A Nonlinear Mapping for Data Structure Analysis

JOHN W. SAMMON, JR.

Abstract—An algorithm for the analysis of multivariate data is presented along with some experimental results. The algorithm is based upon a point mapping of N L-dimensional vectors from the L-space to a lower-dimensional space such that the inherent data "structure" is approximately preserved.

Index Terms—Clustering, dimensionality reduction, mappings, multidimensional scaling, multivariate data analysis, nonparametric, pattern recognition, statistics.

Introduction

THE purpose of this paper is to describe the nonlinear mapping algorithm (NLM) which has been found to be highly effective in the analysis of multivariate data. The analysis problem is to detect and identify "structure" which may be present in a list of N L-dimensional vectors. Here the word structure refers to geometric relationships among subsets of the data vectors in the L-space. Some examples of structure are hyperspherical and hyperellipsoidal clusters, and linear and certain nonlinear relationships among the vectors of some subset.

The algorithm is based upon a point mapping of the N L-dimensional vectors from the L-space to a lower-dimensional space such that the inherent structure of the data is approximately preserved under the mapping. The approximate structure preservation is maintained by fitting N points in the lower-dimensional space such that their interpoint distances approximate the corresponding interpoint distances in the L-space. We shall be primarily interested in mappings to 2- and 3-dimensional spaces since the resultant data configuration can easily be evaluated by human observations in 3 or less dimensions.

The Nonlinear Mapping

Suppose that we have N vectors in an L-space designated X_i, $i=1, \cdots, N$ and corresponding to these we define N vectors in a d-space ($d=2$ or 3) designated Y_i, $i=1, \cdots, N$. Let the distance[1] between the vectors X_i and X_j in the L-space be defined by $d_{ij}^* \equiv \text{dist}[X_i, X_j]$ and the distance between the corresponding vectors Y_i and Y_j in the d-space be defined by $d_{ij} \equiv \text{dist}[Y_i, Y_j]$.

Let us now randomly[2] choose an initial d-space configuration for the Y vectors and denote the configuration as follows:

$$Y_1 = \begin{bmatrix} y_{11} \\ \cdot \\ \cdot \\ \cdot \\ y_{1d} \end{bmatrix} \quad Y_2 = \begin{bmatrix} y_{21} \\ \cdot \\ \cdot \\ \cdot \\ y_{2d} \end{bmatrix} \cdots Y_N = \begin{bmatrix} y_{N1} \\ \cdot \\ \cdot \\ \cdot \\ y_{Nd} \end{bmatrix}.$$

Next we compute all the d-space interpoint distances d_{ij}, which are then used to define an error E, which represents how well the present configuration of N points in the d-space fits the N points in the L-space, i.e.,

$$E = \frac{1}{\sum_{i<j} [d_{ij}^*]} \sum_{i<j}^{N} \frac{[d_{ij}^* - d_{ij}]^2}{d_{ij}^*}. \tag{1}$$

Note that the error is a function of the $d \times N$ variables y_{pq}, $p=1, \cdots, N$ and $q=1, \cdots, d$. The next step in the NLM algorithm is to adjust the y_{pq} variables or equivalently change the d-space configuration so as to decrease the error. We use a steepest descent procedure to search for a minimum of the error (see Appendix I for further details).

Some Computer Results

We have exercised the nonlinear mapping algorithm on several data sets in order to test and evaluate the utility of the program in detecting and identifying structure in data. Some of the results obtained for several different artificially generated data sets[3] are reported for the case where $d=2$. We have also run the algorithm on many real data sets and have achieved highly satisfactory results; however, for demonstration purposes it is useful to work with artificially generated data in order that we can compare our results with the known data structure. The test data sets were as follows.

1) *Straight Line Data:* These data consisted of nine points distributed along a line in a 9-dimensional space. The data points were spaced evenly along the line with an interpoint Euclidean distance of $\sqrt{9}$ units. The initial 2-space configuration was chosen randomly.

Manuscript received August 26, 1968; revised February 2, 1969.

The author was with Rome Air Development Center, Griffiss AFB, Rome, N. Y. He is now with Computer Symbolic, Inc., Rome, N. Y.

[1] Any distance measure could be used; however, if we have no a priori knowledge concerning the data, we would have no reason to prefer any metric over the Euclidean metric. Thus, this algorithm uses the Euclidean distance measure.

[2] For the purpose of this discussion it is convenient to think of the starting configuration as being selected randomly; however, in practice the initial configuration for the vectors is found by projecting the L-dimensional data orthogonally onto a d-space spanned by the d original coordinates with the largest variances.

[3] One exception is data set 3 which is a classical data set. This data set was not artificially generated.

Authorized licensed use limited to: New York University. Downloaded on September 03,2022 at 02:10:10 UTC from IEEE Xplore. Restrictions apply.

2) *Circular Data:* The data consisted of nine points, eight of which were spaced evenly (45° apart) along a circle of radius 2.5 units in a 2-dimensional space. The ninth point was placed at the center of the circle. The initial 2-space configuration was chosen randomly.

3) *Iris Data:* This data set is fairly well-known since it was used by Fisher [4] in several statistical experiments. The data were originally obtained by making four measurements on Iris flowers. These measurements were then used to classify three different species of Iris flowers. Fifty sample vectors were obtained from each of the three species. Thus, the data set consists of 150 points distributed in a 4-dimensional space.

