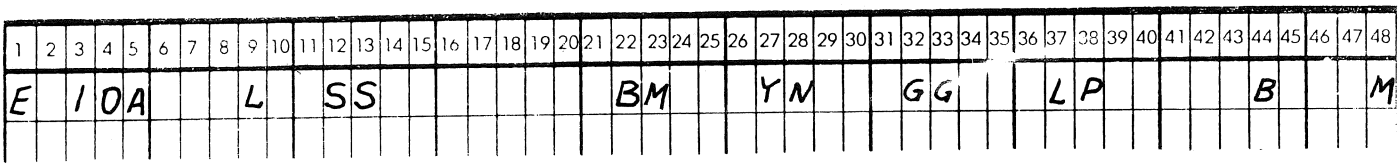
It should be noted that the analyses of large matrices for eigenvectors and eigenvalues involves extensive calculation, and may therefore be expensive to perform.

3.4.11 Note

If any particular option is not required for the user's particular analysis it may be left blank. An interpretation of the options card is printed out at the beginning of the output to serve as a check for the user.

3.4.12 Examples

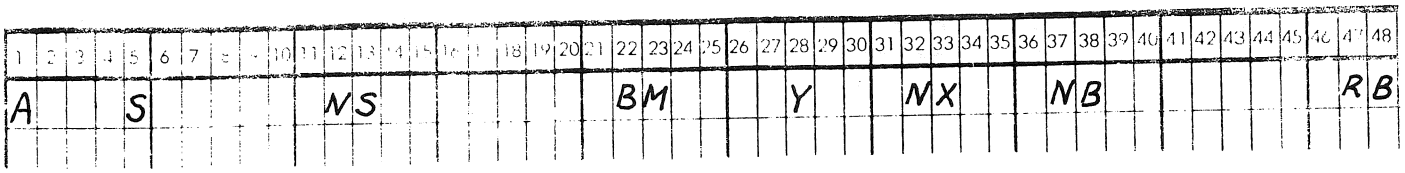
The following are sample options cards for illustrative purposes:



Example 1 above specifies

i input deck in entity form to be divided by 10
ii initial listing to be averaged values
iii log transform the data
iv standardize the logged data in columns by total, for entity analysis
" " " " " rows " " , "
attrbt analysis
v use Bray-Curtis for entity analysis
use Manhattan Metric for attribute analysis
vi print out the entity trellis but not the attribute trellis
vii sort both trellises using group average sorting

viii produce both dendrograms
ix print both logged and unlogged two-way tables
x perform a princ compnt analysis of the attributes

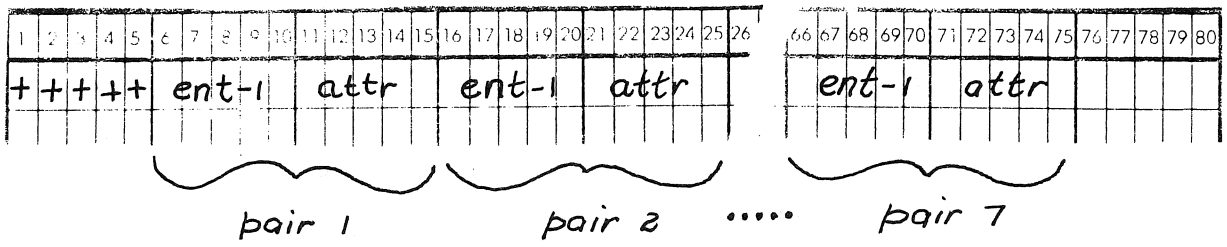


Example 2 above specifies

i Deck is in attribute format, no division, summated printout
ii no log transformation
iii entities untouched, attributes standardised
iv entities Bray-Curtis, attributes Manhattan Metric
v Attribute trellis only
vi entities nearest neighbour sorting, attributes flexible sorting
vii attribute dendrogram only
viii princ coord analysis of entities
Both analyses of attributes

3.5 Missing data options

Any entity-attribute pair which will not occur, and should be flagged as such, is specified on a missing data card. This card has the following form:



Up to seven entity-attribute pairs may be specified, and more than one missing data card is permissable. Users are referred to the general note regarding missing data in section 2.3 of this manual.

4.0 TAXAN-CLUSTR Interface

TAXAN is a program which provides a subset of the facilities of CLUSTR, but in addition handles disordered multistate data.

TAXAN has been modified so that it calls on sections of CLUSTR to provide some of the advanced output and analysis features available in other CLUSTR analyses. Thereby the user gains the advantage of being able to use disordered multistate data.

The input form of a TAXAN data deck is summarized below. The program as modified provides:

(a) normal TAXAN output
(b) plot of a dendrogram
(c) principal coordinate analysis of the produced dissimilarity matrix

TAXAN Card Deck

Card 1

A 0-65 character title starting in column 1

Card 2

The numbers NENT NB NN ND NO J1 J2 PLOT ORDIN in that order in any columns across the card. The numbers must be separated by spaces, and zeroes must be explicitly stated. The numbers represent:

NENT - number of entities in analysis
NB - number of binary attributes
NN - " " numeric "
ND - " " disordered m/s "
NO - " " ordered m/s "
J1 - clustering option
0,1 nearest neighbour
2 furthest neighbour
3 group average
4 simple average
5 centroid
6 incremental sum of squares
7 variance
J2 - standardisation option (numeric and ordered m/s only)
1 standardise by division by range
2 " " " " twice variance

PLOT - optional plot of dendrogram
0 plot required
1 no plot required
ORDIN - optional Principal Coordinate analysis
0 ordination required
1 ordination not required

Card 3

Size of multistates 40 per card, using sufficient cards to give ND+NO values. The values on each card should appear in columns 1,3,5,...,77,79.

