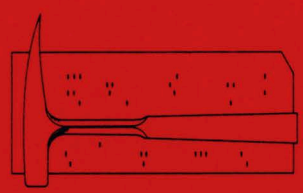
DANIEL F. MERRIAM, Editor

MINIMUM ENTROPY CRITERION FOR ANALYTIC ROTATION

By

RICHARD B. McCAMMON

University of Illinois at Chicago

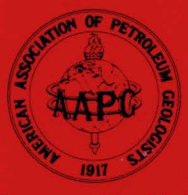


COMPUTER CONTRIBUTION 43

State Geological Survey

The University of Kansas, Lawrence

1970



in cooperation with the
American Association of Petroleum Geologists
Tulsa, Oklahoma

Editor's Remarks

COMPUTER CONTRIBUTION 43, "Minimum entropy criterion for analytic rotation", by R.B. McCammon is a program for structuring data for both principal component and factor analysis. The fact that these two related techniques are used widely now makes this contribution an important one. The program should be of interest to many geologists and other earth scientists involved with classification and correlation.

The program can be purchased for a limited time from the Geological Survey for $15.00 (US). The program is supplied on magnetic tape along with operating instructions. If punched cards are desired an extra $10.00 is necessary to cover cost of handling and postage.

A complete list of COMPUTER CONTRIBUTIONS can be obtained by writing, Editor, COMPUTER CONTRIBUTIONS, Kansas Geological Survey, The University of Kansas, Lawrence, Kansas 66044.

Computer Contribution

1. Mathematical simulation of marine sedimentation with IBM 7090/7094 computers, by J.W. Harbaugh, 1966 . . . . . (out of print)
2. A generalized two-dimensional regression procedure, by J.R. Dempsey, 1966 . . . $0.50
3. FORTRAN IV and MAP program for computation and plotting of trend surfaces for degrees 1 through 6, by Mont O'Leary, R.H. Lippert, and O.T. Spitz, 1966 . . . . $0.75
4. FORTRAN II program for multivariate discriminant analysis using an IBM 1620 computer, by J.C. Davis and R.J. Sampson, 1966 . . . . . $0.50
5. FORTRAN IV program using double Fourier series for surface fitting of irregularly spaced data, by W.R. James, 1966 . . . . . $0.75
6. FORTRAN IV program for estimation of cladistic relationships using the IBM 7040, by R.L. Bartcher, 1966 . . . . . $1.00
7. Computer applications in the earth sciences: Colloquium on classification procedures, edited by D.F. Merriam, 1966 . . . . . $1.00
8. Prediction of the performance of a solution gas drive reservoir by Muskat's equation, by Apolonio Baca, 1967 . . . . . $1.00
9. FORTRAN IV program for mathematical simulation of marine sedimentation with IBM 7040 or 7094 computers, by J.W. Harbaugh and W.J. Wahlstedt, 1967 . . . . . $1.00
10. Three-dimensional response surface program in FORTRAN II for the IBM 1620 computer, by R.J. Sampson and J.C. Davis, 1967 . . . . . $0.75

(continued on inside back cover)

*Active Member, ° Associate Member, + Junior Member, American Association of Petroleum Geologists

MINIMUM ENTROPY CRITERION FOR ANALYTIC ROTATION

by

Richard B. McCammon

ABSTRACT

Minimum entropy is described as an analytic criterion for rotation to simple structure for both principal component and factor analysis data matrices. Minimum entropy rotated matrices come closer to achieving the ideal simple structure than is possible using the varimax method in the sense that a greater proportion of absolute values of the coefficients in the rotated matrix lie closer to zero. This allows greater ease of recognition of the underlying structure in the original data array. The concept of rotation is extended to include rotation of principal components. Numerical examples are given to illustrate the application of minimum entropy rotation in both principal component analysis and factor analysis.

INTRODUCTION

Factor analysis is recognized as an effective statistical tool for extracting meaning from large arrays of multivariate data. The computer programs for factor analysis that are available have reduced the necessary calculations to routine operations (Imbrie, 1953; Cooley and Lohnes, 1962; Manson and Imbrie, 1964; Sampson, 1968; Klovan, 1968; IBM, 1968). This has resulted in an increased use of factor analysis in geology (for a list of applications, see Harbaugh and Merriam, 1968). Factor analysis, however, is not as many would think one single operation, but rather a sequence of statistical procedures in which each procedure entering into the calculations is considered on an individual basis in formulating the final result. It is with one of these procedures that the present paper is concerned - analytic rotation.

Analytic rotation has persisted as a challenge to those engaged in developing factor analytic methods. The reason stems from the desire for a simple factor structure which, as a rule, direct factor solutions do not provide. Although the problem has long been recognized (Thurstone, 1947), it has only been since computers became generally available that objective analytic criteria for rotation to simple structure have been devised for practical usage. Of the several analytic criteria proposed, the varimax method due to Kaiser (1958) is by far the best known and most widely used procedure for rotating an initial factor matrix to a position of simple structure. The computer programs that have been made available have incorporated this method.

The purpose of this paper is to make available a computer program for an analytic criterion of factor rotation proposed earlier (McCammon, 1966), which approaches more closely the intuitive concept of simple structure. The method is based on the entropy concept as it is defined in information theory and is used to describe the state of a given rotated factor matrix derived from an initial factor matrix. Although the results are similar to those obtained by the varimax method, the basic difference is that for the minimum entropy criterion, a greater proportion of factor loading values are closer to zero. For large matrices, this can amount to a significant difference.

The second purpose of the paper is to indicate how the concept of analytic rotation can be extended to the method of principal components, a near relative of factor analysis. Principal component analysis differs from factor analysis in that the extracted components explain the total variance of a given set of variables rather than the intercorrelations. Principal component analysis finds useful application in reducing the number of variables in a study and also in problems of classification. The concept of simple structure can be utilized to interpret the principal components of a system of variables in terms of the individual variables. A given subset of principal components can be rotated into a position of simple structure while preserving the total variance. This is analogous in factor analysis to the rotation of initial factors to a position of simple structure while preserving the total correlation. The minimum entropy concept is applicable to both types of rotation.

Acknowledgments. - The program was written during my employment with Gulf Research and Development Company. I wish to thank Chester Pelto for suggesting that the entropy function be used to define a criterion for rotation and for the advice offered during the course of the study. Tables 4 and 5 are reproduced with permission of the University of Chicago Press. The subroutine to invert a matrix in the computer program described is reproduced with the permission of IBM.

ROTATION OF PRINCIPAL COMPONENTS

The principal components of a set of observations which involves n variables are defined as the eigenvectors of the transformation which reduces the covariance matrix of the variables to diagonal form. The diagonal elements in the reduced matrix

are the eigenvalues corresponding to the eigenvectors and represent the variances of each of the principal components. It is usual to arrange the eigenvalues in order of decreasing value along the diagonal. Let C represent the covariance matrix of n given variables, and D the diagonal matrix with the eigenvalues arranged in order of decreasing value. Because C is a positive definite symmetric matrix, P in the transformation

$$D = P'CP \qquad (1)$$

is an orthogonal matrix (Murdoch, 1957, p. 145) and contains the set of eigenvectors as column vectors. The geometrical interpretation of principal components is that of a rotation of axes about an origin in a space defined by the coordinate axes of the n variables to a new position in which axes defined as linear combinations of the original variables represent variables which are uncorrelated. The origin is taken as the point defined by the means of the variables. Each newly defined variable is termed a principal component. The vector of principal components is defined for the i^{th} sample by a $n \times 1$ column vector Z_i expressed in terms of the original variables as

$$Z_i = P'X_i \qquad (2)$$

where X_i is the $n \times 1$ column vector denoting the original variables, and P is the matrix containing the eigenvectors as column vectors. Without loss of generality, we can assume that the original variables have been previously adjusted to have zero means. Thus, combining (1) and (2) and remembering that the $C = XX'$, the covariance matrix of Z is given by

$$C_z = D \qquad (3)$$

(see Scheffe, 1959, p. 8) where D is the diagonal matrix containing the eigenvalues of C. The variables defined as principal components, therefore, are uncorrelated and each has a variance corresponding to the eigenvalue given in the diagonal matrix.

The point to be made is that the principal component having the largest variance, called the first principal component, represents that linear combination of the variables with the greatest possible variance. The first principal component has a variance at least as large as the greatest variance of any single variable among the original variables. It represents the best single variable for characterizing the original data. The second principal component is defined as the linear combination of variables with the second largest variance but which, also, is uncorrelated with the first principal component. This can be viewed geometrically as choosing an axis which makes a right angle with the axis defined as the first principal component. Chosen in this manner, the two principal component axes represent axes along which the scatter of points representing the samples is at a minimum. By chosing the remaining principal components in the same manner, the result is a set of mutually orthogonal axes. The coordinates of each principal component with respect to the original variables are the eigenvectors. Because the principal component transformation is an orthogonal transformation, the total variance remains unchanged.

