match no. 22 pp. 285-316 1987

PATTERN RECOGNITION TECHNIQUES IN INFRA-RED ANALYSIS. SARF SYSTEM

by R.Vancea[a], D.Ciubotariu[b], A.Graur[c],
St.Holban[b], G.Pentiuc[a], N. Pop[b]

a) Regional Computing Centre,
58oo Suceava - Romania

b) Computer Departament, The Politechnic
Institute, 19oo Timişoara - Romania

c) Institute for Higher Education
58oo Suceava - Romania

(Received: January 1987)

Abstract,

The paper presents directions for use and results of infra-red analyses, in determining the structure of chemical compounds by means of a pattern recognition system. The SARF System (Systѐme Automatique pour Reconnaissance des Formes, Sistem Automat pentru Recunoaşterea Formelor) implemented on a C 1o24 FELIX computer was used for data processing.

1. Introduction

Infra-red (IR) analysis, one of the methods in the study of physico-chemical properties of organic substances, is efficient both in monitoring chemical reactions as well as in finding out the structure of chemical compounds (obtained through synthesis or isolated from natural products).

The biunivocal relationship between the absorption frequency of IR radiation and the structure of a given molecular system helps to determine the structure, by taking into account the whole range of infra-red absorption frequencies. However the complexity of molecular compounds hinders the recognition process.

The following methods can be pointed out for computer analysis of chemical data:

- question-discovery methods, which helped to set up programs with algorithms similar to those used by a chemist in data analysis;
- revising methods, where the unknown IR spectrum is compared to spectra of known substances in the library;
- pattern recognition methods, used to classify objects in disjoint classes, according to some of their measured features.

2. Pattern Recognition Techniques

Pattern recognition usually means discrimination or classification of a set of objects, processes or events, irrespective of their nature. The set of object features is considered to supply information on one property of the object, indirectly measurable and, therefore, considered „obscure". Pattern recognition techniques try to establish relationships between patterns and the „obscure" property, resorting to no theory or „preconceived ideas".

The mathematical methods used to solve the problems of pattern recognition can be grouped in the following categories /1/:

- decision theoretical methods (statistibal)
- syntactic methods (linguistics)

In the first case the classification process takes into consideration a set of measurements, selected from the input pattern (Fig.1) described by N features. Pattern mathematical representation can be an X vector, with measured values of features, or a point in an N-dimensional space of the Ω_x features. These features are presumed invariants to the usual possible distortions. The following aspects can be pointed out in the recognition process:

a) extraction of essential features for the process under consideration. For practical reasons (measurement accessibility, necessary technical means, cost), the usual decision is fairly subjective at this stage. Unfortunately there has not yet been formulated a general theory of selecting the most illustrative features;

b) classification, i.e. adopting the decision of pattern circumscription to classes. The concept of pattern classification can be understood as a partition of space features. Pattern recognition is to determine the class a certain pattern is circumscribed to.

Discriminatory functions play a particular part in the process. Let $\omega_1, \omega_2, \ldots \omega_m$ be m classes of possible patterns, with the following properties:

$$\omega_1 \cup \omega_2 \cup \ldots \cup \omega_m = \Omega_x \qquad (1)$$

$$\omega_1 \cap \omega_2 \cap \cdots \cap \omega_m = F$$

where F is the set of points bordering the classes and

$$\bar{X} = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_N \end{bmatrix} \qquad (2)$$

the form vector, where x_i represents features.

The discriminatory function Dj(x), associated to ω_j pattern class, if the pattern represented by the x vector circumscribed to class ω_i (symbol $x \sim \omega_i$), has its greatest value Di(x). The following condition will be therefore satisfactory for all $x \sim \omega_j$:

$$Di(x) \geqslant Dj(x) \quad i,\ j = 1,\ \ldots,\ m, \quad i \neq j \qquad (3)$$

Thus the limits of partition of the X space features also known as decision limits, are:

$$F = Di(x) - Dj(x) = 0 \quad i,j = 1,\ldots,m \quad i \neq j \qquad (4)$$

2.1. The Linear Discriminatory Function

Di(x) function represents a linear combination of features x_1, x_2, ..., x_N, i.e.

$$Di(x) = \sum_{k=1}^{N} w_{ik}\, x_k + w_{i,N+1} \quad / \quad i = 1,\ldots, m \qquad (5)$$

and the decision limit between the areas in Ω_x, corresponding to classes ω_i and ω_j, assumes the following pattern:

$$D_i(x) - D_j(x) = \sum_{k=1}^{N} w_k\, x_k + w_{N+1} = o \qquad (6)$$

where $w_k = w_{ik} - w_{jk}$ and $w_{N+1} = w_{i,N+1} - w_{j,N+1}$;

equation (6) represents a hyperplane or decision plane.

2.2. Minimum Distance Classifier

This classifier evaluates distances between input pattern and a set of reference vectors (prototype points in the features space). Assuming that $R_1, R_2, \ldots, R_m$ (R_j is circumscribed to class ω_j) are m reference vectors, the minimum distance classifier will distribute x input pattern to class ω_i, if the distance between the pattern and the reference vectors of the class is minimum, i. e.

$$x \sim \omega_i \quad \text{if} \quad d = \left| X - R_i \right| \ \text{minimum.} \qquad (7)$$

Minkovscki distance evolution methods got pre-eminence:

$$d_{Minkovscki} = \left[\sum_{i=1}^{N} (x_i - y_i)^k \right]^{1/k} . \qquad (8)$$

For k = 2, the well - known Euclidian distance is obtained:

$$d_{Euclid} = \left[\sum_{i=1}^{N} (x_i - y_i)^2 \right]^{1/2} \qquad (9)$$

and for k = 1 Manhattan distance is obtained:

$$d_{Manhattan} = \sum_{i=1}^{N} (x_i - y_i) \qquad (1o)$$

If x_i and y_i features are binary encoded, Manhattan distance becomes Hamming distance equivalent to the number of different features in X and Y. Tanimato distance, a normalized Hamming distance, actually eliminates the disadvantages encountered in the case of a series of vectors with very few components having value "1"

$$d_{Tanimato} = \frac{AND(x_i,\ y_i)}{OR\ (\ x_i,\ y_i)} . \qquad (11)$$

These aspects and others are obvious in /1/, /2/ and /3/.