Data Cards

The data cards for each entity follow in the following order:

[Binary data for entity 1
numeric " " " 1
disordered m/s " 1
ordered m/s " 1

[repeat for entity 2

.
.
.

[repeat for entity NENT

Binary data format

Binary data appears as 1,0,'*', or ' ' in successive columns of data cards , 60 values per card using as many cards as necessary to fill the required NB number.

Numeric data format

Numeric data appear 30 numbers per card, again using as many cards as are necessary to fill NN values. The numbers appear in any columns, separated by spaces.

Multistate data format

Multistate data appear as numerically coded values or '*' or ' '. They appear 40 values per card using as many values as are necessary to satisfy ND or NO values respectively. They are coded to appear in columns 1,3,5,...,77,79 of a data card.

LIMITATIONS

Presently TAXAN caters for a maximum of

NENT = 84
NM = 40
NB = 40
NO+ND= 40

Further enquiries regarding the properties of TAXAN should be directed to Dr. Clifford of the University of Queensland's Botany Department. Problems with its operational aspects should, as with CLUSTR, be directed to the Computer Centre.

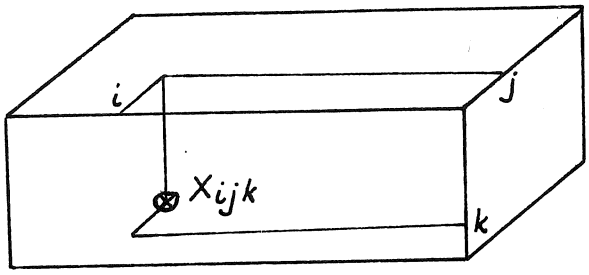
Part II

The Analyses of Three Dimensional Data

5.0 3D CLASSIFICATION - AN OVERVIEW

Three dimensional classification has found application particularly in the ecological literature (e.g. 10,11). The processes and operations which are performed in 3D analysis have analogues in 2D analysis, and so this section of the manual merely extends the concepts introduced in sections 1 and 2.

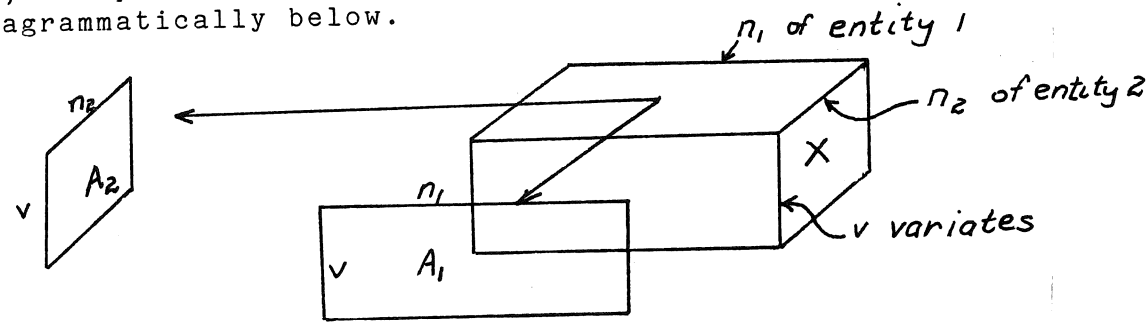
The X matrix now takes on the following form



The values of attribute k when two entities have the values i and j is given by Xijk.

It may therefore be seen that 3D analysis concerns itself with classifying two entities using the values of given attributes occurring for each possible co-incident pair of entities. It may of course not be possible for a given entity-1 entity-2 combination to occur, and so missing data options in 3D analysis are extended.

Having defined the 3D data matrix, the 3D classification reduces itself to two 2D classification problems. From the X matrix (3D), one produces two 2D auxiliary data matrices, A_1 and A_2, shown diagrammatically below.



* * * * * * * * * *

(10) Williams,W.T., and Stephenson,W., "The analysis of three-dimensional data (sites x species x times) in marine ecology", J. Exp. Mar. Ecol. 11,207-227.
(11) Stephenson,W., Williams,W.T., and Cook, S., "The macrobenthos of soft bottoms in southern Moreton Bay (south of Peel Island)" , Mem. Queensl. Mus. 17,73-124.

The A_1 (n_1 x v) matrix is produced by a mathematical reduction over all entity-2 values.

The A_2 (n_2 x v) matrix is produced by a mathematical reduction over all entity-1 values.

Once this reduction is accomplished, both normal and inverse analyses of both A_1 and A_2 matrices can be performed as outlined in part I of the manual.