Usually, a large part of the total variance is accounted for by the first few principal components. This reflects the correlation that exist among the original variables. Thus the variability contained in the original samples can be represented by fewer variables in a lower dimensional space by plotting samples along the axes defined by the few significant principal components. In this manner it is possible to achieve a substantial compression of the data.

For problems involving classification, it is helpful to be able to assign a physical meaning to each principal component that is chosen to represent a group of samples. Ideally, this would mean that some physical measure could be found which was highly correlated with each principal component. More commonly principal components are interpreted in terms of the variables they represent. The dominant variables of any principal component are found by examining the coefficient defined for each variable. The coefficients, being elements of a set of orthonormal eigenvectors, form an orthonormal vector basis. Thus the coefficient denoted by p_{ij} for the i^{th} variable of the j^{th} coefficient is bounded in the interval

$$-1 \leqslant p_{ij} \leqslant 1 ;$$

moreover,

$$\sum_{i=1}^{n} p_{ij}^{\,2} = 1 \qquad j = 1, \ldots . n.$$

The principal components of a set of data are associated most closely with those variables whose coefficient values are close to plus or minus one. With respect to the geometry, the coordinate axes of the principal components are aligned or nearly so with these same variables.

Suppose that for a set of data which involves n variables the first $k(k < n)$ principal components account for the greater part of the total variance. This means that the data can be represented in a k-dimensional space without much loss of information. Suppose further we wish to interpret the principal components we have chosen to represent the n original variables. If we examine the absolute values of the coefficients, we will find usually that in addition to values close to one and zero, there will be values intermediate between these limits making the interpretation more difficult. We can eliminate this difficulty by rotating the principal components within the given k-dimensional subspace to a position where the new axes are aligned as nearly as possible with the axes representing the original variables. For $k = n$, this would be

simply a rotation to the position defined by the original variables. For $k < n$, however, the rotation to be performed is constrained within a lower dimensional space which may or may not contain one or more of the axes representing the original variables. In matrix form,

$$B = PT \tag{4}$$

where B represents the transformed $n \times k$ matrix whose columns are the new coordinate axes representing the first k principal components, and T is a $k \times k$ orthogonal matrix which represents the orthogonal transformation. Because P is composed of orthonormal column vectors and T is an orthogonal matrix, B likewise will be composed of orthonormal column vectors. The rotation preserves the total variance contained within the k-dimensional subspace (McCammon, 1966, p. 728).

Recognizing that components are more readily interpretable if the absolute values of the coefficients are close to either one or zero, the entropy function can be used to characterize the nonuniform distribution of coefficient values. The entropy expression is given by

$$H = - \sum_{i=1}^{n} \sum_{j=1}^{k} b_{ij}^2 \ln b_{ij}^2 . \tag{5}$$

To achieve a state of absolute minimum entropy requires that the squared coefficient values all have a value of either one or zero. Because of the constraints imposed by the dimension of the subspace and the condition of orthogonality for the column vectors of B, however, the minimum entropy will be some greater value. We wish to find a $k \times k$ orthogonal transformation matrix T such that the coefficients, b_{ij}, of the transformed components matrix B minimize the entropy expression H given by (5). The result is a set of k components which are more interpretable in terms of the original variables but yet which will account for the same total variance as the first k principal components. The difference will be that the components will no longer be uncorrelated.

Before considering how to find T, it is of interest to note that the varimax method due to Kaiser (1958) can be formulated to fit the present situation. By imposing the same conditions of orthogonal column vectors for the transformed matrix B in (4), Kaiser's "raw" varimax is given by

$$V = \sum_{j=1}^{k} \left[\sum_{i=1}^{n} (b_{ij}^2)^2 - \frac{\left(\sum_{i=1}^{n} b_{ij}^2 \right)^2}{n} \right] , \tag{6}$$

which is the variance one wishes to maximize. However, because

$$\sum_{i=1}^{n} b_{ij}^2 = 1 \text{ for all } j ,$$

the maximum found for (6) is identical to the maximum found for

$$Q = \sum_{i=1}^{n} \sum_{j=1}^{k} b_{ij}^4 , \tag{7}$$

which is the expression used to define the quartimax method (Harman, 1967, p. 298). Thus, for component rotation, the varimax and quartimax methods give identical solutions.

The methods for finding the maximum of expressions as (6) and (7) which involve orthogonal transformations make use of the relation that a rotation matrix T can be represented as a product of elementary rotation matrices

$$T = \prod_{j > i} T_{ij} ,$$

where T_{ij} has for its elements

δ_{rs}	$r \neq i, s \neq j$
$\cos \theta_{rs}$	$r = i, s = i$, or $r = j, s = j$
$\sin \theta_{rs}$	$r = i, s = j$
$- \sin \theta_{rs}$	$r = j, s = i$

and

$$- \pi/4 \leqslant \theta_{ij} \leqslant \pi/4 .$$

For each elementary plane rotation, it is necessary only to solve for that value of θ_{ij} which maximizes the expression. The rotations are performed in cyclic pairs (i, j) for i less than j. The iterative procedure is continued over successive cycles until no further improvement in the criterion to be maximized is obtained. For both the quartimax and the varimax method, it is possible to obtain a closed analytic expression for θ_{ij} at each iteration which will maximize the expression given in (6) or (7) (Harman, 1967, p. 300, 307). However, for the entropy function, this is not possible. Consider an elementary plane rotation defined by θ_{ij}, and (5) becomes

$$H = - \sum_{r=1}^{n} (p_{ri} \cos\theta_{ij} - p_{rj} \sin\theta_{ij})^2 \ln (p_{ri} \cos\theta_{ij} - p_{rj} \sin\theta_{ij})^2$$

$$-\sum_{r=1}^{n}(p_{ri}\sin\theta_{ij}+p_{rj}\cos\theta_{ij})^2 \ln(p_{ri}\sin\theta_{ij}+p_{rj}\cos\theta_{ij})^2$$

$$-\sum_{r=1}^{n}\sum_{s=1}^{k} b_{rs}^2 \ln b_{rs}^2 \quad . \qquad (8)$$

$$r,s\neq i,j$$

To minimize this expression, the partial derivative with respect to θ_{ij} would be set equal to zero, and then one would solve for the critical value, $\hat{\theta}_{ij}$, which can be done only by solving a nonlinear equation. Rather than solving for $\hat{\theta}_{ij}$ using nonlinear methods, it is more convenient to find $\hat{\theta}_{ij}$ by direct search because $\hat{\theta}_{ij}$ is contained within a known interval. Successive trial values of θ_{ij} may be substituted in (8) until a minimim value is attained. The complete minimization of (5) using elementary plane rotations is obtained in the same manner as for the varimax and quartimax method.

One may ask what advantage there is in the minimum entropy criterion if the varimax method can be adapted for principal component rotation. The answer is that the minimum entropy criterion comes closer to achieving simple structure in the sense that a greater proportion of coefficient values expressed in absolute value lie closer to zero following the transformation. This is illustrated in Figure 1. For large component matrices, this difference can be significant.

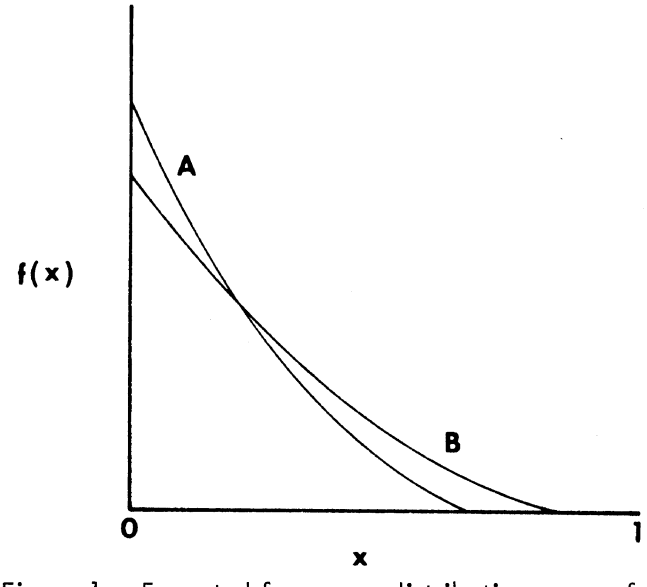


Figure 1.- Expected frequency distribution curves for coefficients of rotated data matrices. Coefficients are expressed as absolute values (A) minimum entropy rotation, and (B) varimax rotation.