4) *Gaussian Data Distributed at the Vertices of a Simplex:* These data consist of 75 points distributed in a 4-dimensional space. There are five spherical Gaussian distributions which have their respective mean vectors located at the vertices of a 4-dimensional simplex.[4] The intervertex distance is $\sqrt{5/4}$ units and each covariance matrix is diagonal with a standard deviation along every coordinate of 0.2 unit. Fifteen points were generated from each of the five Gaussian distributions, making a total of 75 points.

5) *Helix Data:* This data set consisted of 30 points distributed along a 3-dimensional helix. The parametric equations for this helix are

$$X = \cos Z$$
$$Y = \sin Z$$
$$Z = \frac{\sqrt{2}}{2} t.$$

The points are distributed at one-unit intervals along the curve (i.e., $t = 0, 1, 2, \cdots, 29$).

6) *Nonlinear Data:* This data set consisted of 29 points distributed evenly along a 5-dimensional curve. The parametric equations for this curve are

$$X = \cos Z$$
$$Y = \sin Z$$
$$U = 0.5 \cos 2Z$$
$$V = 0.5 \sin 2Z$$
$$Z = \frac{\sqrt{2}}{2} t.$$

The points were distributed at one-unit intervals along the curve (i.e., $t = 0, 1, \cdots, 28$).

Figs. 1 through 9 display the results obtained using the nonlinear mapping algorithm. Convergence was essentially obtained for each case in twenty or less iterations using the gradient searching technique described in Appendix I. As was expected, the data structure inherent in both data sets 1 and 2 was faithfully reproduced under the mapping (see Figs. 1 and 2). This was expected since the data sets are 1- and 2-dimensional, respectively, and therefore a mapping to a 2-space can be accomplished with zero error. In both cases the final error was 10^{-16}.

Observing the result of the Iris data mapping (Fig. 3), we can essentially detect the three species of Iris. The final error was 2×10^{-3} which is considered quite small.

The results obtained on the simplex data (data set 4) were quite interesting. The result of the mapping clearly showed the presence of five clusters (see Fig. 4). However, when we compare this result with the projection of the same data onto the 2-space defined by the two largest eigenvectors of the estimated data covariance matrix, we can only detect four clusters. Two of the clusters overlap completely in the 2-space which fits the data in the least squares sense (see Fig. 5). The resulting NLM error was 0.05. The same experiment has been conducted using Gaussian data distributed at the vertices of higher-dimensional simplexes. Figs. 6 and 7 show the NLM and principal eigenvector plots, respectively, for a 19-dimensional Gaussian simplex distribution. These experiments indicate that for some data sets, the NLM is superior to eigenvector projections for data structure analysis.

The results shown in Figs. 8 and 9 clearly indicate the "string structure" in data sets 5 and 6, respectively. The mapping error for the helix data was 6×10^{-4}. The error for data set 6 was 1.6×10^{-3}.

The utility of any data analysis technique is somehow more convincing when applied to "real" data as opposed to artificially generated data, presuming, of course, that the analysis results are correct. For this reason, the application of the NLM algorithm to an experiment in document retrieval by content is reported here.

The experiment, conducted jointly by Rome Air Development Center (RADC) and the University of Colorado, involved the construction of a document classification space (referred to as the *C*-space) where every document in the library was represented as a 17-dimensional vector. The construction technique devised by Ossorio [8], [9] describes a mapping of 1125 preselected words and phrases into the *C*-space. Documents, or equivalently retrieval requests, were located in the space by computing the vector average of the corresponding key words or phrases which were contained in the document or the request. Retrieval was accomplished by rank-ordering the relevance of the library documents to a given request. The relevance measure was computed using the Euclidean metric between the document vectors and the request vector, the concept being that document vectors which are close in the *C*-space are related by content and therefore should be retrieved together.

[4] The vertices of a simplex are all equidistant from one another as well as from the origin.

Authorized licensed use limited to: New York University. Downloaded on September 03,2022 at 02:10:10 UTC from IEEE Xplore. Restrictions apply.

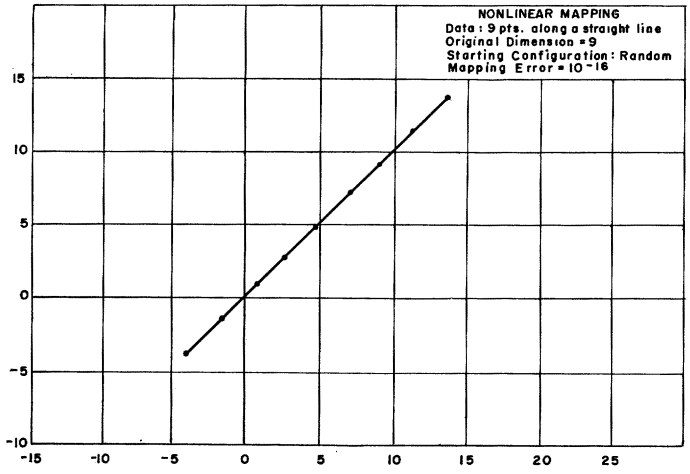

Fig. 1.