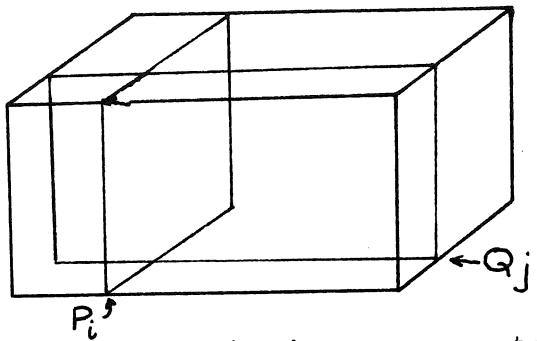
The various methods of mathematical reduction are defined formally in section 6.0 of this manual, and the method of setting out 3D data, options, dimensions, and missing data cards is given in section 7.0.

6.0 CLUSTR operations for 3D data.

The various options available to a user at each level of the classification process are now formally defined. It is assumed at this stage that the raw data matrix has been read from an input medium. The method by which this may be done will be given in sections 7.0 and 8.0.

6.1 Summary of raw data

An initial summary of the raw data is printed in 3D analysis. For an X matrix of the following form:



quantities Pi and Qi are derived, representing entity-1 and entity-2 summaries. The quantities may be:

(a) totals [S]

$$Pi = \sum_{jk}\sum Xijk$$

$$Qj = \sum_{ik}\sum Xijk$$

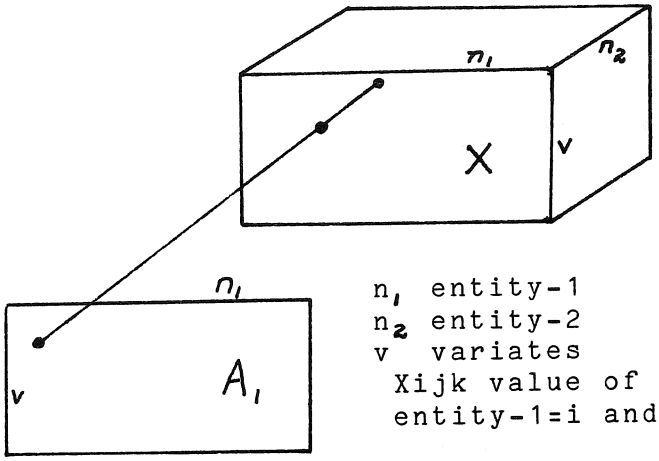
(b) averages [A]

$$Pi = \frac{\sum_{jk}\sum Xijk}{n_2 v}$$

$$Qj = \frac{\sum_{ik}\sum Xijk}{n_1 v}$$

6.2 Reduction options

For an X matrix of the form shown below,



n_1 entity-1
n_2 entity-2
v variates
Xijk value of variate k when entity-1=i and entity-2=k

the quantities in the A_1 matrix may be defined as:

(a) log-total [T]

$$A_1(ik) = \log_{10}\left(\sum_j Xijk + 1\right)$$

(b) log-average [L]

$$A_1(ik) = \log_{10}\left(\frac{\sum_j Xijk}{n_2} + 1\right)$$

(c) average [A]

$$A_1(ik) = \frac{\sum_j Xijk}{n_2}$$

(d) summed [S]

$$A_1(ik) = \sum_j Xijk$$

Similarly, the quantities in A_2 are defined by:

(a) log-total [T]

$$A_2\,jk = \log_{10}\left(\sum_i Xijk + 1\right)$$

(b) log-average [L]

$$A_2\,jk = \log_{10}\left(\frac{\sum_i Xijk}{n_1} + 1\right)$$

(c) average [A]

$$A_2\,jk = \frac{\sum_i Xijk}{n_1}$$

(d) summed [S]

$$A_2\,jk = \sum_i Xijk$$

Subsequent transformation of the A_1 and A_2 matrices, indeed all subsequent classification processes are carried out exactly as if A_1 and A_2 were independent 2D matrices for analysis as in part I of this manual. In particular A_1 and A_2 are operated on as in section 2.2 to produce independent F matrices

It will now be obvious why the statement in the introductory paragraph in 3.4.2 is made. The log transformation of data is part of the process (at least computationally) of reduction of 3D data to 2D data, and so merits a separate block in the options card. Not that the effect obtained is conceptually different than from other conditionings - but the allocation of a separate block allows log transformations to be performed in conjunction with for example standardisations. Also it is noted that log transformation of a 2D matrix (in the part I sense) is achieved by treating it as a degenerate 3D matrix.

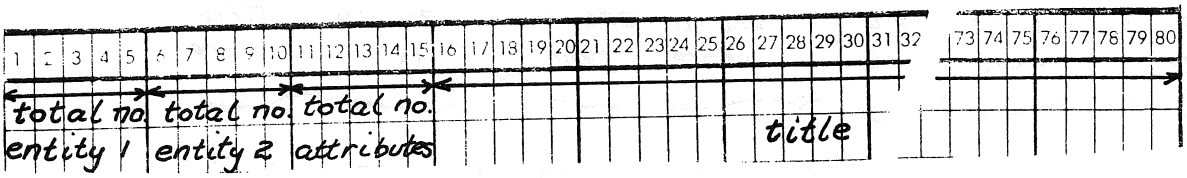
How to specify subsequent operations on the A_1 and A_2 matrices is a topic discussed in section 7.0 following.

7.0 Data input format for 3D data.

The data deck structure for 3D data is exactly the same as for 2D data shown in section 3.1. In general, the options specified for 2D data are immediately transferable to 3D data, however some extensions are allowable and will be discussed in the following paragraphs.