FACTOR ANALYTIC ROTATION

In the previous section, we indicated how minimum entropy serves to define an analytic criterion for rotation of principal components. With some modification, we can apply the same concept to factor rotation.

In factor analysis, an initial factor matrix is obtained by any one of a number of methods (Harman, 1967, Pt. II, Chaps. 7-11). For the principal axis-solution, the initial factor matrix is made up of column vectors which are the eigenvectors multiplied by the square root of their associated eigenvalues derived from the correlation matrix of the original variables. If communalities rather than unities are placed in the diagonal elements of the correlation matrix, the resulting factor matrix is called the principal factor solution. In either situation, the initial factor matrix is the starting point for factor rotation.

In factor analysis, the interest lies in the observed correlations between variables. Each derived element in the initial factor matrix turns out to be the correlation coefficient between each variable and the corresponding factor. The goal of simple structure is to reduce to near zero as many of the correlations as possible while retaining a factor structure which accounts for the same total correlation observed for the original variables. One method for this is by an appropriate orthogonal transformation on the initial factor matrix.

The varimax method is designed to achieve simple structure. Given an initial nxk factor matrix A defined for n variables and the first k principal factors, a kxk rotation matrix T is found that defines a new nxk factor matrix B given by

$$B = AT \qquad (9)$$

in such a manner that the expression

$$V = \sum_{j=1}^{k}\sum_{i=1}^{n}\left(\frac{b_{ij}}{h_i}\right)^4 - \frac{\sum_{j=1}^{k}\left(\sum_{i=1}^{n}\left(\frac{b_{ij}}{h_i}\right)^2\right)^2}{n} \qquad (10)$$

is a maximum (Harman, 1966, p. 306). The communality, h_i^2, defined for each variable as

$$h_i^2 = \sum_{j=1}^{k} a_{ij}^2$$

remains invariant under the transformation.

To develop an analogous minimum entropy criterion, it is necessary to recognize the difference

that exists in the initial matrix to be rotated as compared with the initial matrix to be rotated in principal component analysis. For the initial factor matrix, the column vectors are no longer unit vectors, and, consequently, the length of the vectors will not remain invariant under an orthogonal transformation. Also, for the varimax method, the row vectors are normalized prior to rotation. The effect of this is to distribute the lengths of the column vectors of the rotated matrix as evenly as possible subject to the constraints imposed by the transformation. An expression which at its minimum value distributes the squared coefficient values within a column as unevenly as possible and the lengths of the column vectors as evenly as possible following an orthogonal transformation is given by

$$H^* = \frac{\sum_{i=1}^{n}\sum_{j=1}^{k} \frac{b_{ij}^2}{A_j} \ln \frac{b_{ij}^2}{A_j}}{\sum_{j=1}^{k} \frac{A_j}{A} \ln \frac{A_j}{A}} , \quad (11)$$

where $A_j = \sum_{i=1}^{n} b_{ij}^2$, and $A = \sum_{j=1}^{k} A_j$.

This expression defines the minimum entropy criterion for factor rotation. It has the property that, if $A_j = 1$ for all j, the solution in terms of the rotated matrix B is the same as the rotated matrix obtained in the situation of principal components. By substituting $A_j = 1$ for all j in (11),

$$H^* = \frac{H}{\ln k} \quad (12)$$

indicating that the derived solutions for both entropy functions give rise to the same minima.

The method for finding the rotation matrix T which minimizes the entropy expression in (11) is the same as before, that is, a direct search using elementary plane rotations. Convergence to the minimum value is attained in the same manner following an unspecified number of cyclic rotations.

The results obtained using the minimum entropy criterion may be compared with those obtained by the varimax method. Again, for the minimum entropy criterion, a higher proportion of absolute coefficient values in the transformed matrix lie closer to zero. It is for this reason that the minimum entropy criterion comes closer to satisfying the ideal simple structure.

PROGRAM OPERATION

Program Dimensions

The program is dimensioned to rotate twelve components or factors for fifty variables. However the limits may be raised by increasing the dimensions for the appropriate program variables. No change in the logic of the program is necessary. To facilitate the changing of program variables, the main program and all subroutines have been written so that all double subscripted arrays are expressed as single vector arrays. In order to change the dimension of any variable, it is only necessary to make the change in the main program.

Order of Input Cards

1. Program control card
2. Title card
3. Input - output data format card
   Data cards
4. Blank card

Program Usage

Card 1

Columns

1-5	NR = number of rows (variables) in initial matrix
6-10	NC = number of columns (components or factors) in initial matrix
11	MODE = 2 analytic factor rotation 1 analytic component rotation

Card 2

Columns 1-72 may be used for a title

Card 3

Columns 1-24 are used to specify the format to read in the initial data matrix. The data in the matrix are to be read one column at a time starting with the first column.

Columns 25-48 are used to specify the format to write out the data matrix. The data are to be written one row at a time starting with the first row.

Data Cards

(if a new set of data is to read, go to Card 1)

Card 4

A blank card follows the last data card.

Subroutines Required

All subroutines required are provided along with the main program. The user need only check to see that all subroutine decks are present before program execution. The subroutines included are:

MINENT	subroutine to perform the minimum entropy rotation.
MINSEK	subroutine to find the minimum of a nonlinear function.
COLENT	subroutine to evaluate the partial entropy for one column of a matrix
SHOOT	subroutine to calculate the entropy function for an elementary plane rotation.
MINV	subroutine to invert a matrix.

GEOLOGIC EXAMPLES

Environmental Classification

In a recent paper, multiple component analysis was shown to be an effective method for classifying large numbers of samples taken in environmental studies (McCammon, 1968). Of the other methods of classification that were compared, a method which produced similar results was based on minimum entropy rotated principal components derived from the correlation matrix of the original variables. Listed in Table 1 are the first three principal components in this study which accounted for the greater part of the total variance among the twelve original variables (Purdy, 1960). The minimum entropy rotated components are given also in Table 1. This rotation provided a set of components which could be more readily interpreted in terms of the original variables and produced a classification similar to the one obtained through multiple component analysis. In order that this example might serve as a test problem for the program on a different computer, Table 2 lists the input data for this problem and Table 3 lists the output from which Table 1 was constructed.

Biostratigraphic Correlation

In an earlier paper (McCammon, 1966), principal component analysis was used as a means for establishing a regional biostratigraphic marker horizon over an area in which the relative abundance for a large number of Foraminifera had been tabulated. Table 4 lists the first seven principal components of the covariance matrix obtained for 48 Foraminifera species. The minimum entropy rotated components are given in Table 5. After rotation, the number of coefficients whose absolute values exceeded 0.25 was reduced from 34 to 24, and the number of zero loadings (rounded to two decimals) was increased from 43 to 60. Thus the rotated components were easier to interpret in that fewer variables contributed to each component. Rotated component II was used subsequently to define a subsurface marker horizon which could be traced over the entire area of study (McCammon, 1966, fig. 5).

Brine Analysis

As a final example, we consider the minimum entropy rotation in factor analysis. In this instance, we can compare results with those obtained by the varimax method. Table 6 gives an initial factor matrix based on data for a large number of brine analyses (Sampson, 1968). The rotated matrix due to the varimax method is given in Table 7. The rotated matrix due to the minimum entropy criterion is given in Table 8. As can be seen, the results are almost identical. The minimum entropy criterion, however, does produce a matrix of coefficients whose average absolute value is 0.297 which is slightly less than the average value of 0.298 obtained for the varimax method. For larger size arrays, it is expected that the differences between these two methods will be significantly greater.

Table 1.-First three principal components obtained from correlation matrix of Bahamian sediment data taken from Purdy (1960) and rotated to position of simple structure using minimum entropy criterion.