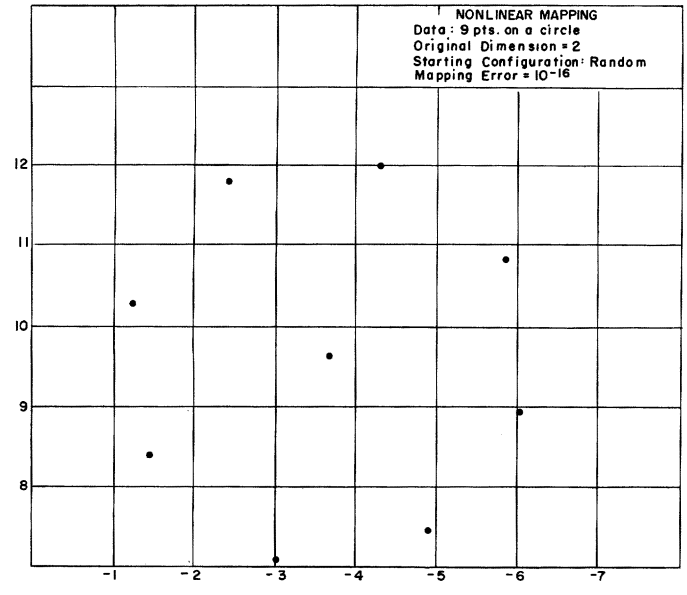

Fig. 2.

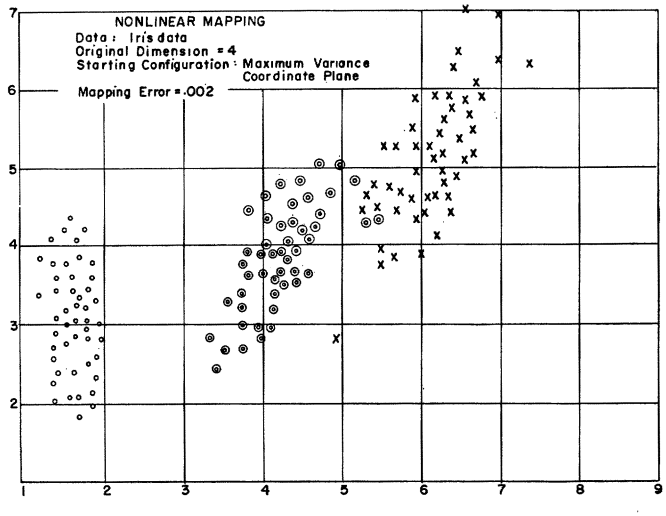

Fig. 3.

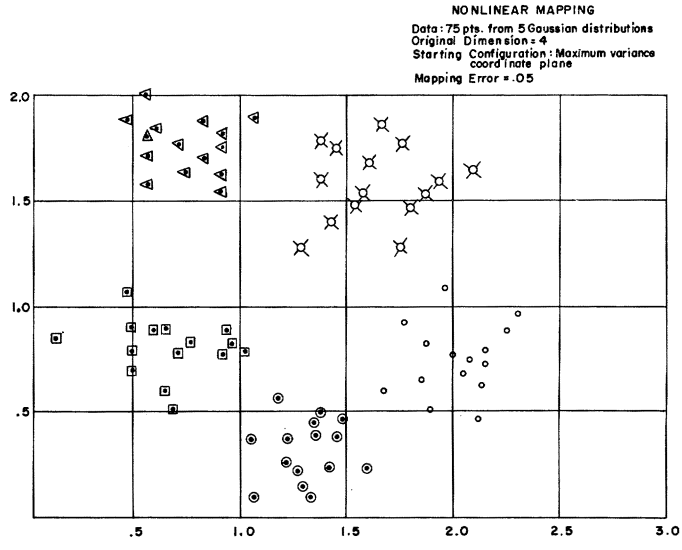

Fig. 4.

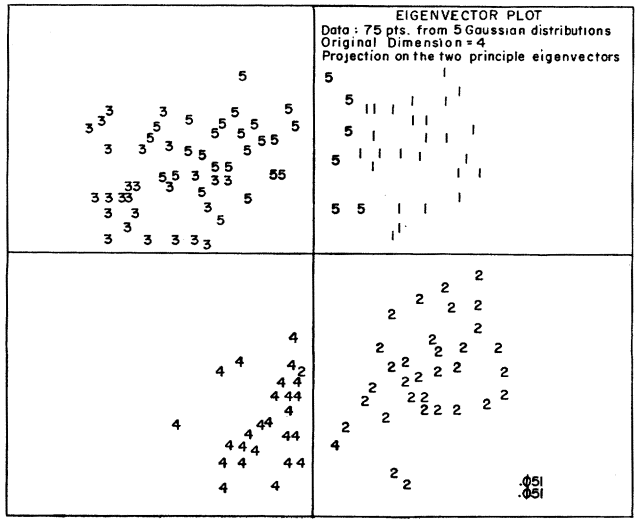

Fig. 5.

Authorized licensed use limited to: New York University. Downloaded on September 03,2022 at 02:10:10 UTC from IEEE Xplore. Restrictions apply.