2.3. Classifier of the nearest vector

Assume that $R_1, R_2, \ldots, R_m$ are m sets of prototype vectors circumscribed to classes $\omega_1, \omega_2, \ldots, \omega_m$ respectively and let mark R_j vectors $R_j(k)$, i.e.

$$R_j^{(k)} \in R_j \;/\; k = 1, \ldots, u_j \tag{12}$$

u_j representing the number of reference vectors of the R_j set, circumscribed to ω_j class.

The distance between the input pattern represented by X vector and the R_j reference vectors set is

$$d(X, R_j) = \min \left| X - R_j^{(k)} \right| \tag{13}$$

$$j = 1, \ldots, m \quad \text{and } k = 1, \ldots, u_j$$

The classifier will distribute pattern X to the class represented by the set of reference vectors in proximity to X.

2.4. The Polynominal Discriminatory Function

Such a function of r order is given by the relation:

$$D_i(x) = w_{i1}+f_1(x)+w_{i2}f_2(x)+\ldots+w_{i2}f_2(x)+w_{i2}+1 \tag{14}$$

where

$$f_j(x) = x_{k_1}^{n_1} x_{k_2}^{n_2} \ldots x_{k_n}^{n_n} \quad \text{for } k_i = 1, \tag{15}$$

and n_i = 0 or 1. The decision limit between two classes assumes the pattern of a r order polynom.

2.5. Bayes Classifier

The aim of Bayes method is to find out an optimum decision of classification. The $x_1, x_2, \ldots, x_N$ features are considered aleatory variables, and for each ω_j class is kwown the distribution of multidimensional probability density of the X - $P(\omega_j)$ pattern vector, also known as class ω_j a priori probability.

The above information is used to define the particular conditions when the clasifier classifies with a minimum probability of erroneous recognition.Thus the problem of classification is formulated as a problem of statistic decision, where m statistic hypotheses are being tested by defining a decision function d(x), where d(x) = d_i means that H_i : $x \sim \omega_i$ hypothesis is accepted.

If $L(\omega_i, d_j)$ is the loss in case of an X input pattern for which d_j decision has been erroneous adopted which circumscribes it to class ω_i, the conditioned loss or risk is defined:

$$r(\omega_i, d) = \int_{\Omega x} L(\omega_i, d_j).p(x/\omega_i)dx. \quad (16)$$

For a given set of a priori probabilities

$P = \left\{ P(\omega_1), P(\omega_2), \ldots, P(\omega n) \right\}$ the average loss is:

$$R(P,d) = \sum_{i=1}^{m} P(\omega_i).\ r\ (\omega_i, d) \quad (17)$$

or

$$R(P,d) = \int_{\Omega x} P(x)\ .\ r_x(P,d).\ dx \quad (18)$$

where $r_x(P,d)$ represents the average loss a posteriori conditioned, when adopting decision d for a given X.

The problem is to take an optimum d_j decision, so as to have a minimum R(P,d) average risk, or in other words, to minimize the maximum of the conditioned average risk $r(\omega_i, d)$ - the minimax criterion.

If d is optimum decision, i.e. the average loss is minimum, then

$$r_x\ (P,\ d^*) \leqslant r_x(P,\ d) \quad (19)$$

i.e.

$$\sum_{i=1}^{m} L(\omega_i, d^*)\ \ P(\omega_i)\ p\ (x/\omega_i) \leqslant$$

$$\sum_{i=1}^{m} L(\omega_i, d)\ P(\omega_i)\ p\ (x/\omega_i) \quad (2o)$$

In case of a function of symmetrical loss (o,1) assuming the pattern:

$$L(\omega_i, d_j) = 1 - \theta_{ij} = \begin{cases} o & \text{for } i=j \\ 1 & \text{for } i \neq j \end{cases} \quad (21)$$

the average loss represents the probability of erroneous classification, and the rule of decision is:

$$d^* = d_i \qquad x \sim \omega_i$$

if

$$P(\omega_i)\ p(x/\omega_i) \geq P(\omega_j)\ p(x/\omega j) \qquad (22)$$

If the numerical ratios between classes ω_i and ω_j are defined

$$\lambda = \frac{p(x/\omega_i)}{p(x/\omega_j)} \qquad (23)$$

then equation (22) becomes

$$d^* = d_i \quad \text{if} \quad \lambda \geq \frac{P(\omega_j)}{P(\omega_i)} \qquad (24)$$

The discriminatory implemented function by Bayes classifier is

$$D_i(x) = P(\omega_i);\ \ p(x/\omega_i) \qquad i = 1, \ldots, m \qquad (25)$$

and the decision limits between areas in the space of Ω_x features circumscribed to ω_i classes are

$$P(\omega_i).\ \ p(x/\omega_i) - P(\omega_j).\ p(x/\omega_j) = 0 \quad i \neq j \qquad (26)$$

The main difficulties encountered in actual applications are in close connection to the a priority estimation of probabilities $P(\omega_i)$ and $p(x/\omega_i)$ where $p(x/\omega_i)$ distribution function of multidimensional probability assumes the pattern of Gauss normal function. Therefore, if the average vector is k_i, then:

$$p(x/\omega_i) = \frac{1}{(2\pi)^{N/2} |k_i|^{1/2}} \exp\left[- \frac{1}{2} (x-M_i)^{T} ki^{-1} . \right.$$
$$\left. . (x - M_i) \right] \qquad (27)$$

The a priori class probabilities are frequently often considered equal, $P(\omega_i) = 1/m$.