Particle Constituents	Components					
	Initial			Rotated (Minimum Entropy)		
	I	II	III	I	II	III
Coralline algae	.095	.410	-.391	-.009	.001	-.574
Halimeda	.231	.262	-.245	.157	-.041	-.395
Corals	.095	.401	-.442	-.026	-.039	-.602
Oolite	-.321	-.298	-.342	-.429	-.351	.039
Grapestone	-.217	.293	.489	-.035	.587	.161
Cryptocryst. gr.	-.141	.417	.413	.020	.603	.008
Peneroplidae	.372	-.024	.126	.391	-.031	.035
Other foraminifera	.349	.177	.160	.391	.138	-.081
Mollusks	.333	.216	.041	.341	.090	-.187
Wgt. % < 1/8 mm	.428	-.215	.134	.435	-.174	.166
Fecal pellets	.300	-.336	.067	.286	-.269	.231
Mud aggregates	.337	-.125	-.010	.309	-.183	.021

Table 2.-Input for Bahamian sediment data.

```
000000000111111111122222222223333333333444444444455555555556666666666777777777 8
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

   12    31

    BAHAMIAN SEDIMENTS DATA - FIRST THREE PRINCIPAL COMPONENTS

(12F5.3)                (1H010X3F8.3)

  095  231  095 -321 -217 -141  372  349  333  428  300  337

  410  262  401 -298  293  417 -024  177  216 -215 -336 -125

 -391 -245 -442 -342  489  413  126  160  041  134  067 -010
```

Table 3.-Results of minimum entropy rotation for Bahamian sediments data.

INITIAL COMPONENT MATRIX

0.095	0.410	-0.391
0.231	0.262	-0.245
0.095	0.401	-0.442
-0.321	-0.298	-0.342
-0.217	0.293	0.489
-0.141	0.417	0.413
0.372	-0.024	0.126
0.349	0.177	0.160
0.333	0.216	0.041
0.428	-0.215	0.134
0.300	-0.336	0.067
0.337	-0.125	-0.010

CRITERION VALUES

1	0.29421E 01
2	0.26303E 01
3	0.25414E 01
4	0.25411E 01
5	0.25411E 01

TRANSFORMATION MATRIX

0.947	-0.262	-0.182
0.058	0.698	-0.713
0.315	0.666	0.677

ROTATED MATRIX

-0.009	0.001	-0.574
0.157	-0.041	-0.395
-0.026	-0.039	-0.602
-0.429	-0.351	0.039
-0.035	0.587	0.161
0.020	0.603	0.008
0.391	-0.031	0.035
0.391	0.138	-0.081
0.341	0.090	-0.187
0.435	-0.174	0.166
0.286	-0.269	0.231
0.309	-0.183	0.021

Table 4.-First seven principal components extracted from covariance matrix of Foraminifera data taken from McCammon (1966). Leading decimal of coefficients has been omitted for convenience. Coefficients whose absolute value is 0.25 or greater are underlined.

Foraminifera Species	I	II	III	IV	V	VI	VII
1	09	22	-54	-27	14	16	57
2	00	00	01	00	00	00	01
3	03	11	04	-05	-03	09	-05
4	32	01	00	-37	-05	08	-07
5	14	32	-38	29	-22	-67	-11
6	02	02	02	04	04	-01	01
7	32	-07	-01	10	02	17	-15
8	12	04	02	-04	-07	27	-19
9	10	42	-20	09	-13	23	-24
10	00	12	01	-03	18	-01	-08
11	00	36	05	-06	42	-03	-31
12	06	00	-22	-03	06	-07	13
13	05	-17	-13	-14	06	-06	-06
14	24	14	20	-05	03	-15	-01
15	30	-04	04	-29	-18	-17	-30
16	32	04	07	-14	-48	08	05
17	14	-24	-25	03	17	-09	-10
18	02	-03	04	02	03	-05	02
19	04	08	-02	-10	25	-02	-07
20	02	-05	01	-03	00	-02	00
21	04	27	05	-18	05	16	08
22	01	00	-02	-02	-02	-03	00
23	30	-34	-14	19	14	06	-01
24	16	-12	-20	-08	15	07	-22
25	07	-12	-05	06	08	-01	-05
26	24	-07	06	-27	06	-15	11
27	11	23	08	07	08	-05	11
28	02	04	01	00	00	06	-01
29	19	00	32	-02	02	-16	18
30	07	-05	-11	00	01	-02	04
31	-01	14	00	-04	30	-06	-20
32	24	-28	-10	23	17	00	09
33	05	-04	13	02	00	-10	12
34	07	-04	14	07	-01	-09	15
35	22	12	-04	48	-11	39	-02
36	04	-02	-01	09	04	-01	03
37	02	03	02	-01	-04	02	00
38	09	00	01	-02	00	00	-02
39	03	-02	08	07	-05	-11	09
40	00	01	00	01	05	00	-03
41	01	00	00	02	00	02	-01
42	-01	02	01	00	09	01	-05
43	00	04	-02	-01	02	01	-01
44	00	00	00	00	01	00	-01
45	-02	05	01	00	17	-01	-10
46	26	14	23	07	28	-05	18
47	19	18	21	25	16	00	22
48	00	00	00	01	01	-02	00

Table 5.-Rotated components using minimum entropy criterion obtained from initial principal components in Table 4. Coefficients whose absolute value is 0.25 or greater have been underlined.

Foraminifera Species	I	II	III	IV	V	VI	VII
1	00	00	00	00	-02	-01	89
2	01	00	01	00	00	00	00
3	01	09	-10	-08	04	05	00
4	00	-02	00	-48	03	11	12
5	00	00	-01	00	00	-92	00
6	05	-01	-01	02	03	-01	00
7	05	-26	-25	-18	02	06	-07
8	-09	-01	-26	-17	03	15	-06
9	-14	16	-50	-10	14	-16	08
10	04	03	-01	-01	22	-01	01
11	05	06	-05	-03	57	-01	-07
12	-02	-10	06	00	01	-11	23
13	-12	-15	15	-11	05	00	04
14	28	05	01	-21	07	-08	-07
15	-08	-02	06	-52	01	-12	-17
16	05	08	-17	-41	-40	-04	01
17	-13	-37	09	-03	11	-09	04
18	06	-03	04	00	00	-01	-03
19	04	-02	05	-05	27	02	06
20	00	-03	04	-03	-01	02	-01
21	10	22	-11	-11	11	12	19
22	00	-01	03	-02	02	-02	02
23	03	-53	-05	-02	-03	02	00
24	-18	-26	-04	-16	20	03	04
25	-01	-18	02	00	04	00	-03
26	16	-06	23	-29	00	02	10
27	26	08	-09	00	07	-08	08
28	00	02	-06	-02	01	03	01
29	39	02	14	-10	-08	02	-08
30	-02	-10	01	-02	-02	-04	08
31	02	01	01	00	39	-03	-04
32	12	-46	00	07	-04	00	02
33	17	-01	10	00	-08	00	-04
34	21	-02	07	02	-11	-01	-04
35	11	-18	-62	14	-13	02	-02
36	05	-07	-02	05	00	-03	00
37	01	03	-03	-03	-03	00	-01
38	00	00	01	-02	01	00	-01
39	12	-01	06	03	-10	-07	-04
40	00	-01	00	01	06	00	-01
41	00	-01	-03	01	00	01	00
42	00	-01	00	01	10	02	-02
43	-01	01	-02	00	04	00	02
44	00	00	00	00	01	00	00
45	01	-01	00	03	20	01	-03
46	48	-06	-02	-03	16	04	06
47	46	-03	-14	12	03	-01	03
48	01	-01	01	01	00	-01	00

Table 6.-Initial factor matrix of Arbuckle brine data taken from Sampson (1968).

.8667	-.3192	.0506	-.0655
.7482	-.3418	-.1126	-.0811
.2827	.4587	.0834	.5449
-.2062	-.0498	-.6663	.4700
.2528	.6907	-.3526	-.0244
.9051	-.0967	-.0543	-.0043
.7919	.2839	.0184	.0244
.8451	-.0578	-.0491	-.0338
-.0369	-.3663	.3888	.7147
-.7214	-.3258	-.3670	-.0862
-.3923	.3040	.5243	-.1116

Table 7.-Varimax rotated factor matrix of Arbuckle brine data taken from Sampson (1968).

.9175	.0233	.0994	.0872
.8303	-.0611	-.0492	.0191
.0498	.7553	-.0721	.1264
-.1639	.0852	-.8214	.0315
.0388	.5408	-.1411	-.5933
.8823	.2235	.0249	-.0501
.6405	.4919	.1326	-.1973
.8158	.2159	.0445	-.0839
-.0167	.1704	-.0705	.8736
-.5241	-.5750	-.4041	.0027
-.5025	.0903	.5205	.0459

Table 8.-Minimum entropy rotated factor matrix of Arbuckle brine data.

.9143	.0931	.0901	.0850
.8319	.0091	-.0597	.0133
-.0144	.7422	-.0629	.1981
-.1810	.0719	-.8175	.0606
-.0151	.5957	-.1515	-.5365
.8598	.3019	.0145	-.0310
.5959	.5605	.1239	-.1549
.7941	.2917	.0340	-.0657
-.0235	.0868	-.0471	.8872
-.4784	-.6135	-.4023	-.0398
-.5010	.0394	.5286	.0416

REFERENCES

Cooley, W.W., and Lohnes, P.R., 1962, Multivariate procedures for the behavioral sciences: John Wiley & Sons, Inc., New York, 211 p.