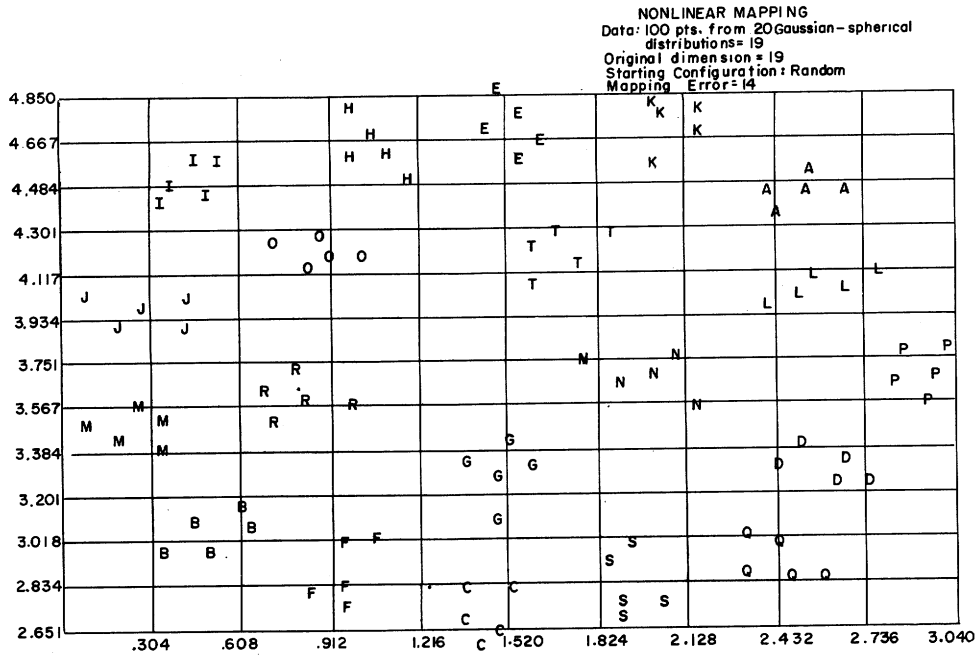

Fig. 6.

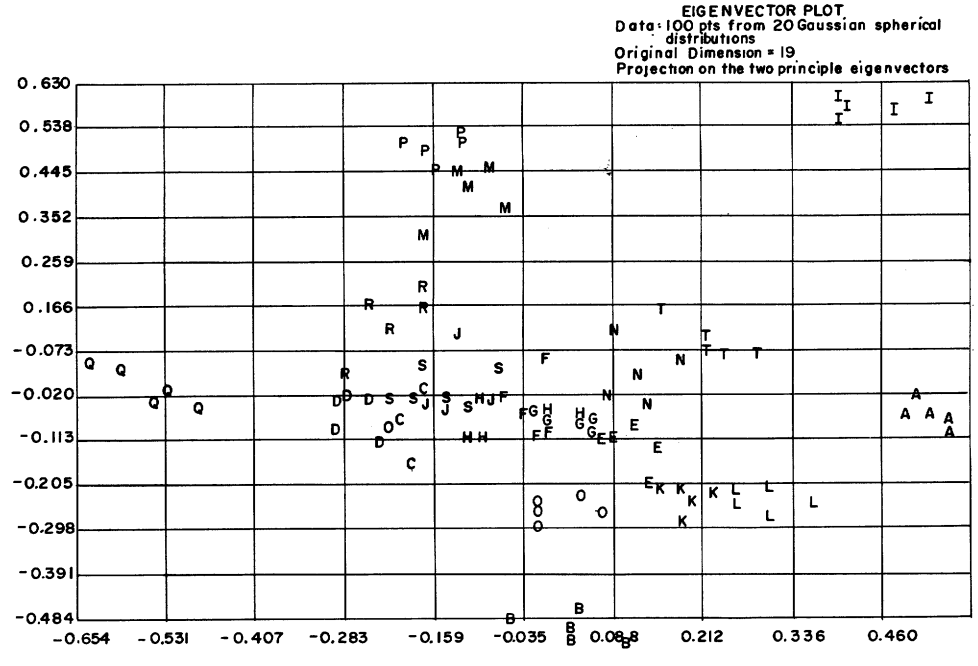

Fig. 7.

Briefly, the C-space construction proceeded as follows. First the subject content covered by the 188 documents in the experimental library was subjectively partitioned into 23 technical fields (see Appendix II for a listing of these fields). Several experts representing each field rated the relevance of each of the 1125 words or phrases to his field, using a scale from 0 to 8. The rating by the experts within each field were then averaged to obtain a word-by-field relevance matrix designated X; the ijth element of X represents the relevance of word or phrase i to field j. (It is convenient to think of the 1125 words or phrases as being represented by vectors in a 23-dimensional space spanned by the 23 coordinate fields.) Next, a 23×23 field correlation matrix C was computed, where the ijth element represented the correlation between the ith and jth fields. C was then factored using the minimum residual method and rotated to a Varimax criterion. Seventeen orthogonal factors were then selected to define the 17-dimensional C-space.

All 1125 words and phrase vectors were mapped into the 17-dimensional C-space using a simple nonlinear formula which tended to emphasize large coordinate

Authorized licensed use limited to: New York University. Downloaded on September 03,2022 at 02:10:10 UTC from IEEE Xplore. Restrictions apply.

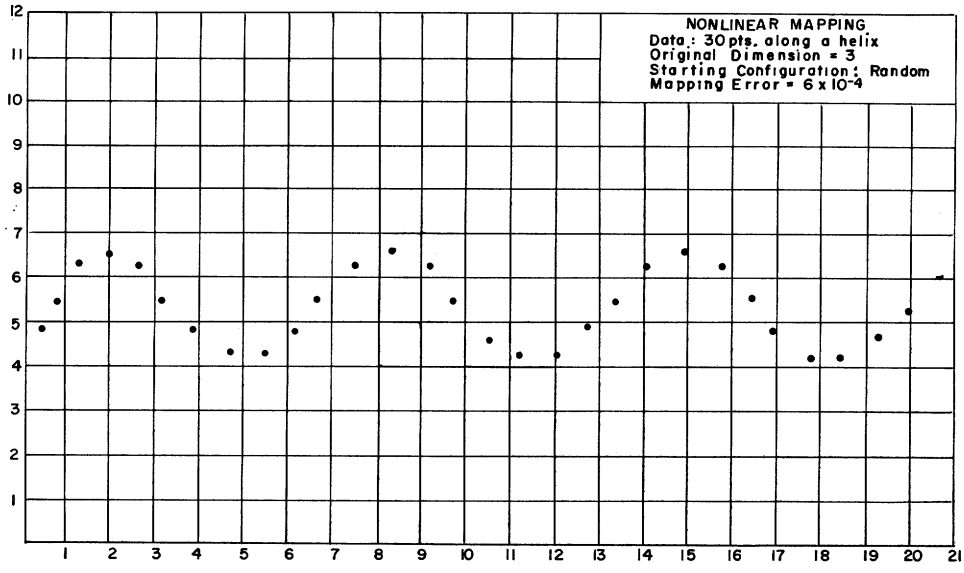

Fig. 8.