3. Classifier Formation and Evaluation

The term „formation" designates the series of methods and procedures that develop a classifier able to successfully circumscribe patterns to proper classes. With that end in view the patterns already circumscribed to classes are

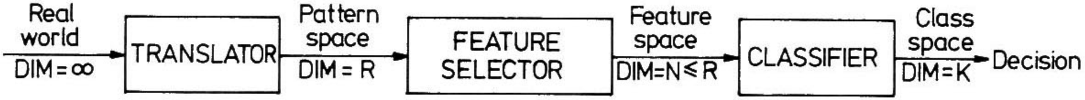

Fig.1 General Diagram of a Pattern Recognition System.

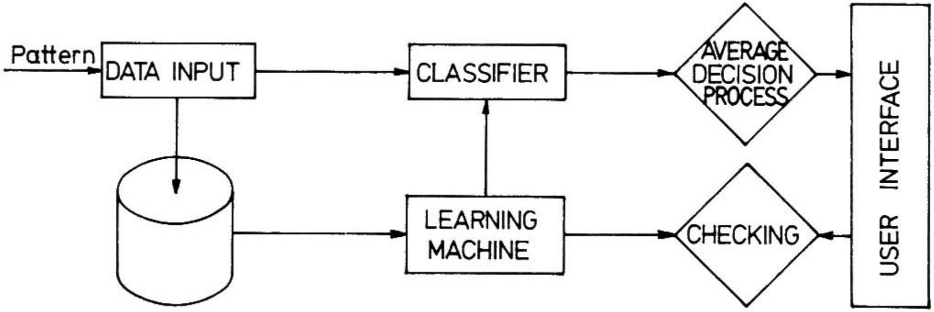

Fig. 2 SARF System.

aleatory divided into:

- formation set, used to develop a classifier able to better recognize the circumscription of the pattern set to proper classes;
- predicted set, to test the developed classifier.

The percentage of correctly classified patterns in the formation set is known as recognition ratio while the percentage of correctly classified patterns in the prediction set (patterns that were not used in the classifier formation) defines the predictive ability, both representing preliminary criteria of evaluation of a classifier performances.

A very good recognition of pattern circumscription to classes is possible when the correct values of weights $w_1, w_2, \dots, w_{N+1}$, decision vector $\overline{W}$ are given. As these values are not actually known, the problem is to evaluate the best values of weights, using the formation set by means of a feedback adjusting process.

Assuming that y is a transformed pattern vector

$$\overline{Y} = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_N \\ 1 \end{bmatrix} = \begin{bmatrix} \overline{X} \\ 1 \end{bmatrix} \qquad (28)$$

where $\overline{Y}$ is the ordinary pattern vector of the formation set. For simplification, let us assume that the $\overline{X}$ patterns in the formation set are classified in two classes (ω_1, ω_2).

The method of classifier formation shown above (except Bayes) is based on error correction, modifying $\overline{W}$ decision vector from step i - 1 as follows:

$$\overline{W}_i = \overline{W}_{i-1} \pm \alpha \; \overline{Y} \; (= \overline{W}') \qquad (29)$$

where α represents the correction coefficient possible to select through methods /1/:

- constant correction method, where α is any positive number, conveniently selected, constant throughout the formation process;

- absolute correction method, where α assumes the value of the smallest integer number for which the $\bar{Y} \cdot \bar{W}$ dot product is higher than the threshold value, i.e.

$$\alpha = \text{the smallest integer number} > \frac{|\bar{Y} \cdot \bar{W}|}{\bar{Y} \cdot \bar{Y}} \qquad (3o)$$

- fraction correction method, where   is selected so as:

$$\left|\bar{Y} \circ \bar{W} - \bar{Y} \cdot \bar{W}\right| = \lambda \left|\bar{Y} \cdot \bar{W}\right| ; \quad 0 \leqslant \lambda \leqslant 2 \qquad (31)$$

It is shown /1/, that these methods are converging towards an optimum decision vector, in a finite number of steps.

4. SARF SYSTEM

This system in FORTRAN and LISP (on FELIX Clo24, Iloo and CORAL computers) uses techniques of pattern recognition. The system is made up of basis modules (Fig.2): Data Acceptance Module, Classification Module, Average Decision Process and Learning Machine, each with a variable number of programs. The following table presents the SARF system programs. A description of the mathematical apparatus partially employed is appended.

Table

Program Name	Function
A. Data Acceptance Module	
1. INPUT	This program brings coded-record (normally, card or card-image) data into SARF. Some manipulation of the data will be done to make them compatible with the SARF system: category - type data will be arranged into category-groups and renamed by order of first encounter. Constant and redundant features will be removed and data flagged as "missing" will be assigned category (as data set) mean values.
2. CHANGE	This program provides a variety of feature category, pattern and file changes.
3. ANALYSIS	This program analyses the input data (member

of patterns within a class, dispersion, deviation, maximum and minimum value of a feature within a class).

4. DISTANCE This program calculates the different distance metrics. The distance matrix (in lower diagonal form) may be listed on the line printer.

5. COREL This program generates all feature-feature and feature-property correlations, with confidence intervals about the correlations and an estimate of the probability that the data could have come from an uncorrelated parent population. The interfeature covariance can also be listed.

6. WEIGHT This program evaluates the individual importance of each feature for the description of the property associated with the training set patterns.