Harbaugh, J.W., and Merriam, D.F., 1968, Computer applications in stratigraphic analysis: John Wiley & Sons, Inc., New York, 282 p.

Harman, H.H., 1967, Modern factor analysis (2nd ed.): Univ. Chicago Press, Chicago, 474 p.

IBM, 1968, System 360 scientific subroutine package (360A-CM-03X) version III programmer's manual: IBM Appl. Prog. H 20-0205-3, Tech. Publ. Dept., White Plains, New York, 454 p.

Imbrie, J., 1963, Factor and vector analysis programs for analyzing geologic data: Office of Naval Research, Geography Branch, ONR Task No. 389-135, Tech. Rept. No. 6, 83 p.

Kaiser, H.F., 1958, The varimax criterion for analytic rotation in factor analysis: Psychometrika, v. 23, p. 187-200.

Klovan, J.E., 1968, Q-mode factor analysis program in FORTRAN IV for small computers: Kansas Geol. Survey Computer Contr. 20, p. 38-50.

Manson, V., and Imbrie, J., 1964, FORTRAN program for factor and vector analysis of geologic data using an IBM 7090 or 7094/1401 computer system: Kansas Geol. Survey Sp. Dist. Publ. 13, 46 p.

McCammon, R.B., 1966, Principal component analysis and its application in large-scale correlation studies: Jour. Geology, v. 74, no. 5, pt. 2, p. 721-733.

McCammon, R.B., 1968, Multiple component analysis and its application in classification of environments: Am. Assoc. Petroleum Geologists, v. 52, no. 11, pt. 1, p. 2178-2196.

Murdoch, D.C., 1957, Linear algebra for undergraduates: John Wiley & Sons, Inc., New York, 239 p.

Purdy, E.G., 1960, Recent calcium carbonate facies of the Great Bahama Bank: Unpubl. doctoral dissertation, Columbia Univ., 174 p.

Sampson, R.J., 1968, R-mode factor analysis program in FORTRAN II for the IBM 1620 computer: Kansas Geol. Survey Computer Contr. 20, p. 13-37.

Scheffe, H., 1959, The analysis of variance: John Wiley & Sons, Inc., New York, 477 p.

Thurstone, L.L., 1947, Multiple factor analysis: Univ. Chicago Press, Chicago, 535 p.