7.1 Dimensions card

The format for the dimensions card is as follows:

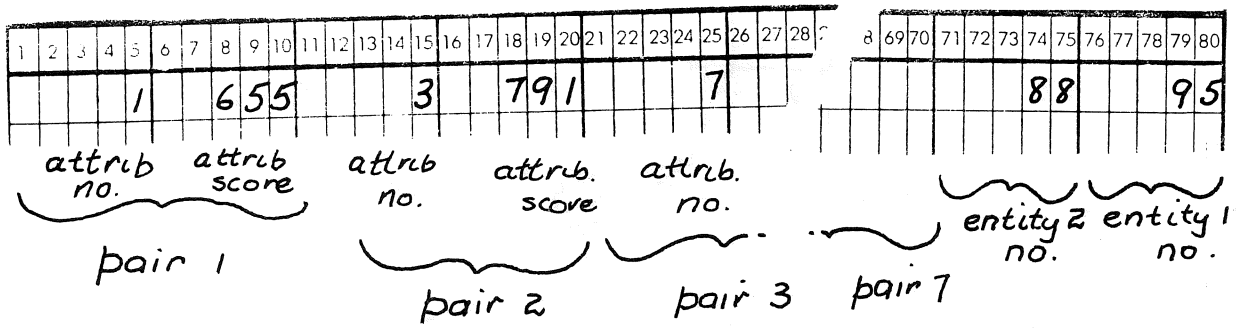


7.2 Samples cards

Once again there are two formats for data input.

Entity format

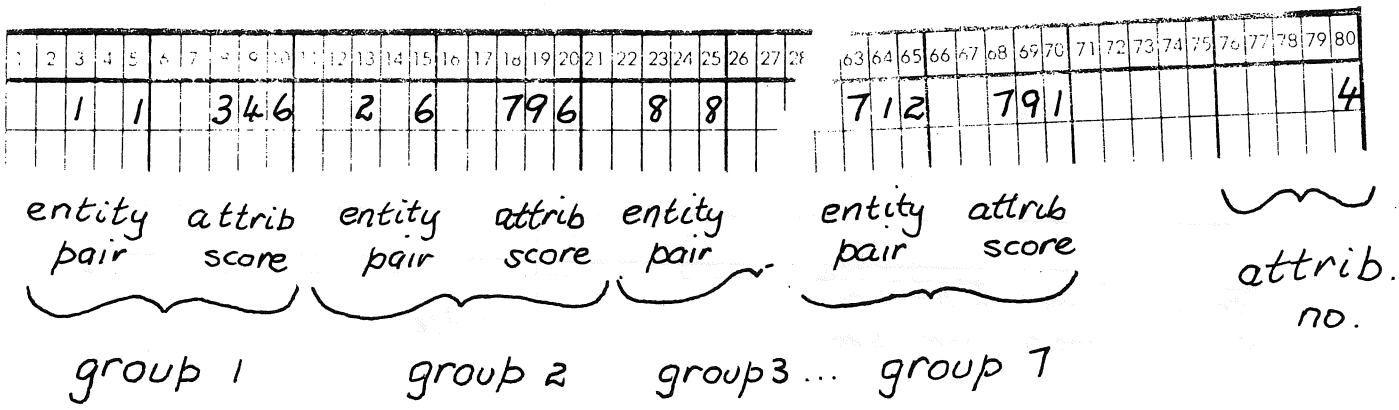
A user specifies attribute scores for each entity1-entity2 combination as folows:



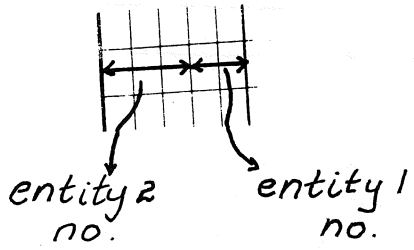
More than one card for a given entity pair is allowed.

Attribute format

A user specifies the scores for several entity pairs for that one attribute on one or more cards. The form is:



Each entity pair is specified in its own 5 column block as follows:



More then one card per attribute is permissable.

7.3 3D options card

The options card for 3D analysis is merely an extension of the 2D options card to allow for the extra 2D analysis required for A .

It is organized into exactly the same eleven blocks as shown in 3.4, with the exception that the log option block (cols 6-10) is now better termed the REDUCTION BLOCK (see also 6.2)

7.3.1 Data input block

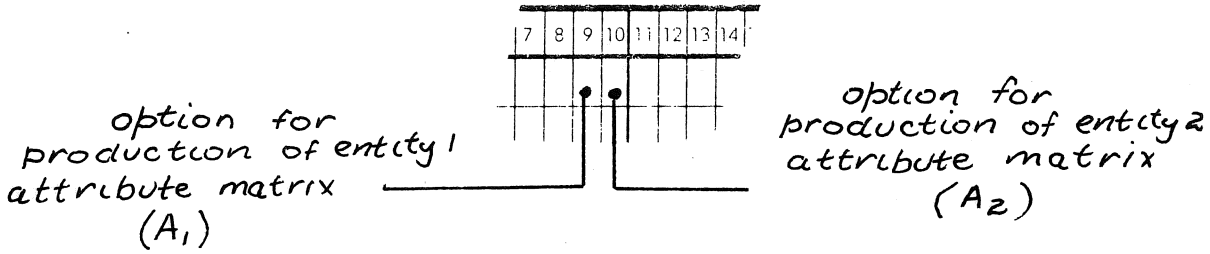
The form of the data input block is exactly the same for 3D analysis as it is for 2D analysis as shown in 3.4.1.