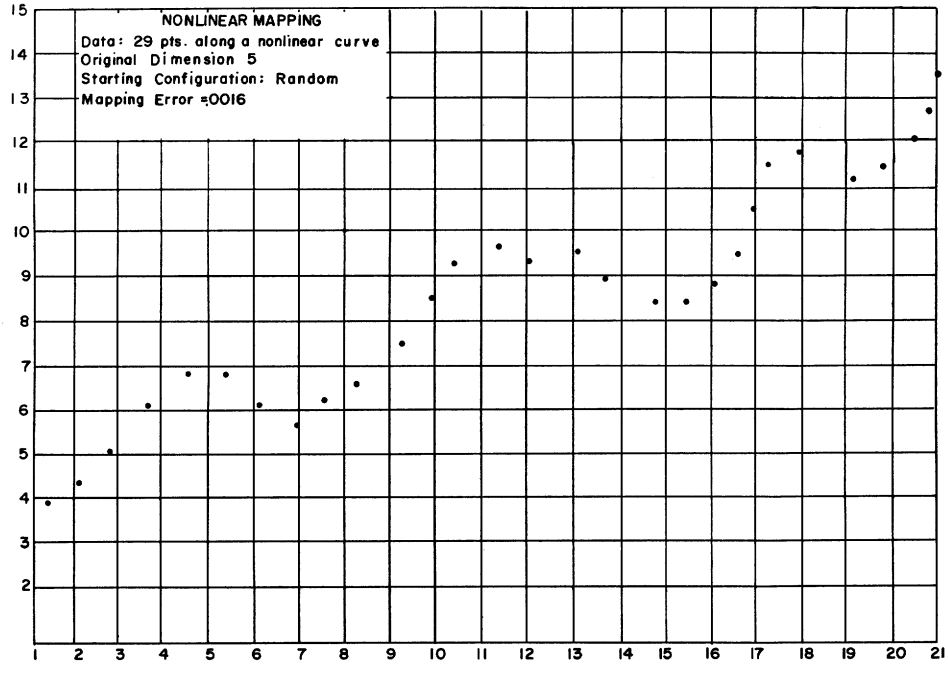

Fig. 9.

projections and minimize small coordinate projections. Finally, the 188 documents were located in the *C*-space by algebraically averaging the word or phrase vectors corresponding to the word or phrases which appeared in the documents.[5]

In order to evaluate the *C*-space as a potential method for document indexing, several individuals were asked to generate English queries (see Appendix III for the pertinent queries used here) which were then keypunched, automatically scanned for key word or phrase content, and finally mapped into the *C*-space. Each requester was then asked to identify those documents of the entire 188 which he felt were most relevant to his query. The *C*-space was then evaluated by examining the rank ordering of the retrieved documents to compare them to the list of relevant documents specified by the requester. The results of this evaluation can be found in Ossorio [8].

The nonlinear mapping algorithm was used to evaluate the "structure" of the documents in the *C*-space. Specifically, we were interested in how the documents considered relevant to a particular request were clustered, and further, how these clusters were interrelated to each other and to the entire library. To accomplish

[5] The entire document was never searched for key words or phrases. Rather, for one half of the documents only the abstracts were used, and for the remainder several paragraphs from each document were used.

Authorized licensed use limited to: New York University. Downloaded on September 03,2022 at 02:10:10 UTC from IEEE Xplore. Restrictions apply.

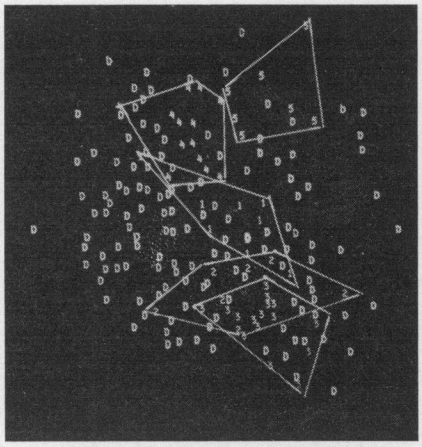

Fig. 10. Nonlinear mapping-photograph of CRT display. Data: 1 = eight Request 1 vectors; 2 = seven Request 2 vectors; 3 = sixteen Request 3 vectors; 4 = thirteen Request 4 vectors; 5 = seven Request 5 vectors. Starting configuration: maximum variance coordinate plane. Mapping error = 0.062.

this analysis, all 188 17-dimensional vectors were used as the input data to the NLM. The numerals 1 through 5 were used in the resulting 2-dimensional mapping to designate the documents labeled relevant to queries 1 through 5. In addition, the symbol D was used to designate the remaining library documents. It is important to note that the NLM algorithm did not utilize the numeric query designations in computing the mapping. Only at the time of plotting the final 2-space configuration of the 188 points were the numeric and symbolic designators used to distinguish the data. The error in achieving the NLM shown in Fig. 10 was 0.062, which was considered to be acceptable for adequate 2-space representation.

The following facts were obtained upon investigation of the NLM result.