B. Classification Module

7. DENDRO This program produces a "dendogram" which describes the hierarchical clustering of the member of training set patterns. The dendogram connects groups of patterns at levels of similarity.

8. KNN This program performs the Nearest Neighbor classification for category-type data, where K = 1 ÷ lo, "Nearness" is defined on the interpattern distance.

9. BAYES This program performs an approximate multivariant Bayes rule classification. It also produces the frequency histograms for each feature over each category and over all categories. Since the "true" probability distributions for each feature are presumed to be unknown, the frequency histograms are used in place of the probability distributions in the Bayes classification.

lo. TREE — This program generates a minimal spanning tree over the training set patterns. The spanning tree is then evaluated ("pruned") for self-consistent cluster of pattern.

11. SIMCA — This program classifies on the basis of pattern similarity to a principal component model of each category. The optimum member of principal components for each category may also be determined, using cross-validation.

12. KARLOV — This program performs the Karhunen-Loeve transformation on the training set data. New features are generated as linear combinations of the old features.The new features are linearly independent and are ordered according to decreasing variance.

C. Average Decision Process

13. DIALOG — This program eliminates patterns with the smallest / slightest prediction probability from the list of probable variants.

D. Learning Machine

14. MULTI — This program is a multicategory linear learning machine. Hyperplanes separating each category from all other patterns are iteratively developed, using negative feedback - training.

15. POTENTIAL — This program classifies according to potential functions.

The addendum presents the source list of Distance program.

5. Infra-red Spectral Analysis

Infra-red absorption spectra are vibrational-rotational spectra of which the simplest one have been computed through quantum - mechanical methods. The analysis of IR spectra due to polyatomic molecules is difficult for two reasons: first, because of a higher range of vibration and rotation possibilities and, secondly, because the same atoms can simultaneously participate in more then one vibration.

Each type of covalent bond has one or more characteristic frequencies which are only slightly influenced by other bands of the molecule. It is out of this information that "empiric spectroscopy" emerged and made possible the recognition of certain bands or groups in a molecule.

Pattern recognition methods resulted in the development of classifiers able to recognize classes of organic chemical compounds. Pattern vectors are directly computed from the IR spectrum, by dividing it in intervals of 0 . 1 μ m, each interval corresponding to an x_i component of the pattern vector. The number of intervals is approximately 13o.Absorption values corresponding to each interval in the spectrum were used to compute vector components by means of the following methods:

- the measured absorptions are the pattern vector components
- the digit numbers of components are undervalues
- IR spectra are binary encoded, each component assuming value 1 if it appears in the interval, and 0 if it does not.

Through the utilization of this last encoding / codification of "supervised learning", after the data processing by means of the SARF system, the authors got very good results both in the identification of the nature of urolithiasis / 5a, 5c, 9/ and in the recognition of the structure of heterocycles / 5b/. The results are given in the table below.

Class	Number of spectra	Total predictive ability			Classifiers
		13o features	25 features	7 features	
Urolithiasis monocomponent	25oo	1oo	-	1oo	BAYES,KNN,MULTI
Urolithiasis bicomponent	25oo	1oo	-	93	BAYES,KNN,MULTI
Heterocycles	35oo	1oo	88	-	BAYES,KNN,MULTI

As a result of characteristic selection it was possible a drastic reduction of the number of characteristics, by a proper selection of the characteristic fields in the IR spectrum, from 13o to 7 for urolithiasis and 25 for heterocycles.

The comparison of the three methods of classification (BAYES, KNN and MULTI) applied to the IR binary encoded spectra gave us the possibility to hierarchize them in the following decreasing order according to performances: the BAYES classifier based on distances > the Nearest K - neighbor classifier with Tanimoto or Hamming distance > MULTI classifier.

As a result of the heterocyles proceeding by means of the DENDRO "unsupervised learning" program we obtained a division of the pattern set in 2o classes. The examination of these divisions pointed out/made evident relationships between the patterns of certain classes, as they are subdivisions of a more comprehensive class. A second processing subdivided the set of forms in 12 classes.

The result of the last division was entirely satisfactory with the conclusion that the DENDRO program represents a most efficient computer assisted research method of some spectra with an unknown or partially known classification.

The forms of classification in the case of "unsupervised learning" represent a "training set" for the program of "supervised learning" which will finally induce the optimum classification method for the problem under consideration.

The use of binary classifiers for the recognition of 19 classes of compounds (the training set consists of 5oo spectra and the prediction set of 35oo) resulted in an average total predictive ability of 73-87% /4/. If the population in the 19 classes was significantly different, the results were unsatisfactory.

Lidell and Jurs /6/ obtained goods results for a set of only 212 IR spectra. A proper feature selection resulted in a total predictive ability of 92-98%.

Class	Total predictive ability (%)	
	128 features	lo features
Carboxylic acids	96	93
Esters	97	92
Primary amines	95	93

Iscuhor and others /7/ tested a series of classification methods in the case of binary encoding of IR spectra, using a library of 26oo spectra with 13 classes of compounds. The molecular formulas of compounds were of type $C_{1-15} H_x O_y N_z$, and the spectra were divided by Lowry /8/ into 13 intervals in the wavelength range 2-15.9 μ m.

The results in the case of the distance evaluation classifier had 9o% predictive ability for a discrimination between two classes and of 82% for the recognition of a class out of 13. Similar predictive abilities were obtained in the case of Bayes classifeer.

The results in the case of the use of linear discriminatory functions on areas are shown in the table below:

Number of spectra		Predictive abilities %		
Class 1	Class 2	P1	P2	P
2oo	24oo	55	96	76
2oo	2oo	86	66	76

The results are worse when 3 or 5 vectors are taken into account than in the case of one vector. The use of Tanimato distances yielded better results.