7.3.2 Reduction block

The reduction block specifies the way in which the 3D matrix is reduced to two 2D matrices. The options as defined previously are

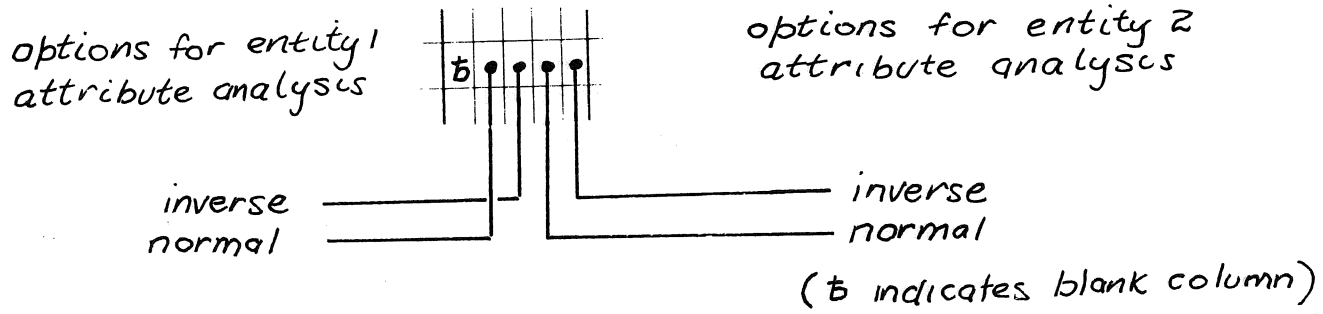
T - log total
L - log average
A - average
S - summation

The form of the block is:



7.3.3 Transformation block
7.3.4 Power block
7.3.5 Dissimilarity block
7.3.6 Printout block
7.3.7 Sorting block
7.3.8 Dendrogram block

Each of these blocks in 3D analysis contain exactly the same range of options as they did in 2D analysis (see 3.4.3 to 3.4.8 for available options). The blocks specify options for each of the four possible analyses, as follows:



7.3.9 Two way table block

The options available for two-way table printout are the same as outlined in 3.4.9, the ordering within the block being the same as in 7.3.2 .

7.3.10 Ordination block

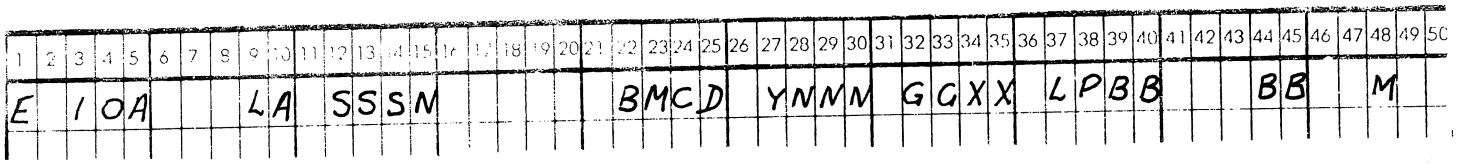
The options available for ordination are the same as those outlined in 3.4.10, the ordering being the same within the block as shown in 7.3.3-7.3.8 .

7.3.11 Note

If a particular analysis is not required for the particular users task the appropriate option column may be left blank where indicated in the previous text. CLUSTR recognizes the situation where 3D data is input but only for one of the 2D analyses by recognizing the appropriate blank column in the reduction block.

7.3.12 Example

The following example is an extension of the example in 3.4.12. As well as all the options specified there, the extra options specified for entity-2 attribute analysis are enumerated below.



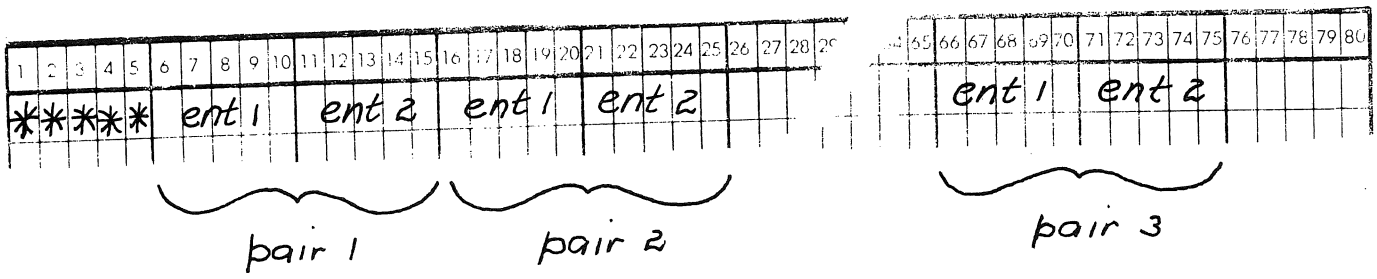
(i) 'average' reduction for ent2 attribute matrix
(ii) standardise for normal analysis
no transform for inverse analysis
(iii) Canberra metric for normal analysis
Euclidean distance squared for inverse analysis
(iv) no trellis printout
(v) flexible sorting for both analysis
(vi) all dendrograms required
(vii) both two way tables required
(viii) no ordination required

7.4 Missing data options

There are three types of missing data cards in 3D analysis. They are enumerated below.