1) The documents considered relevant to a given request were clustered, lending evidence to support the hypothesis that related documents have C-space vectors which are close.

2) There does not appear to be any natural C-space structure relating subsets of documents. Instead, the documents tend to be uniformly distributed throughout the space.

3) Clusters 2 and 3 tend to overlap, yet they are well-separated from clusters 4 and 5. This can easily be accounted for since requests 2 and 3 are both concerned with the common subject of statistical data analysis, whereas 4 and 5 involve completely different subjects. In general, the intercluster relationships seem consistent with their respective subject relationships.

In summary, we have found the NLM algorithm to be of considerable value in aiding us in our understanding of the C-space as well as other document spaces. Presently we are planning to incorporate a similar mapping technique in an on-line document retrieval system in order to improve the retrieval via geometric means. The experimental system will operate as follows. The on-line user would examine the 30 highest-ranked documents by retrieving and reading their abstracts. He would then indicate those he considered relevant. Next, a scatter diagram similar to Fig. 10 would be presented upon the CRT display where each of the 30 documents would be indicated by an I or an R, depending upon its relevance. In addition, the original query vector will be displayed as a Q. After examining the relative positions of the documents in the mapping, the user would select (using a light pen) one or more relevant documents to be used to generate a new query vector(s). The concept is that the query vector can be moved to highly relevant regions of the document space by interacting at a display console with a geometric representation of the space.

Relationship of NLM to Other Structure Analysis Algorithms

A mapping algorithm which bears a relationship to the NLM algorithm is one developed by Shepard [11] and later improved by Kruskal [5], [6]. Briefly, the Shepard–Kruskal algorithm seeks to find a configuration of points in a t-space such that the resultant interpoint distances preserve a monotonic relationship to a given set of interelement similarities (or dissimilarities). Specifically, they wish to analyze a set of interelement similarities (or dissimilarities) given by S_{ij}, $i = 1, \cdots, N, j = 1, \cdots, N$. Suppose these similarities are ordered in increasing magnitude, such that

$$S_{p_1q_1} \leq S_{p_2q_2} \leq \cdots \leq S_{p_nq_n}.$$

The Kruskal-Shepard algorithm seeks to find a set of N t-dimensional vectors y_i, $i = 1, \cdots, N$, such that the order of the interpoint distances $d_{ij} = \text{dist}[y_i, y_j]$ deviates as little as possible from the monotonic ordering of the corresponding similarities. Although the mathematical formulations are similar, the underlying mapping criterions are quite different.

Ball [1] has compiled an excellent survey of clustering and clumping algorithms which are useful in solving the "structure analysis" problem. However, it has been our experience in using clustering techniques that these algorithms suffer to some extent from the following four deficiencies.

1) When using a particular algorithm, the resulting cluster configuration is highly dependent upon a set of control parameters which must be fixed by the user. Some examples of such parameters are:

a) the similarity measure;
b) various similarity thresholds;
c) number of iterations required;
d) thresholds which control the increase or reduction of the number of clusters;
e) the minimum number of vectors required to define a cluster.

Authorized licensed use limited to: New York University. Downloaded on September 03,2022 at 02:10:10 UTC from IEEE Xplore. Restrictions apply.

When choosing the control parameters for complex data, the user must either have a good deal of a priori information regarding the "structure" of his data, or he must apply the algorithm many times for different values of the control parameters. This second alternative is, at best, tedious.

2) Most of the existing clustering algorithms are particularly sensitive to hyperspherical structure and are inefficient in detecting more complex relationships in the data.

3) Perhaps the most serious deficiency involving present-day clustering algorithms is that there do not exist really good ways for evaluating a resultant cluster configuration.

4) When two clusters are close, the vectors between tend to form a bridge and cause spurious mergers [7].

We feel that the nonlinear mapping is a highly promising structure analysis algorithm since it suffers little from the listed clustering deficiencies. Consider the following facts concerning the algorithm.

1) The routine does not depend upon any control parameters that would require a priori knowledge about the data. Specifically, the user must set the number of iterations and the convergence constant (MF in Appendix I).

2) It is highly efficient in identifying hyperspherical, hyperellipsoidal, and other complex data structures.

3) The resulting mapping (scatter diagram) is easily evaluated by the researcher, thereby taking advantage of the human ability to detect and identify data structure.

4) The problem concerning extraneous data and spurious mergers is not present since humans easily eliminate troublesome data points by making global evaluations (machines have difficulty performing this function).

5) The algorithm is simple and efficient.

Limitations and Extensions

There are, of course, limitations to every algorithm and the nonlinear mapping is no exception. There exist two limitations which we are presently investigating. The first has to do with the reliability of the scatter diagram in displaying extremely complex high-dimensional structure. It is conceivable that the minimum mapping error is too large ($E \gg 0.1$) and the 2-dimensional scatter plot fails to portray the true structure. However, we feel that for data structures composed of superpositions of hyperspherical and hyperellipsoidal clusters, the nonlinear mapping algorithm will, in general, display adequate representations of the true data "structure."

The second limitation of the nonlinear mapping algorithm is related to the number of vectors that it can handle. Since we must compute and store the interdistance matrix, which consists of $N(N-1)/2$ elements, we are limited at present to $N \leq 250$ vectors.[6] In those cases where $N > 250$, we suggest using a data compression technique to reduce the data set to less than 250 vectors. Specifically, we propose to use the Isodata [2] clustering algorithm to perform data compression. This is actually a natural function of clustering since we replace several vectors with a typical representative vector (i.e., the cluster center). Our previous objections to present-day clustering algorithms do not apply here since we are only concerned with fitting the data with 250 cluster centers. We are specifically not using the clustering algorithm to detect structure.