6. Conclusions

The methods of pattern recognition are extremely useful for the automatic recognition of organic structures based by IR spectra.

If the classifier formation and evaluation supposes the analysis of a large amount of data, which asks for the use of large computers, once the method of pattern recognition is chosen, it would be easily implemented on personal computers connected on-line to IR spectometers through digital - analog interface.

The spectrum library can be created and implemented on the external memory: the computer recognizes rapidly the class to which the analysed compound belongs. The confidence level of the distribution is measured by the predictive ability, analysis in the class is performed and an adequate program recognizes if the given spectrum shows features characteristic to one of the known classes of compounds.

Acknowledgement

We are much indebted to Professor A.T.Balaban,Professor M.Drăğănescu and Dr.V.Baltac for useful advice and suggestions they offered us both throughout the development of the SARF System as well as in writing the present paper.

Bibliography

1.K.S.Fu, Digital Pattern Recognition,Commun and Cyb.,vol.1o, Springer, Berlin, 198o

2.B.E.Batchelor, Practical Approch to Pattern Classification, Plenum, London, 1974

3.W.S.Meisel,Computer Oriented Approach to Pattern Recognition, Academic Press, New York, 1972

4.B.R.Kowalski, P.C.Jurs, T.L.Isenhour, C.N.Reilley, Anal. Chem. 41, 1945 (1969)

5.a) R.Vancea et al., Lucrările celui de al VII-lea Simpozion de Informatică Medicală, Timişoara, 1984, p.61-63; b)R.Vancea et al., Lucrările Sesiunii Jubiliare,Iaşi,1986, p.9-21; R.Vancea et al., Lucrările celui de al VII-lea Simpozion de Informatică Medicală, Timişoara, 1984, p.95-98

6.a) R.W.Lidell, P.C.Jurs, App.Spectrosc., 27, 371 (1973); b) Anal. Chem., 46, 2126 (1974)

7.a)H.B.Woodruff, S.R.Lowry, T.L.Isenhour, Anal. Chem., 46, 215o (1974); b) Appl.Spectrosc., 29, 226 (1975); c) H.B. Woodruff, G.L. Ritter, S.R. Lowry, T.L. Isenhour, Appl. Spectrosc., 3o, 213 (1976)

8.S.R.Lowry, T.L. Isenhour, J.Chem. Inf. Comput. Sci., 15, 212 (1975)