Listing of minimum entropy rotation program.

```
C                                                                       1
C                  MINIMUM ENTROPY ROTATION PROGRAM                     2
C                                                                       3
C                                                                       4
C          A COMPUTER PROGRAM TO PERFORM AN ANALYTIC ROTATION BASED      5
C     ON THE MINIMUM ENTROPY CRITERION ON AN INITIAL DATA MATRIX. THE    6
C     ROTATION MAY BE PERFORMED ON EITHER A PRINCIPAL COMPONENT MATRIX   7
C     OR A FACTOR MATRIX. THE PRESENT PROGRAM IS DIMENSIONED TO ROTATE   8
C     UP TO A 50 BY 12 MATRIX. THESE LIMITS CAN BE EXTENDED BY          9
C     INCREASING THE DIMENSIONS OF THE APPROPRIATE PROGRAM VARIABLES.   10
C                                                                       11
C                                                                       12
C     ORDER OF INPUT CARDS                                              13
C       1. PROGRAM CONTROL CARD                                         14
C       2. TITLE CARD                                                   15
C       3. INPUT-OUTPUT FORMAT CARD                                     16
C         ** DATA CARDS **                                              17
C       4. BLANK CARD                                                   18
C                                                                       19
C                                                                       20
C     PROGRAM CONTROL PARAMETERS                                        21
C                                                                       22
C         NR - NUMBER OF ROWS IN INITIAL MATRIX                         23
C         NC - NUMBER OF COLUMNS IN INITIAL MATRIX                      24
C       MODE - MODE = 1, COMPONENT ROTATION                             25
C              MODE = 2, FACTOR ROTATION                                26
C                                                                       27
C                                                                       28
C     PROGRAM VARIABLES                                                 29
C                                                                       30
C         X - INPUT DATA VECTOR OF LENGTH NR*NC                         31
C         C - WORKING VECTOR OF LENGTH NR*NC                            32
C         D,D1 - WORKING VECTORS OF LENGTH NC**2                        33
C         S - WORKING VECTOR OF LENGTH NR                               34
C         EV - CRITERION VECTOR OF LENGTH 10*NC                         35
C         AC,AE,LC,MC - WORKING VECTORS OF LENGTH NC                    36
C                                                                       37
C                                                                       38
C     SUBROUTINES REQUIRED                                              39
C                                                                       40
C       MINENT   MINIMUM ENTROPY ROTATION                               41
C                                                                       42
C       MINSEK   MINIMUM OF NONLINEAR FUNCTION                          43
C                                                                       44
C       COLENT   COLUMN ENTROPY CRITERION                               45
C                                                                       46
C       SHOUT    ENTROPY CRITERION FOR ELEMENTARY PLANE ROTATION        47
C                                                                       48
C       MINV     MATRIX INVERSION                                       49
C                                                                       50
C                                                                       51
C                                                                       52
C     ***********                                                       53
C     MAIN PROGRAM                                                      54
```

```
C      ***********                                                          55
C                                                                           56
       DIMENSION FMT(18),FMI(6),FMO(6),X(600),C(600),S(50),D(144),D1(144)   57
      1,EV(120),LC(12),MC(12),AE(12),AC(12)                                 58
     1 READ(5,100) NR,NC,MODE                                               59
   100 FORMAT(2I5,I1)                                                       60
       IF(NR.EQ.0) STOP                                                     61
       READ(5,101) (FMT(I),I=1,18)                                          62
   101 FORMAT(18A4)                                                         63
       READ(5,102) (FMI(I),I=1,6),(FMO(I),I=1,6)                            64
   102 FORMAT(12A4)                                                         65
       I2=0                                                                 66
       DO 10 J=1,NC                                                         67
       I1=I2+1                                                              68
       I2=I1+NR-1                                                           69
    10 READ(5,FMI) (X(I),I=I1,I2)                                           70
       WRITE(6,200) (FMT(I),I=1,18)                                         71
   200 FORMAT(1H1/1H T30,'MINIMUM ENTROPY ROTATION'    //1H 5X18A4  /    )  72
       GO TO (8,9),MODE                                                     73
     8 WRITE(6,203)                                                         74
   203 FORMAT(1H '      INITIAL COMPONENT MATRIX'      )                    75
       GO TO 11                                                             76
     9 WRITE(6,205)                                                         77
   205 FORMAT(1H '      INITIAL FACTOR MATRIX'       )                      78
    11 I2=(NC-1)*NR                                                         79
       DO 12 I=1,NR                                                         80
       I2=I2+1                                                              81
    12 WRITE(6,FMO) (X(J),J=I,I2,NR)                                        82
       IF(MODE.EQ.1) GO TO 15                                               83
       DO 17 I=1,NR                                                         84
       K=I-NR                                                               85
       S(I)=0.                                                              86
       DO 17 J=1,NC                                                         87
       K=K+NR                                                               88
    17 S(I)=S(I)+X(K)**2                                                    89
       DO 18 I=1,NR                                                         90
       S(I)=SQRT(S(I))                                                      91
       K=I-NR                                                               92
       DO 18 J=1,NC                                                         93
       K=K+NR                                                               94
    18 X(K)=X(K)/S(I)                                                       95
    15 K=C                                                                  96
       DO 40 I=1,NR                                                         97
       DO 40 J=1,NC                                                         98
       K=K+1                                                                99
    40 C(K)=X(K)                                                           100
       CALL MINENT(NR,NC,X,NITN,EV,AE,AC)                                  101
       WRITE(6,207) (I,EV(I),I=1,NITN)                                     102
   207 FORMAT(1HO'      CRITERION VALUES' //(1H 10XI3,E20.5))               103
       IF(NITN.LT.(10*NC)) GO TO 13                                        104
       WRITE(6,208) NITN                                                   105
   208 FORMAT(1HO'      ERROR MESSAGE: NUMBER OF CYCLES EQUALS SET LIMIT O  106
      1F 10*NC = ',I3  // )                                                107
    13 CONTINUE                                                            108
       DO 41 I=1,NC                                                        109
       K1=I-NC                                                             110
       DO 41 J=1,NC                                                        111
       K1=K1+NC                                                            112
       D(K1)=C.                                                            113
       K2=(I-1)*NR                                                         114
```

```
      K3=(J-1)*NR                                                       115
      DO 41 K=1,NR                                                      116
      K2=K2+1                                                           117
      K3=K3+1                                                           118
   41 D(K1)=D(K1)+C(K2)*X(K3)                                           119
      IF(MODE.EQ.1) GO TO 60                                            120
      DO 62 I=1,NC                                                      121
      K1=I-NC                                                           122
      DO 62 J=1,NC                                                      123
      K1=K1+NC                                                          124
      D1(K1)=0.                                                         125
      K2=(I-1)*NR                                                       126
      K3=(J-1)*NR                                                       127
      DO 62 K=1,NR                                                      128
      K2=K2+1                                                           129
      K3=K3+1                                                           130
   62 D1(K1)=D1(K1)+C(K2)*C(K3)                                         131
      CALL MINV(D1,NC,DET,LC,MC)                                        132
      DO 64 I=1,NC                                                      133
      K1=I-NC                                                           134
      DO 64 J=1,NC                                                      135
      K1=K1+NC                                                          136
      C(K1)=0.                                                          137
      K2=I-NC                                                           138
      K3=(J-1)*NC                                                       139
      DO 64 K=1,NC                                                      140
      K2=K2+NC                                                          141
      K3=K3+1                                                           142
   64 C(K1)=C(K1)+D1(K2)*D(K3)                                          143
      K=0                                                               144
      DO 65 I=1,NC                                                      145
      DO 65 J=1,NC                                                      146
      K=K+1                                                             147
   65 D(K)=C(K)                                                         148
      DO 28 I=1,NR                                                      149
      K=I-NR                                                            150
      DO 28 J=1,NC                                                      151
      K=K+NR                                                            152
   28 X(K)=X(K)*S(I)                                                    153
   60 CONTINUE                                                          154
      WRITE(6,209)                                                      155
  209 FORMAT(1H0'      TRANSFORMATION MATRIX'          )                156
      I2=(NC-1)*NC                                                      157
      DO 16 I=1,NC                                                      158
      I2=I2+1                                                           159
   16 WRITE(6,FMC) (D(J),J=I,I2,NC)                                     160
      WRITE(6,211)                                                      161
  211 FORMAT(1H0'      ROTATED MATRIX'          )                       162
      I2=(NC-1)*NR                                                      163
      DO 14 I=1,NR                                                      164
      I2=I2+1                                                           165
   14 WRITE(6,FMC) (X(J),J=I,I2,NR)                                     166
      GO TO 1                                                           167
       END                                                              168
```

```
C                                                                       169
C                                                                       170
C     ************                                                      171
      SUBROUTINE MINENT(NR,NC,X,NITN,EV,AE,AC)                          172
C     ************                                                      173
C                                                                       174
C     SUBROUTINE TO PERFORM MINIMUM ENTROPY ROTATION                    175
      DIMENSION X(1),AE(1),AC(1),EV(1),T(4),H(10)                       176
      A1=-.785398                                                       177
      A2=.785398                                                        178
      H(1)=A2/10.                                                       179
      DO 14 J=2,5                                                       180
   14 H(J)=H(J-1)/10.                                                   181
      SA=0.                                                             182
      K=0                                                               183
      DO 32 J=1,NC                                                      184
      XS=0.                                                             185
      AE(J)=0.                                                          186
      DO 52 I=1,NR                                                      187
      K=K+1                                                             188
      XC=X(K)**2                                                        189
      XS=XS+XC                                                          190
      IF(XC)52,52,34                                                    191
   34 AE(J)=AE(J)-XC*ALOG(XC)                                           192
   52 CONTINUE                                                          193
      SA=SA+XS                                                          194
   32 AC(J)=-XS*ALOG(XS)                                                195
      CN=0.                                                             196
      CD=0.                                                             197
      ASA=ALOG(SA)                                                      198
      ASA=SA*ASA                                                        199
      DO 33 J=1,NC                                                      200
      CN=CN+AE(J)                                                       201
   33 CD=CD+AC(J)                                                       202
      NITN=1                                                            203
      ENT=(CN+ASA)/(CD+ASA)                                             204
      EV(NITN)=ENT                                                      205
      NTOL=10*NC                                                        206
      NC1=NC-1                                                          207
   22 IF(NITN.GE.NTOL) GO TO 50                                         208
   23 NITN=NITN+1                                                       209
      ENT=0.                                                            210
      DO 25 II=1,NC1                                                    211
      II=II                                                             212
      CN=CN-AE(II)                                                      213
      CD=CD-AC(II)                                                      214
      IJ=II+1                                                           215
      DO 20 JJ=IJ,NC                                                    216
      JJ=JJ                                                             217
      K1=(II-1)*NR                                                      218
      K2=(JJ-1)*NR                                                      219
      CN=CN-AE(JJ)                                                      220
      CD=CD-AC(JJ)                                                      221
      CALL MINSEK(B,H,A1,A2,NR,NC,II,JJ,CN,CD,ASA,X)                    222
      T(1)=COS(B)                                                       223
      T(4)=T(1)                                                         224
      T(3)=SIN(B)                                                       225
      T(2)=-T(3)                                                        226
      XN=0.                                                             227
      XS=0.                                                             228
```

```
      DO 5 I=1,NR                                                       229
      K1=K1+1                                                           230
      K2=K2+1                                                           231
      Y=X(K1)*T(1)+X(K2)*T(2)                                           232
      X(K2)=X(K1)*T(3)+X(K2)*T(4)                                       233
      XC=X(K2)**2                                                       234
      IF(XC)53,53,54                                                    235
   54 XN=XN-XC*ALOG(XC)                                                 236
   53 XS=XS+XC                                                          237
    5 X(K1)=Y                                                           238
      AE(JJ)=XN                                                         239
      AC(JJ)=-XS*ALOG(XS)                                               240
      CN=CN+AE(JJ)                                                      241
      CD=CD+AC(JJ)                                                      242
   20 CONTINUE                                                          243
      XN=0.                                                             244
      XS=0.                                                             245
      K1=(II-1)*NR                                                      246
      DO 55 I=1,NR                                                      247
      K1=K1+1                                                           248
      XC=X(K1)**2                                                       249
      XS=XS+XC                                                          250
      IF(XC)55,55,35                                                    251
   35 XN=XN-XC*ALOG(XC)                                                 252
   55 CONTINUE                                                          253
      AE(II)=XN                                                         254
      AC(II)=-XS*ALOG(XS)                                               255
      CN=CN+AE(II)                                                      256
      CD=CD+AC(II)                                                      257