We have used the NLM to analyze multivariate data from two or more classes for the purpose of determining how well the classes can be discriminated from one another. In these cases, it is recommended that the dimensionality be reduced to the smallest number of variables which still preserve discrimination.[7] In many problems certain measurements provide little discriminatory information; yet if these measurements are included, the NLM will attempt to "fit" interpoint distances along these "noisy" directions as well as along discriminating directions. In truly high-dimensional problems, the resulting mapping may show considerable overlap between classes and still a high degree of discrimination may be possible. This phenomena occurred when analyzing a 4-class, 24-dimensional data set. The resulting NLM (the final error was 0.5, which was considered high) showed considerable overlap among the data from three of the classes; yet, using a piecewise linear discrimination technique (based upon the use of a Fisher's linear discriminant between all pairs of classes), 94 percent correct classification was achieved. In this case, the NLM *did not* give an incorrect result since the

[6] The nonlinear map is programmed in FORTRAN IV and runs on a GE-635 computer equipped with 128 K of core. The computation time can be estimated by

$$T \simeq (1.1 \times 10^{-5}) \frac{I \cdot N(N-1)}{2}$$

minutes, where

I = number of iterations
N = number of vectors.

[7] A number of techniques may be used for this purpose. We often use the following:

a) Discriminant measure

$$M(X) \equiv \sum_{i<j} \frac{(\mu_{xi} - \mu_{xj})^2}{\sigma_{xi}^2 + \sigma_{xj}^2}.$$

b) Interpoint measure

$$M(X) \equiv \frac{1}{\sigma_x^2} \sum_{i<j} \frac{1}{N_i N_j} \sum_{p=1}^{N_i} \sum_{q=1}^{N_j} (X_p^{(i)} - X_q^{(j)})^2$$

where

μ_{x_i} = mean of class i along X
$\sigma_{x_i}^2$ = variance of class i along X
σ_x^2 = variance of all data along X
$X_p^{(i)}$ = the pth sample from the ith class along X
N_i = number of samples from the ith class.

c) Multilinear discriminant defined in Wilks [14].

Authorized licensed use limited to: New York University. Downloaded on September 03,2022 at 02:10:10 UTC from IEEE Xplore. Restrictions apply.

classes greatly overlapped in approximately 20 dimensions and mildly overlapped in the remaining space. The NLM weighted all coordinates equally in an attempt to fit the interpoint distances, and therefore the resulting mapping indicated the predominant overlap which actually existed.

The NLM algorithm described here is one of many algorithms which are being programmed and incorporated into a large on-line graphics-oriented computer system, entitled the On-Line Pattern Analysis and Recognition System (OLPARS) [10].[8] Once the NLM algorithm is incorporated into the OLPARS system, the on-line user will be able to designate a data set, and from the graphics console execute the NLM. The user shall specify a mapping to a 2-space or a 3-space. For $d=2$, the resultant scatter diagram will be displayed upon the CRT; for $d=3$, a perspective scatter plot will be displayed. If the 3-space option is selected, the user will be able to dynamically analyze the resultant perspective scatter diagram by selecting various rotations of the three space. When the user selects $d=2$, he will be given the capability to designate subsets of data (via piecewise linear boundaries drawn on the CRT) representing a collection of points in the scatter diagram which exhibit structure, and thereby partition the initial data list into structured subsets.

Appendix I

Let $E(m)$ be defined as the mapping error after the mth iteration, i.e.,

$$E(m) \equiv \frac{1}{c} \sum_{i<j}^{N} [d_{ij}^* - d_{ij}(m)]^2 / d_{ij}^*$$

where

$$c = \sum_{i<j}^{N} [d_{ij}^*]$$

and

$$d_{ij}(m) = \sqrt{\sum_{k=1}^{d} [y_{ik}(m) - y_{jk}(m)]^2}\,.$$

The new d-space configuration at time $m+1$ is given by

$$y_{pq}(m+1) = y_{pq}(m) - (MF) \cdot \Delta_{pq}(m)$$

where

$$\Delta_{pq}(m) = \frac{\partial E(m)}{\partial y_{pq}(m)} \Big/ \left| \frac{\partial^2 E(m)}{\partial y_{pq}(m)^2} \right|$$

and MF is the "magic factor" which was determined empirically to be $MF \approx 0.3$ or 0.4. The partial derivatives are given by

$$\frac{\partial E}{\partial y_{pq}} = \frac{-2}{c} \sum_{\substack{j=1 \\ j \neq p}}^{N} \left[\frac{d_{pj}^* - d_{pj}}{d_{pj} d_{pj}^*} \right] (y_{pq} - y_{jq})$$

and

$$\frac{\partial^2 E}{\partial y_{pq}^2} = \frac{-2}{c} \sum_{\substack{j=1 \\ j \neq p}}^{N} \frac{1}{d_{pj}^* d_{pj}} \cdot \left[(d_{pj}^* - d_{pj}) - \frac{(y_{pq} - y_{jq})^2}{d_{pj}} \left(1 + \frac{d_{pj}^* - d_{pj}}{d_{pj}} \right) \right].$$

In our program we take precautions to prevent any two points in the d-space from becoming identical. This prevents the partials from "blowing up."