9.R.Vancea et al., Info-Iaşi '83, p. 491-51o

1o.SARF System Manual

```
      INTEGER C,S
      COMMON D(5000),X(1250),Y(1250)
      COMMON /PATT/NVAR,NCAT,NPAT,NTEST,NTOT,NPEING,NPEJNG,
     1INDO,INDLO,J,DNAX,DAVG,C,S,NGR,X1,X2,PDD,PSS,ISALT,APN,ANX
      CALL INITDS(&100)
      CALL DISTANTE
C
CACCEPT THE USER'S PARAMETERS
C
CPROCESSING FILE -5-  FOR COMPUTING THE REQUIRED DISTANCE
C

C
      IF(C.EQ.-1) CALL PSEUDOSD
C
C ...     IF S=12 THEN COMPUTE S(X,Y)=1-D(X,Y)/DNAX
C
C
      IF(S.EQ.12) CALL PSEUDOSD
      IF(S.GE.1.AND.C.NE.-1) NGR=-999
      CALL FINALDS
 100  STOP '*PSG81*'
      END
      SUBROUTINE INITDS(*)
      COMMON D(5000),X(1250),Y(1250)
      COMMON/PATT/NVAR,NCAT,NPAT,NTEST,NTOT,NPEING,NPEJNG,INDO,
     1INDLO,J,DAVG,DMAX,DD,SS,NGR,X1,X2,PD,PS,ISALT,APN,ANX
      DATA NOUNEW/'NOU'/
      I16=16
      I128=128
      READ (30) KEY,NPAT,NTEST,NVAR,NCAT,(NA,J=1,NVAR),(NA,J=1,NVAR)
      IF(KEY.EQ.NOUNEW) RETURN 1
      CALL CITPARDI(&100)
      DAVG=0
      DMAX=0
      NTOT=NPAT+NTEST
      NPEING=(NVAR+4)/I16
      IR=NVAR+4-NPEING*I16
      IF(IR.NE.0) NPEING=NPEING+1
      NPEJNG=NPAT/I128
      IR=NPAT-NPEJNG*I128
      IF(IR.NE.0) NPEJNG=NPEJNG+1
      INDO=1-NPEING
      INDLO=1-NPEJNG
      IND=INDO
      DO 1 I=1,NTOT
      READ (30) ID,NN1,NN2,CN,(X(K),K=1,NVAR)
      IND=IND+NPEING
      WRITE(46'IND,46) ID,NN1,NN2,CN,(X(K),K=1,NVAR)
   46 FORMAT(16A4)
    1 CONTINUE
      RETURN
  100 RETURN 1
      END
      SUBROUTINE CITPARDI(*)
      COMMON /PATT/NVAR,NCAT,NPAT,NTEST,NTOT,NPEING,NPEJNG,INDO,
     1INDLO,J,DMAX,DAVG,DD,SS,NGR,X1,X2,PDD,PSS,ISALT,APN,ANX
C
C                 C I T P A R D I = ACCEPT PARAMETER CARD
C        DD =    DISTANCE TYPE
C        NGR=    GENERAL MAHALANOBIS DISTANCE ORDER
C
C        PDD=    IF NOT ZERO  DISTANCE MATRIX IS PRINTED
```

```
C
C      SS =   TYPE OF SIMILARITY COEFICIENT
C
C      PSS=   IF NOT ZERO SIMILARITY MATRIX IS PRINTED
C              (UPPER DIAGONAL MATRIX ONLY)
C      X1 =   THE FIRST VALUE WHEN D=4 (RATIO DISTANCE OF O.U. ANDERS)
C
C      X2 =   THE SECOND VALUE
C
      COMMON C(5000),X(1250),Y(1250)
      DATA IREAD,X1IMPL,X2IMPL,IWRITE/105,0.66666666,1.5,108/
      INTEGER DD,SS,PDD,PSS,D,S
      EQUIVALENCE (D,DD),(S,SS),(N,NGR)
      READ(IREAD,101) DD,NGR,PDD,SS,PSS,X1,X2
      WRITE(IWRITE,201) DD,NGR,PDD,SS,PSS,X1,X2
C
C   ... COMPUTE  -ISALT-
C
      IF(S.NE.0) GO TO 1
C  ...    S=0
11    IF(D.EQ.1) GO TO 3
      IF(D.LE.0.OR.D.GT.4) GO TO 4
C
C  ...   S = 0  AND D CORRECT
C
      IF(D.NE.2) GO TO 5
C
C  ...   D = 2  ...
C
      IF(N.EQ.2) GO TO 6
      IF(N.LE.1.OR.N.GT.23) GO TO 7
      ISALT=2
      RETURN
C
C           D=1  CHECK FOR BINARY FEATURES
C
  3   READ(30) ID,NA,NB,CN,(X(K),K=1,NVAR)
      DO 30 K=1,NVAR
      IF(X(K).NE.0.AND.X(K).NE.1) GO TO 8
  30  CONTINUE
      REWIND 30
      READ (30) IKEY
      ISALT=1
      GO TO 999
  8   WRITE(IWRITE,202)
  202 FORMAT(' *** ERROR : FEATURES NOT BINARY')
      GO TO 1000
  7   WRITE(IWRITE,203)
  203 FORMAT(' ERROR : ILEGAL VALUE FOR "N"'/5X,
     1'"N" FIXED TO 2'/)
      N=2
  6   ISALT=5
      GO TO 999
  5   IF(N.EQ.0) N=1
      IF(D.EQ.4) GO TO 14
      ISALT=D
      GO TO 999
  4   WRITE(IWRITE,204)
  204 FORMAT('*** ERROR : ILEGAL VALUE FOR DISTANCE TYPE'/)
      GO TO 1000
C      ***  S  NOT ZERO
```

```
  1    IF(D.GT.0) GO TO 11
       IF(S.LT.1.OR.S.GE.12) GO TO 2
       IF(S.EQ.11) GO TO 9
C
C   ***  BINARITY CHECK
C
       READ (30) ID,NA,NB,CN,(X(K),K=1,NVAR)
       DO 100 K=1,NVAR
       IF(X(K).NE.0.AND.X(K).NE.1) GO TO 8
 100   CONTINUE
       REWIND 30
       READ (30) IKEY
  9    ISALT=5+S
       GO TO 999
  2    IF(S.EQ.12) GO TO 10
       WRITE(IWRITE,205)
 205   FORMAT(' *** ERROR : ILEGAL PARAMETER VALUE =S='/)
       GO TO 1000
 10    D=2
       NOR=2
       ISALT=5
       WRITE(IWRITE,207)
 207   FORMAT('*** D AND N ARE FIXED TO 2'/)
       GO TO 999
C
C ...    CASE D=4
C
 14    IF(X1.NE.0.OR.X2.NE.0) GO TO 15