      CALL COLENT(TENT,II,X,NR)                                         258
      ENT=ENT+TENT                                                      259
   25 CONTINUE                                                          260
      CALL COLENT(TENT,NC,X,NR)                                         261
      ENT=ENT+TENT                                                      262
   38 ENT=(CN+ASA)/(CD+ASA)                                             263
      EV(NITN)=ENT                                                      264
      TT=(FENT-ENT)/FENT                                                265
      IF(TT-.000001)50,50,29                                            266
   29 FENT=ENT                                                          267
      GO TO 22                                                          268
   50 CONTINUE                                                          269
      RETURN                                                            270
      END                                                               271
```

```
C                                                                       272
C                                                                       273
C     ************                                                      274
      SUBROUTINE MINSEK (B,H,A1,A2,NR,NC,II,JJ,CN,CD,ASA,X)             275
C     ************                                                      276
C                                                                       277
C     SUBROUTINE TO FIND THE MINIMUM OF A NONLINEAR FUNCTION            278
      DIMENSION X(1),H(1)                                               279
      TMB=0.                                                            280
      ICY=0                                                             281
    2 I=0                                                               282
      MIN=2                                                             283
      CALL SHOOT(TMF,TMB,II,JJ,NR,NC,X,ASA,CN,CD)                       284
   20 B2=TMB                                                            285
      I=I+1                                                             286
      IF(I-5)21,21,40                                                   287
   21 B1=B2-H(I)                                                        288
      IF(B1-A1)24,22,22                                                 289
   22 CALL SHOOT(F,B1,II,JJ,NR,NC,X,ASA,CN,CN)                          290
      IF(F-TMF)23,24,24                                                 291
   23 TMF=F                                                             292
      TMB=B1                                                            293
      MIN=1                                                             294
   24 B3=B2+H(I)                                                        295
      IF(B3-A2)25,25,16                                                 296
   25 CALL SHOOT(F,B3,II,JJ,NR,NC,X,ASA,CN,CN)                          297
      IF(F-TMF)26,16,16                                                 298
   26 TMF=F                                                             299
      TMB=B3                                                            300
      MIN=3                                                             301
      GO TO 16                                                          302
   10 MIN=2                                                             303
      B1=TMB-H(I)                                                       304
      IF(B1-A1)16,11,11                                                 305
   11 CALL SHOOT(F,B1,II,JJ,NR,NC,X,ASA,CN,CN)                          306
      IF(F-TMF)12,16,16                                                 307
   12 TMF=F                                                             308
      TMB=B1                                                            309
      MIN=1                                                             310
      GO TO 16                                                          311
   30 MIN=2                                                             312
      B3=TMB+H(I)                                                       313
      IF(B3-A2)31,31,16                                                 314
   31 CALL SHOOT(F,B3,II,JJ,NR,NC,X,ASA,CN,CN)                          315
      IF(F-TMF)32,16,16                                                 316
   32 TMF=F                                                             317
      TMB=B3                                                            318
      MIN=3                                                             319
   16 GO TO(10,20,30),MIN                                               320
   40 IF(ICY)41,41,43                                                   321
   41 IF(ABS(TMB)-A2+.00001)44,44,42                                    322
   42 ICY=1                                                             323
      B=TMB                                                             324
      TF=TMF                                                            325
      IF(B)46,47,47                                                     326
   46 TMB=A2                                                            327
      GO TO 2                                                           328
   47 TMB=A1                                                            329
      GO TO 2                                                           330
   43 IF(TMF-TF)44,45,45                                                331
   44 B=TMB                                                             332
   45 RETURN                                                            333
      END                                                               334
```

```
C                                                                       335
C                                                                       336
C     ************                                                      337
      SUBROUTINE COLENT(E,J,X,NR)                                       338
C     ************                                                      339
C                                                                       340
C     SUBROUTINE TO EVALUATE THE ENTROPY CRITERION FOR ONE COLUMN OF A  341
C     MATRIX                                                            342
      DIMENSION X(1)                                                    343
      T=0.                                                              344
      E=0.                                                              345
      K=(J-1)*NR                                                        346
      DO 4 I=1,NR                                                       347
      K=K+1                                                             348
      XC=X(K)**2                                                        349
      IF(XC)4,4,3                                                       350
    3 E=E-XC*ALOG(XC)                                                   351
    4 T=T+XC                                                            352
      E=E/T+ALOG(T)                                                     353
       END                                                              354
```

```
C                                                                       355
C                                                                       356
C     ************                                                      357
      SUBROUTINE SHCUT(F,B,II,JJ,NR,NC,X,ASA,CN,CD)                     358
C     ************                                                      359
C                                                                       360
C     SUBROUTINE TO EVALUATE THE ENTROPY CRITERION FOR AN ELEMENTARY    361
C     PLANE ROTATION                                                    362
      DIMENSION X(1),T(4),IJ(2)                                         363
      IJ(1)=II                                                          364
      IJ(2)=JJ                                                          365
      T(1)=COS(B)                                                       366
      T(4)=T(1)                                                         367
      T(3)=SIN(B)                                                       368
      T(2)=-T(3)                                                        369
      F=0.                                                              370
      CN1=0.                                                            371
      CD1=0.                                                            372
      DO 5 J=1,2                                                        373
      TOT=0.                                                            374
      TE=0.                                                             375
      DO 7 I=1,NR                                                       376
      Y=0.                                                              377
      K2=(J-1)*2                                                        378
      DO 6 K=1,2                                                        379
      L=IJ(K)                                                           380
      K1=I+(L-1)*NR                                                     381
      K2=K2+1                                                           382
    6 Y=Y+X(K1)*T(K2)                                                   383
      Y=Y**2                                                            384
      IF(Y)7,7,8                                                        385
    8 TE=TE-Y*ALOG(Y)                                                   386
    7 TOT=TOT+Y                                                         387
      CN1=CN1+TE                                                        388
      CD1=CD1-TOT*ALOG(TOT)                                             389
    5 CONTINUE                                                          390
      F=(CN+CN1+ASA)/(CD+CD1+ASA)                                       391
      RETURN                                                            392
       END                                                              393
```

```
C                                                                       394
C                                                                       395
C       ************                                                    396
        SUBROUTINE MINV(A,N,D,L,M)                                      397
C       ************                                                    398
C                                                                       399
C       SUBROUTINE TO INVERT A MATRIX TAKEN FROM IBM SYSTEM 360 SCIENTIFIC 400
C       SUBROUTINE PACKAGE (360A-CM-03X) VERSION III PROGRAMMER'S MANUAL 401
C       APPL PROG M 20-0205-3 ,TECH. PUBL. DEPT.,WHITE PLAINS,NEW YORK, 402
C       454 P.                                                          403
        DIMENSION A(1),L(1),M(1)                                        404
        D=1.0                                                           405
        NK=-N                                                           406
        DO 80 K=1,N                                                     407
        NK=NK+N                                                         408
        L(K)=K                                                          409
        M(K)=K                                                          410
        KK=NK+K                                                         411
        BIGA=A(KK)                                                      412
        DO 20 J=K,N                                                     413
        IZ=N*(J-1)                                                      414
        DO 20 I=K,N                                                     415
        IJ=IZ+I                                                         416
   10   IF( ABS(BIGA)- ABS(A(IJ))) 15,20,20                             417
   15   BIGA=A(IJ)                                                      418
        L(K)=I                                                          419
        M(K)=J                                                          420
   20   CONTINUE                                                        421
C                                                                       422
C          INTERCHANGE ROWS                                             423
C                                                                       424
        J=L(K)                                                          425
        IF(J-K) 35,35,25                                                426
   25   KI=K-N                                                          427
        DO 30 I=1,N                                                     428
        KI=KI+N                                                         429
        HOLD=-A(KI)                                                     430
        JI=KI-K+J                                                       431
        A(KI)=A(JI)                                                     432
   30   A(JI) =HOLD                                                     433
C                                                                       434
C          INTERCHANGE COLUMNS                                          435
C                                                                       436
   35   I=M(K)                                                          437
        IF(I-K) 45,45,38                                                438
   38   JP=N*(I-1)                                                      439
        DO 40 J=1,N                                                     440
        JK=NK+J                                                         441
        JI=JP+J                                                         442
        HOLD=-A(JK)                                                     443
        A(JK)=A(JI)                                                     444
   40   A(JI) =HOLD                                                     445
C                                                                       446
C          DIVIDE COLUMN BY MINUS PIVOT (VALUE OF PIVOT ELEMENT IS      447
C          CONTAINED IN BIGA)                                           448
C                                                                       449
   45   IF(BIGA) 48,46,48                                               450
   46   D=0.0                                                           451
        RETURN                                                          452
   48   DO 55 I=1,N                                                     453
        IF(I-K) 50,55,50                                                454
   50   IK=NK+I                                                         455
```

```
      A(IK)=A(IK)/(-BIGA)                                              456
   55 CONTINUE                                                         457
C                                                                      458
C        REDUCE MATRIX                                                 459
C                                                                      460
      DO 65 I=1,N                                                      461
      IK=NK+I                                                          462
      HOLD=A(IK)                                                       463
      IJ=I-N                                                           464
      DO 65 J=1,N                                                      465
      IJ=IJ+N                                                          466
      IF(I-K) 60,65,60                                                 467
   60 IF(J-K) 62,65,62                                                 468
   62 KJ=IJ-I+K                                                        469
      A(IJ)=HOLD*A(KJ)+A(IJ)                                           470
   65 CONTINUE                                                         471
C                                                                      472
C        DIVIDE ROW BY PIVOT                                           473
C                                                                      474
      KJ=K-N                                                           475
      DO 75 J=1,N                                                      476
      KJ=KJ+N                                                          477
      IF(J-K) 70,75,70                                                 478
   70 A(KJ)=A(KJ)/BIGA                                                 479
   75 CONTINUE                                                         480
C                                                                      481
C        PRODUCT OF PIVOTS                                             482
C                                                                      483
      D=D*BIGA                                                         484
C                                                                      485
C        REPLACE PIVOT BY RECIPROCAL                                   486
C                                                                      487
      A(KK)=1.0/BIGA                                                   488
   80 CONTINUE                                                         489
C                                                                      490
C        FINAL ROW AND COLUMN INTERCHANGE                              491
C                                                                      492
      K=N                                                              493
  100 K=(K-1)                                                          494
      IF(K) 150,150,105                                                495
  105 I=L(K)                                                           496
      IF(I-K) 120,120,108                                              497
  108 JQ=N*(K-1)                                                       498
      JR=N*(I-1)                                                       499
      DO 110 J=1,N                                                     500
      JK=JQ+J                                                          501
      HOLD=A(JK)                                                       502
      JI=JR+J                                                          503
      A(JK)=-A(JI)                                                     504
  110 A(JI) =HOLD                                                      505
  120 J=M(K)                                                           506
      IF(J-K) 100,100,125                                              507
  125 KI=K-N                                                           508
      DO 130 I=1,N                                                     509
      KI=KI+N                                                          510
      HOLD=A(KI)                                                       511
      JI=KI-K+J                                                        512
      A(KI)=-A(JI)                                                     513
  130 A(JI) =HOLD                                                      514
      GO TO 100                                                        515
  150 RETURN                                                           516
      END                                                              517
```