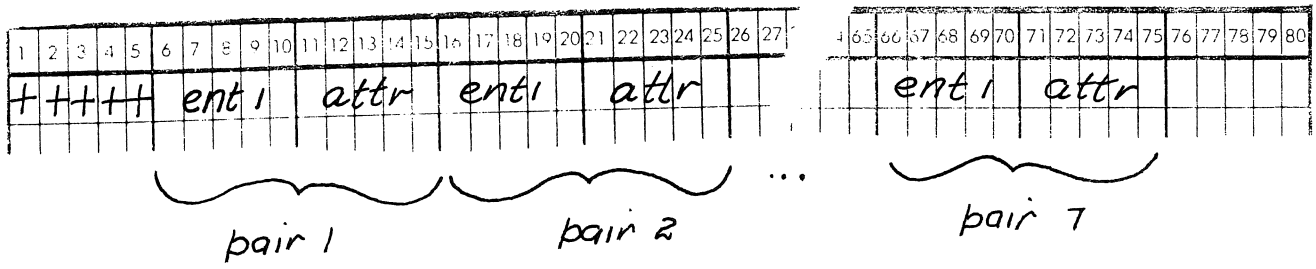
(a) Missing entity1-entity2 combination

If a given entity1-entity2 combination does not occur, i.e. value of all attributes for that combination cannot occur, the following missing data card is used:



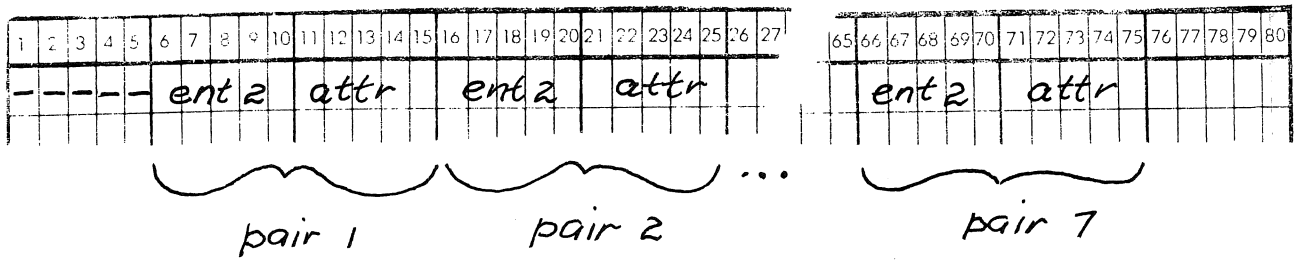
(b) Missing entity-1 attribute combination

If for a given entity1 a given attribute is impossible, regardless of the value of entity-2, the following form is used:



(c) Missing entity-2 attribute combination

If for a given entity-2, a given attribute cannot occur, regardless of the value of entity-1, the following form is used:



* * * * * * * * * * * * * *

Part III

CLUSTR on the PDP10

8.0 CLUSTR on the PDP-10

This section defines the system control cards required to control CLUSTR on the University of Queensland PDP-10 timesharing system where it currently operates under version 6.02 of the monitor.

8.1 The UQ Batch System

CLUSTR requires input from a card reader, and gives output to both line printer and plotter. Intermediate storage is required on moving head disc, on the users area.

The line printer, card reader and plotter on the PDP-10 are so-called spooling devices. That is, input from, or output to the real devices occurs well before or after the program's actual running. In the intervening time interval data to or from these devices is stored as files of information on moving head disc.

The task of providing control cards for CLUSTR therefore involves associating the appropriate spooled disc files with the running of the program at the appropriate time.

The following deck setup is typical and will be examined in some detail.

```
$SEQUENCE
$JOB   <user supplied parameters>
$DATA


   <  CLUSTR  data deck  >


$EOD
$TOPS10
.NOERROR
.R  STA:CLUSTR
.PRINT *.LPT
.PLOT PLT1:=*.PLT
.DEL *.DAT
$EOJ
```

The $SEQUENCE and $JOB cards identify the current users job to the system, the job being terminated by a $EOJ card.

The $DATA to $EOD cards include the CLUSTR data deck and instruct the batch system to set up a disc file. This file will later be used for input to CLUSTR as if it had come directly from the card reader. It is thus one of the spooled files. The disc file will be placed on the disc area set aside for the current user.

Having set up the appropriate input streams, the command .R STA:CLUSTR then causes the CLUSTR to begin execution.

During the course of its execution CLUSTR may produce disc files meant for the line printer and plotter, i.e. spooled output files. These files are placed in the appropriate queues for the real devices by the .PRINT and .PLOT commands.

The .DELete command tidies the user disc area by deleting the disc files which CLUSTR has created during the course of its running (xxxxxx.DAT files).

This sample deck setup is quite typical of most operations a user is likely to perform. It is modifiable, and will largely depend on whether or not the CLUSTR data deck already exists on disc as from a previous run or is to be read during this run using a $DATA - $EOD construction.

8.2 The UQ Remote Terminal

CLUSTR is primarily designed for use via the PDP-10 batch stream. To use it via a remote terminal a little knowledge of the FORTRAN operating system is reqyired, though once again only sufficient to organise the required input/output.