       X1=X1IMPL
       X2=X2IMPL
 15    ISALT=4
 999   APN=1./NVAR
       ANX=NVAR*(NVAR-1)
       RETURN
1000   WRITE(IWRITE,208)
208    FORMAT('*** EXECUTION ABORTED')
       RETURN 1
101    FORMAT(5I5,3F10.6)
201    FORMAT('  D= ',I5,5X,' N= ',I5,5X,'PD=',I5,5X,'S= ',I5,
      15X,'PS=',I5,5X,'X1=',F11.6,5X,'X2=',F11.6/)
       END
       FUNCTION FS(X,Y)
       DIMENSION X(1),Y(1)
       COMMON/PATT/NVAR,AUX(21)
C
C      F S =  COMPUTE THE NUMBER  OF COMMON ATTRIBUTES
C             OF PATTERNS -X- AND -Y-
C
       FS=0
       DO 1 K=1,NVAR
       FS=FS+X(K)*Y(K)
 1     CONTINUE
       RETURN
       END
       FUNCTION FT(X,Y)
       DIMENSION X(1),Y(1)
       COMMON/PATT/NVAR,AUX(21)
C
C        F T= COMPUTE THE NUMBER OF ATTRIBUTES
C               THAT PATTERN  -X- HASN'T AND PATTERN -Y- HASN'T TOO
```

```
C
      FT=0
      DO 1 K=1,NVAR
      IF(X(K).EQ.0.AND.Y(K).EQ.0) FT=FT+1
 1    CONTINUE
      RETURN
      END
      FUNCTION FU(X,Y)
      DIMENSION X(1),Y(1)
      COMMON /PATT/NVAR,AUZ(21)
C
C       F U  = COMPUTE THE NUMBER OF ATTRIBUTES THAT THE PATTERN X HAS
C                AND THE PATTERNS Y HASN´T
C
      FU=0
      DO 1 K=1,NVAR
      IF(X(K).EQ.1.AND.Y(K).EQ.0) FU=FU+1
 1    CONTINUE
      RETURN
      END
      FUNCTION FV(X,Y)
      DIMENSION X(1),Y(1)
      COMMON /PATT/NVAR,AUX(21)
C
C      F V = COMPUTE THE NUMBER OF ATTRIBUTES THAT PATTERN Y HAS
C              AND X DIDN´T
C
      FV=0
      DO 1 K=1,NVAR
      IF(X(K).EQ.0.AND.Y(K).EQ.1) FV=FV+1
 1    CONTINUE
```

```
      RETURN
      END
      SUBROUTINE SIM11
C
C     S I M 1 1 =  COMPUTE THE SIMILARITY COEFFICIENT TYPE 11 (COS(X,Y)
C
      COMMON/PATT/NVAR,IUX(8),J,AUX(12)
      COMMON D(5000),X(1250),Y(1250)
      SXY=0
      SX2=0
      SY2=0
      DO 1 K=1,NVAR
      XK=X(K)
      YK=Y(K)
      SXY=SXY+XK*YK
      SX2=SX2+XK*XK
      SY2=SY2+YK*YK
 1    CONTINUE
      D(J)=SXY/SQRT(SX2)/SQRT(SY2)
      RETURN
      END
      SUBROUTINE PSEUDOSD
C
C   PSEUDOSD = COMPUTE SIMILARITY COEFFICIENT TYPE 12
C                ON THE BASIS OF DISTANCE PREVIOUS COMPUTED OR
C                PSEUDODISTANCE    D(X,Y)=1-S(X,Y)/DMAX
      COMMON D(5000),X(1250),Y(1250)
      COMMON /PATT/NVAR,NCAT,NPAT,NTEST,NTOT,NPETNG,NPCJNG,INDO,
     1INDLO,J,DMAX,DAVG,C,S,NGR,X1,X2,PS,ISALT,APN,ANX
      DAVG=0
      DMAXN=-1.E+10
      IND=INDLO
      DO 1 NP=1,NTOT
      IND=IND+NPEJNG
      READ(55'IND,55) (D(K),K=1,NPAT)
  55  FORMAT(128A4)
      DO 1 K=1,NPAT
      D(K)=1.-D(K)/DMAX
      DK=D(K)
      IF(DMAXN.LT.DK) DMAXN=DK
      DAVG=DAVG+DK
   2  CONTINUE
   1  CONTINUE
      DMAX=DMAXN
      RETURN
      END
      SUBROUTINE DISTD3
      COMMON /PATT/NVAR,A(8),J,B(4),N,C(5),APN,ANX
      COMMON D(5000),X(1250),Y(1250)
C
C       D I S T D 3 = COMPUTE THE CITY BLOCK DISTANCE
C
       S=0
       DO 1 K=1,NVAR
       S=S+ABS(X(K)-Y(K))
   1   CONTINUE
       D(J)=S
       RETURN
       END
       SUBROUTINE DISTD2N
```

```
        COMMON /PATT/NVAR,A(8),J,B(4),N,C(5),APN,ANX
        COMMON D(5000),X(1250),Y(1250)
C
C     D I S T D 2 N = COMPUTE GENERAL MAHALANOBIS DISTANCE OF ORDER N
C
        S=0
        DO 1 K=1,NVAR
        S=S+(X(K)-Y(K))**N
 1      CONTINUE
        D(J)=S**APN
        RETURN
       END
       SUBROUTINE DISTD4
C
C    D I S T D 4 = COMPUTE THE RATIO DISTANCE OF O.U. ANDERS
C
       COMMON/PATT/NVAR,AUX(8),J,B(5),X1,X2,C(3),APN,ANX
       COMMON D(5000),X(1250),Y(1250)
       S=0
       DO 1 I=2,NVAR
       K1=K-1
       XI=X(I)
       YI=(I)
       DO 2 K=1,K1
       RIJ=XI*Y(K)/X(K)/YI
       IF(RIJ.LT.X1.OR.RIJ.GT.X2) S=S+1
  2    CONTINUE
  1    CONTINUE
       D(J)=2*S/ANX
       RETURN
```

```
      END
      SUBROUTINE DISTEUCL
C
CD I S T E U C L  -  COMPUTE THE EUCLIDIAN DISTANCE
C
      COMMON D(5000),X(1250),Y(1250)
      COMMON/PATT/NVAR,NCAT,NPAT,NTEST,NTOT,NPEING,NPEJNG,INDO,
     1INDLO,J,DMAX,DAVG,AUX123(10)
      DIST=0
      DO 1 K=1,NVAR
      DIF=X(K)-Y(K)
      IF(DIF.EQ.0) GO TO 1