MINIMUM ENTROPY ROTATION

ARBUCKLE BRINE DATA - FIRST FOUR PRINCIPAL FACTORS

INITIAL FACTOR MATRIX

0.8667	-0.3192	0.0506	-0.0655
0.7482	-0.3418	-0.1126	-0.0811
0.2827	0.4587	0.0834	0.5449
-0.2062	-0.0498	-0.6663	0.4700
0.2528	0.6907	-0.3526	-0.0244
0.9051	-0.0967	-0.0543	-0.0043
0.7919	0.2839	0.0184	0.0244
0.8451	-0.0578	-0.0491	-0.0338
-0.0369	-0.3663	0.3888	0.7147
-0.7214	-0.3258	-0.3670	-0.0862
-0.3923	0.3040	0.5243	-0.1116

CRITERION VALUES

1	0.25163E 01
2	0.22956E 01
3	0.22612E 01
4	0.22546E 01
5	0.22539E 01
6	0.22539E 01
7	0.22539E 01

TRANSFORMATION MATRIX

0.9018	0.4201	0.0855	-0.0543
-0.3984	0.7557	0.1795	-0.4878
-0.0809	0.0576	0.8734	0.4768
-0.1465	0.4991	-0.4446	0.7293

ROTATED MATRIX

0.9143	0.0931	0.0901	0.0850
0.8319	0.0091	-0.0597	0.0133
-0.0144	0.7422	-0.0629	0.1981
-0.1810	0.0719	-0.8175	0.0606
-0.0151	0.5957	-0.1515	-0.5365
0.8598	0.3019	0.0145	-0.0310
0.5959	0.5605	0.1239	-0.1549
0.7941	0.2917	0.0340	-0.0657
-0.0235	0.0868	-0.0471	0.8872
-0.4784	-0.6135	-0.4023	-0.0398
-0.5010	0.0394	0.5286	0.0416

KANSAS GEOLOGICAL SURVEY COMPUTER PROGRAM
THE UNIVERSITY OF KANSAS, LAWRENCE

PROGRAM ABSTRACT

Title (If subroutine state in title):

MINIMUM ENTROPY CRITERION FOR ANALYTIC ROTATION

Date: June, 1969

Author, organization: Richard B. McCammon

Department of Geological Sciences, University of Illinois at Chicago

Direct inquiries to:

Name: Richard B. McCammon Address: Department of Geological Sciences

University of Illinois at Chicago

Purpose/description: To reduce a factor analytic or principal component data matrix to a state of minimum entropy using an orthogonal transformation

Mathematical method: A modified Jacobi procedure

Restrictions, range: The program is currently dimensioned to rotate twelve principal factors or components for fifty variables.

Computer manufacturer: IBM Model: 360/50

Programming language: FORTRAN IV

Memory required: 3 K Approximate running time:

Special peripheral equipment required:

Remarks (special compilers or operating systems, required word lengths, number of successful runs, other machine versions, additional information useful for operation or modification of program)

(continued from inside front cover)

11. FORTRAN IV program for vector trend analyses of directional data, by W.T. Fox, 1967 . $1.00
12. Computer applications in the earth sciences: Colloquium on trend analysis, edited by D.F. Merriam and N.C. Cocke, 1967 . . . . . $1.00
13. FORTRAN IV computer programs for Markov chain experiments in geology, by W.C. Krumbein, 1967 . . . . . $1.00
14. FORTRAN IV programs to determine surface roughness in topography for the CDC 3400 computer, by R.D. Hobson, 1967 . . . . . $1.00
15. FORTRAN II program for progressive linear fit of surfaces on a quadratic base using an IBM 1620 computer, by A.J. Cole, C. Jordan, and D.F. Merriam, 1967 . . . . . $1.00
16. FORTRAN IV program for the GE 625 to compute the power spectrum of geological surfaces, by J.E. Esler and F.W. Preston, 1967 . . . . . $0.75
17. FORTRAN IV program for Q-mode cluster analysis of nonquantitative data using IBM 7090/7094 computers, by G.F. Bonham-Carter, 1967 . . . . . $1.00
18. Computer applications in the earth sciences: Colloquium on time-series analysis, edited by D.F. Merriam, 1967 . . . . . $1.00
19. FORTRAN II time-trend package for the IBM 1620 computer, by J.C. Davis and R.J. Sampson, 1967 . . . . . $1.00
20. Computer programs for multivariate analysis in geology, edited by D.F. Merriam, 1968 . $1.00
21. FORTRAN IV program for computation and display of principal components, by W.J. Wahlstedt and J.C. Davis, 1968 . . . . . $1.00
22. Computer applications in the earth sciences: Colloquium on simulation, edited by D.F. Merriam and N.C. Cocke, 1968 . . . . . $1.00
23. Computer programs for automatic contouring, by D.B. McIntyre, D.D. Pollard, and R. Smith, 1968 . . . . . $1.50
24. Mathematical model and FORTRAN IV program for computer simulation of deltaic sedimentation, by G.F. Bonham-Carter and A.J. Sutherland, 1968 . . . . . $1.00
25. FORTRAN IV CDC 6400 computer program to analyze subsurface fold geometry, by E.H.T. Whitten, 1968 . . . . . $1.00
26. FORTRAN IV computer program for simulation of transgression and regression with continuous-time Markov models, by W.C. Krumbein, 1968 . . . . . $1.00
27. Stepwise regression and nonpolynomial models in trend analysis, by A.T. Miesch and J.J. Connor, 1968 . . . . . $1.00
28. KWIKR8 a FORTRAN IV program for multiple regression and geologic trend analysis, by J.E. Esler, P.F. Smith, and J.C. Davis, 1968 . . . . . $1.00
29. FORTRAN IV program for harmonic trend analysis using double Fourier series and regularly gridded data for the GE 625 computer, by J.W. Harbaugh and M.J. Sackin, 1968 . . $1.00
30. Sampling a geological population, by J.C. Griffiths and C.W. Ondrick, 1968 . . . $1.00
31. Multivariate procedures and FORTRAN IV program for evaluation and improvement of classifications, by Ferruh Demirmen, 1969. . . . . $1.00
32. FORTRAN IV programs for canonical correlation and canonical trend-surface analysis, by P.J. Lee, 1969 . . . . . $1.00
33. FORTRAN IV program for construction of Pi diagrams with the Univac 1108 computer, by Jeffrey Warner, 1969 . . . . . $1.00
34. FORTRAN IV program for nonlinear estimation, by R.B. McCammon, 1969 . . . . $0.75
35. FORTRAN IV computer program for fitting observed count data to discrete distribution models of binomial, Poisson and negative binomial, by C.W. Ondrick and J.C. Griffiths, 1969 . $0.75
36. GRAFPAC, graphic output subroutines for the GE 635 computer, by F.J. Rohlf, 1969 . . $1.00
37. An iterative approach to the fitting of trend surfaces, by A.J. Cole, 1969 . . . . $1.00
38. FORTRAN II programs for 8 methods of cluster analysis (CLUSTAN I), by David Wishart, 1969 $1.50
39. FORTRAN IV program for the generalized statistical distance and analysis of covariance matrices for the CDC 3600 computer, by R.A. Reyment, Hans-Ake Ramden, and W.J. Wahlstedt, 1969 . . . . . $1.00
40. Symposium on computer applications in petroleum exploration, edited by D.F. Merriam, 1969 $1.00
41. FORTRAN IV program for sample normality tests, by D.A. Preston, 1970 . . . . $1.00
42. CORFAN-FORTRAN IV computer program for correlation, factor analysis (R- and Q-mode) and varimax rotation, by C.W. Ondrick and G.S. Srivastava, 1970 . . . . $1.50
43. Minimum entropy criterion for analytic rotation, by R.B. McCammon, 1970 . . . $1.00