CLUSTR recognises when it is being run from a remote terminal, and expects input from FORTRAN unit number 5, and does output to unit number 6 (as opposed to 2 and 3 when running from the batch system). A user then uses interactive commands to associate units 5 and 6 with the appropriate (possibly spooled) peripheral devices. The following command sequence is typical:

```
.ASSIGN CDR:5
.ASSIGN LPT:6
    .
    .
.SET CDR ABC
    .
    .
.R STA:CLUSTR
```

The .ASSIGN commands associate units 5 and 6 with the card reader and line printer respectively. The .R command is as previously discussed in 8.1.

Then a user invokes exactly the same .PRINT, .PLOT and .DEL commands as shown in 8.1.

It has been assumed that a .CDR file has already been set up on user disc area, perhaps by a previous $DECK - $EOD combination from a previous batch run, or perhaps using EDITOR to create the file interactively.

A possible variation on the above interactive procedure is the following

```
.ASSIGN DSK:5
.ASSIGN DSK:6
.R STA:CLUSTR
```

In this case the non-spooled disc device has been assigned for input and output. By convention the disc file expected for input is the file FOR05.DAT, and output is sent to FOR06.DAT. No .SET CDR command is required as the card reader (spooled) has no longer been involved.

9.0 CLUSTR system components and interactions

This section is not aimed at the average user, but is rather designed to give an overview of CLUSTR structure to the programmer who may be faced with the maintenance or modification of CLUSTR. A resume of each program is given, and the way in which inter-program communication is achieved via disc files. Further the structure of each of the disc files is given. The explanations are all in terms of 3D analyses, because as far as CLUSTR is concerned ALL analyses are 3D, possibly with unit thickness in the 2D case.

It is assumed that the entity1 x entity2 x attribute analysis to be performed has dimensions $n_1 x n_2 x v$.

The diagram in fig. 9.1 summarizes CLUSTR action.

It may be seen that CLUSTR is really only the first of seven programs which are run in sequence automatically. The programs communicate via disc data files which are all binary files. It will also be evident that if execution aborts at any stage due to some external error(etc), executeion may be re-commenced at one of the intervening stages.

9.1 Program synopses

CLUSTR

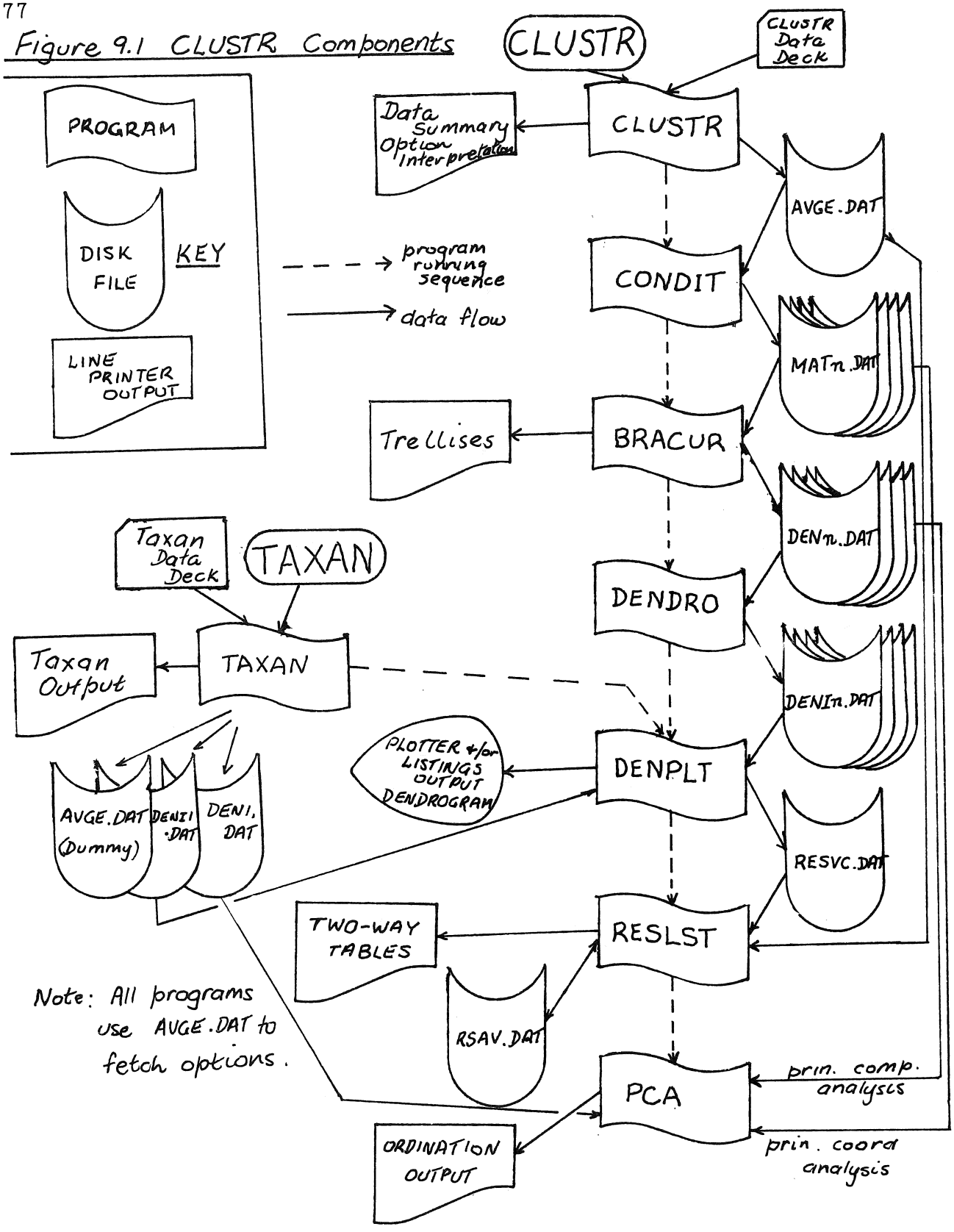
The first program in the suite reads dimensions and options cards, de-codes them and writes codified versions of them to disc for use by later programs. It then reads the samples cards storing two matrices for each of the A_1 and A_2 reduced matrices. The two matrices for each one record total entity 1 or 2 attribute values, and the number of occurences of each entity 1 or 2 to allow for possible missing data.