      DIST=DIST+DIF*DIF
  1   CONTINUE
      DIST=SQRT(DIST)
      D(J)=DIST
      RETURN

      END
      SUBROUTINE FINALDS
C
C     CREATE THE FILE DISTANCE
C     AND LISTED ON LINE PRINTER THE UPPER DIAGONAL
C                         DISTANCE OR SIMILARITY  MATRIX
C
      COMMON D(5000),X(1250),Y(1250)
      COMMON /PATT/NVAR,NCAT,NTEST,NTOT,NPEING,NPEJNG,INDO,INDLO,
     1J,DMAX,DAVG,DD,SS,NGR,X1,X2,PDD,PSS,ISALT,APN,ANX
      DATA IKEY/'DISM'/
      EQUIVALENCE(N,NGR)
      DATA IW/108/
      ILIST=0
     IF(PSS.NE.0.AND.DD.EQ.0) ILIST=1
     IF(PDD.NE.0.AND.DD.NE.0) ILIST=1
     IF(ILIST.EQ.0) GO TO 2
     IF(SS.NE.0.AND.DD.EQ.0) WRITE(IW,101) DAVG,DMAX
     IF(DD.GT.0.AND.SS.EQ.0) WRITE(IW,102)DAVG,DMAX
     IF(DD.EQ.-1) WRITE(IW,102) DAVG,DMAX
     IF(SS.EQ.12) WRITE(IW,101) DAVG,DMAX
101  FORMAT(//'**** DISTANCE MATRIX *** AVERAGE=',G14.6,
    1'    **** MAXIMUM =',G14:6,'    ***'/)
102  FORMAT(//'**** SIMILARITY COEFICIENT MATRIX  ',
    1'***  AVERAGE =',G14.6,'    ***   MAXIMUM =',G14.6,'    *'
    2'**'/)
 2   NOUT=31
     DAVG=DAVG/NTOT/NPAT
     REWIND NOUT
     WRITE (NOUT) IKEY,NPAT,NTEST,NVAR,NCAT,(NN1,J=1,NVAR),
    1(NN2,J=1,NVAR),N,N,N,DAVG,DMAX
     IND=INDO
     INDL=INDLO
     DO 1 I=1,NTOT
     IND=IND+NPEING
     INDL=INDL+NPEING
     READ(46'IND,46) ID,NN1,NN2,CN,(X(K),K=1,NVAR)
     READ(55'INDL,55) (D(K),K=1,NPAT)
     WRITE(NOUT) ID,NN1,NN2,CN,(X(K),K=1,NVAR),(D(K),K=1,NPAT)
 46  FORMAT(16A4)
 55  FORMAT(128A4)
     IF(ILIST.EQ.0.OR.I.GT.NPAT) GO TO 1
     K=I+1
     WRITE(IW,103) 1,NN1,NN2,(JK,D(JK),JK=K1,NPAT)
 103  FORMAT('**',I5,1X,2A4,'*',5(1X,I5,2X,G14.6)/,
     1(18X,I6,2X,G14.6,I6,2X,G14.6,I6,2X,G14.6,I6,2X,
     2G14.6,I6,2X,G14.6))
 1    CONTINUE
      RETURN
      END
      SUBROUTINE DISTANTE
C
CMONITORING THE DISTANCE COMPUTATION
C
      COMMON D(5000),X(1250),Y(1250)
      COMMON/PATT/NVAR,NCAT,NPAT,NTEST,NTOT,NPEING,NPEJNG,INDO,
     1INDLO,J,DMAX,DAVG,DD,SS,NRG,X1,X2,PDD,PSS,ISALT,APN,ANX
      IND=INDO
      INDL=INDLO
      DO 1 I=1,NTOT
      IND=IND+NPEING
      READ(46'IND,46) ID,NN1,NN2,CN,(X(K),K=1,NVAR)
  46  FORMAT(16A4)
      INDJ=INDO
      DO 2 J=1,NPAT
      INDJ=INDJ+NPEING
      READ(46'INDJ,46) ID,NM1,NM2,CN,(Y(K),K=1,NVAR)
      GO TO (101,102,103,104,105,106,107,108,109,110,111,112,113,
     1114,115,116) ISALT
C
C   ...  CONDITIONAL JUMP  - ISALT -
C
C
 101  D(J)=FS(X,Y)
```

```
      GO TO 199
  102 CALL DISTD2N
      GO TO 199
  103 CALL DISTD3
      GO TO 199
  104 CALL DISTD4
      GO TO 199
  105 CALL DISTEUCL
      GO TO 199
C
C ...    RUSSEL AND RAO COEFICIENT (1940)
C
  106 D(J)=FS(X,Y)
      GO TO 199
C
C ...    KENDAL COEFICIENT (1958)
C
  107 D(J)=1-(FU(X,Y)+FV(X,Y))/NVAR
      GO TO 199
C
C ...    ROGER AND TANIMOTO COEFICIENT (1960)
C
  108 SUV=FU(X,Y)+FV(X,Y)
      D(J)=(NVAR-SUV)/(NVAR+SUV)
      GO TO 199
C
C ...     JACCARD COEFICIENT (1908)
C
  109 SXX=FS(X,Y)
      D(J)=SXX/(SXX+FU(X,Y)+FV(X,Y))
      GO TO 199
C
C ...  KULZINSKY COEFICIENT (1927)
C
 110   D(J)=FS(X,Y)/(FU(X,Y)+FV(X,Y))
      GO TO 199
C
C ...  OCHIAI COEFICIENT (1957)
C
 111  SXX=FS(X,Y)
      D(J)=SXX/SQRT((SXX+FU(X,Y))*(SXX+FV(X,Y)))
      GO TO 199
C
C ...   DICE COEFICIENT  (1945)
C
 112  SXX=FS(X,Y)
      D(J)=SXX/(SXX+(FU(X,Y)+FV(X,Y))/2.)
      GO TO 199
C
C ...  KULEZINSKY COEFICIENT  2
C
 113  SXX=FS(X,Y)
      D(J)=(SXX/(SXX+FU(X,Y))+SXX/(SXX+FV(X,Y)))/2.
      GO TO 199
C
C ...  PEARSON COEFICIENT
C
 114  SXX=FS(X,Y)
       TXX=FT(X,Y)
      UXX=FU(X,Y)
      VXX=FV(X,Y)
      D(J)=(SXX*TXX-UXX*VXX)/SQRT((SXX+VXX)*(SXX+UXX))/
     1SQRT((TXX+UXX)*(TXX+VXX))
      GO TO 199
C
C ...  YULE COEFFICIENT (1911)
C
 115  STX=FS(X,Y)*FT(X,Y)
      UVX=FU(X,Y)*FV(X,Y)
      D(J)=(STX-UVX)/(STX+UVX)
C
C ...  COSINUS
C
      GO TO 199
 116  CALL SIM11
 199  DAVG=DAVG+D(J)
      IF(DMAX.LT.D(J)) DMAX=D(J)
 2    CONTINUE
      INDL=INDL+NPEJNG
      WRITE(55'INDL,55) (D(K),K=1,NPAT)
 55   FORMAT(128A4)
  1   CONTINUE
      RETURN
      END
```