Appendix II

Classification Space Fields

1) Adaptive Systems
2) Analog Computers
3) Applied Mathematics
4) Automata Theory
5) Computer Components and Circuits
6) Computer Memories
7) Computer Softwave
8) Display Consoles
9) Human Factors
10) Information Retrieval
11) Information Theory
12) Input–Output Equipment
13) Language Translation
14) Linear Algebra
15) Multivariate Statistical Analysis
16) Nonnumeric Data Processing
17) Numerical Analysis
18) Pattern Recognition
19) Probability and Statistics
20) Programming Languages
21) Stochastic Processes
22) System Design and Evaluation
23) Time-Sharing Systems.

Appendix III

Requests

Request 1: What is known about the statistical distributions of words or concepts in English text? What impact does this knowledge or lack of knowledge have on the effectiveness of standard statistical methods to information retrieval problems? Are nonparametric methods more applicable?

Request 2: I am interested in techniques for data analysis. In particular, I wish information on "cluster-seeking" techniques as opposed to those of factorial analysis and discriminant analysis. "Cluster-seeking" techniques may be classified as follows: probabilistic techniques,

[8] For other examples of interactive pattern analysis systems, see Ball and Hall [3], Stanley *et al.* [12], and Walters [13].

Authorized licensed use limited to: New York University. Downloaded on September 03,2022 at 02:10:10 UTC from IEEE Xplore. Restrictions apply.

signal detection, clustering techniques, clumping techniques, eigenvalue-type techniques, and minimal mode-seeking techniques.

Request 3: I would like any information concerning Bayesian statistics. In particular, I would like to know if one can define or devise multiple-decision procedures from the Bayes approach. Also, how sensitive are Bayes procedures to the prior distribution? Finally, I would like a comparison of the Bayes approach to other classical decision theoretic approaches.

Request 4: What is the structure and characteristics of paging techniques?

Request 5: Are there survey documents (information) available which discuss or detail the relative practicality of memories; for example, capacity versus utilization, density, weight, environmental features, failure rates, economics, etc.?

Acknowledgment

The author expresses his appreciation to D. Elefante for his efforts in developing an efficient FORTRAN IV program for the Nonlinear Mapping Algorithm.

References

[1] G. H. Ball, "A comparison of some cluster-seeking techniques," Rome Air Development Center, Rome, N. Y., Tech. Rept. RADC-TR-66-514, November 1966.
[2] G. H. Ball and D. Hall, "Isodata," portion of Stanford Research Institute (SRI) Final Report to RADC Contract AF30(602)-4196, September, 1967.
[3] ——, "Promenade—an improved interactive graphics man/machine system for pattern recognition," Stanford Research Institute, Menlo Park, Calif., Project 6737, October, 1968.
[4] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Ann. Eugenics*, vol. 7, pp. 178–188, 1936.
[5] J. B. Kruskal, "Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis," *Psychometrika*, pp. 1–27, March 1964.
[6] ——, "Nonmetric multidimensional scaling: a numerical method," *Psychometrika*, vol. 29, pp. 115–129, June 1964.
[7] G. Nagy, "State of the art in pattern recognition," *Proc. IEEE*, vol. 56, pp. 836–861, May 1968.
[8] P. G. Ossorio, "Classification space analysis," RADC-TDR-64-287, October 1964.
[9] ——, "Attribute space development and evaluation," RADC-TDR-67-640, January 1968.
[10] J. W. Sammon, "On-line pattern analysis and recognition system (OLPARS)," RADC-TR-68-263, August 1968.
[11] R. N. Shepard, "The analysis of proximities: multidimensional scaling with an unknown distance function," *Psychometrika*, vol. 27, pp. 125–139, 219–246, 1962.
[12] G. L. Stanley, G. G. Lendaris, and W. C. Nienow, "Pattern recognition program," AC Electronics Defense Research Labs., Santa Barbara, Calif., TR-567-16, November 1967.
[13] C. M. Walters, "On line computer based aids for the investigation of sensor data compression, transmission and delay problems," *1966 Proc. Natl. Telemetry Conf.*, Boston, Mass.
[14] S. S. Wilks, *Mathematical Statistics*. New York: J. Wiley, 1962.

Mathematical Analysis of Ferrite Core Memory Arrays

WILLIAM T. WEEKS

Abstract—**A mathematical model for simulating pulse propagation in ferrite core memory arrays is described. Although specifically developed to analyze 3-dimensional arrays, the model is sufficiently general to give a satisfactory analysis of pulse propagation, waveform deterioration, and noise generation in a wide variety of memory configurations. The model treats the memory as a generalized, mutually coupled, multiconductor transmission line system. Insofar as is possible, the transmission line parameters are calculated from the array geometry, thus leaving only a small number of parameters that must be supplied empirically. Following a discussion of the equations which define the model and the methods by which they are solved, a sample array calculation is given to illustrate the kind of information that can be obtained from the model.**

Index Terms—**Arrays, computers, ferrite cores, memories, pulse propagation, transmission line system.**

Manuscript received October 23, 1968; revised February 10, 1969.
The author is with IBM Corporation, Components Division, Poughkeepsie, N. Y. 12602.

Introduction

During the past five years, considerable progress has been made in the development of mathematical models for simulating the electrical properties of ferrite core memory arrays. The purpose of this paper is to describe the techniques available for the analysis of 3-dimensional ferrite core memory arrays. The techniques presented here represent a substantial advance in the state-of-the-art over earlier reported work [1], [2], which dealt mainly with the simulation of 2-dimensional arrays.

A precise mathematical description of a memory array, backed up by a rigorous and practically realizable method for solving the resulting equations, would be of inestimable value to a memory designer, for it would remove much of the uncertainty from the design process.

Authorized licensed use limited to: New York University. Downloaded on September 03,2022 at 02:10:10 UTC from IEEE Xplore. Restrictions apply.