Throughout the listings copious references to sites, times and species occur. CLUSTR was originally written for ecological analyses, and later generalised. For sites, times and species, in the listings should be read entity1, entity2 and attributes.

Routines LISTC and LISRAW generate data summaries.

Routines LOGRD1,AV1,LOGAV1,LOGRD2,AV2, and LOGAV2 then perform the reductions using the four 'A_1 and A_2' matrices. These are written to disc for CONDIT.

Figure 9.1 CLUSTR Components



Note: All programs use AVGE.DAT to fetch options.

CONDIT

The second program is responsible for conditioning the reduced matrices ready for dissimilarity analysis by BRACUR.

Four matrices are generated in the conditioning process, corresponding to one for each of the inverse and normal analyses for both entity1/attribute and entity2/attribute data.

As BRACUR does dissimilarity calculations only by column, for inverse analysis CONDIT writes the matrix transpose to disc.

BRACUR

The third program writes four trellises ready for sorting by DENDRO.

DENDRO

This program is the later part of Prof. Burr's TAXAN2 program. The CLUSTR suite grew from an original desire to add facilities to this program. It was considered that better facilities could be made available if the sorting section only of TAXAN2 was added to CLUSTR and not vice versa. Consequently most of TAXAN2 has been discarded and this is the only section of code remaining not written by the author. It would, for obvious reasons gleaned from the listings, bear replacement, and be so replaced at a later stage.

The basic method used is that given by Clifford and Stephenson(1), and following the Lance and Williams approach(5).

DENPLT

The DENPLT program is the program which plots dendrograms from tha output of the DENDRO program.

It is basically a recursive binary tree drawer(!), made iterative due to the constraints of FORTRAN.

The ordering of leaves along the baseline of the dendrogram defines the resorting of the original data matrix, and is written to disc for use by the two-way table printer.

RESLST

The RESLST program prints a resorted version of the matrices contained in AVGE.DAT, after resorting according to the ordering reflected in RESVC.dat, written out by DENPLT.

RSAV.DAT is an auxiliary file used in the sorting.

PCA

PCA is the program which performs all the ordination using either trellis diagrams (DENn.DAT) or conditioned data matrices (MATn.DAT).

General notes

1. At each stage of the classification process only those options specified in AVGE.DAT from the options card are performed.

2. Owing to a quirk on the author's part, all arrays are stored by row and not by the conventional FORTRAN column. This means that the first subscript always refers to a column position along a row, and the second refers to a row position down a column for each element.

   Particular note of this fact should be taken in PCA where interface to standard SSP routines occurs.

3. Use is made throughout of FORTRAN's variable dimension facility, and the utility routine MORCOR. For further details consult the listing of MORCOR.

4. In each case where four matrices are written out, their names are of a form like for example MATn.DAT. The 'n' specifies the analysis for which the data is intended, viz:

1	entity1/attribute data	normal analysis
2	"	inverse "
3	entity2/attribute "	normal "
4	" "	inverse "

9.2 File Formats

The following is a record by record description of each binary file:

AVGE.DAT

a) 1 record of 3 words n_1 n_2 v
   and 13 words A5 title
b) 1 record of 36 words coded options
c) v records of n_1 words A_1 matrix

d) v records of n_2 words A_2 matrix

(TAXAN dummy AVGE.DAT does not include c and d)

MATn.DAT (e.g. MAT1.DAT)

a) 1 record of 2 words (number rows, number columns)
e.g. v n_1
b) conditioned matrix e.g. v records of n_1 words

DENn.DAT (e.g. DEN1.DAT)

a) 1 record of 1 word - size of (square) matrix
b) 1 record of 21 words - title and subtitle
c) lower triangle of the square matrix
e.g. $n_1 - 1$ records of 1,2,3,4...$n_1 - 1$ words each

DENIn.DAT (e.g. DENI1.DAT)

a) 1 record of 1 word - no. of entities on dendrogram baseline
b) 1 record of 21 words - title subtitle for dendrogram
c) fusions e.g. $n_1 - 1$ records with four words a,b,c,d.
these indicate entities a and b fuse to give c at level d

RESVC.DAT

a) n_1 records of 1 word each giving order along baseline of entity1 from dendrogram 1.
b) v records similarly from dendrogram 2
c) n_2 " " " " 3
d) v " " " " 4

RSAV.DAT - resorted conditioned matrix

a) v records of n_1 entities
b) v records of n_2 entities