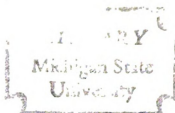


DEVELOPMENT OF A BASIS FOR
GOAL ORIENTED CLUSTERING

Dissertation for the Degree of Ph.D.
MICHIGAN STATE UNIVERSITY
ALI BEHFOROOZ
1975



This is to certify that the
thesis entitled
Development of a Basis for
Goal Oriented Clustering

presented by

Ali Behforooz

has been accepted towards fulfillment
of the requirements for

Ph.D degree in Computer Science

A handwritten signature in blue ink that reads "M.A. Rahimi".

Major professor

Date October 22, 1975

695596

ABSTRACT
DEVELOPMENT OF A BASIS FOR
GOAL ORIENTED CLUSTERING

By

Ali Behforooz

This thesis is concerned with the first two of the following group of problems which were faced during the design of a Television-Computer Education System: 1) representation of a nominal-scale variable with unknown nominal value by numeric data for cluster analysis, 2) development of a method of clustering for a set of data items defined by such nominal-scale variables, 3) assignments of an appropriate sequence of actions to every generated cluster, 4) measurement of the effectiveness of the assigned actions, and 5) control or prediction of changes in the clusters and the members of the clusters. The basic structure of a clustering system which is designed so as to provide solutions to these problems is developed. It is called a Goal Oriented Clustering System, because every step of this system is dependent on the purpose(s) of the clustering.

In this thesis a complete solution is provided for the first problem. The nominal-scale variables are represented by a vector of dimension M , where M is the number of categories of the variable. If the nominal value of a nominal variable V is known then the vector representing this variable is a deterministic vector. If the nominal value of V is unknown then the vector representing this variable is a probabilistic vector.

Also, in this thesis a complete solution is provided for the clustering problem. To cluster a set of data items defined by nominal-scale variables with unknown nominal value, a data item is represented by an $n \times M$ matrix, where n is the number of variables. An M -dimensional distance function is proposed to calculate the association measure between data items. Two different clustering methods are developed which are capable of clustering data items represented by $n \times M$ matrices as well as data items represented by n -dimensional vectors.

The last three problems mentioned at the beginning are left for future research by others.

DEVELCPMENT OF A BASIS FOR
GOAL ORIENTED CLUSTERING

by

Ali Behforooz

A THESIS

Submitted to

Michigan State University
Department of Computer Science

in partial fulfillment of the
requirements for the degree of

DOCTOR OF PHILOSOPHY

1975

TO

FARIDEH & AMIR

ACKNOWLEDGMENT

I am grateful to Dr. Morteza A. Rahimi, my major professor, for his encouragement during the course of this research. My thanks also go to Dr. Harry G. Hedges, Dr. Richard J. Reid, Dr. John J. Forsyth, and Dr. Dennis D. Gilliland for their serving on my guidance committee and their critical review of this thesis.

My special thanks go to National Educational Television of Iran for their financial support of this work. Finally, my thanks go to Dr. Martin O. Holien, Chairman of the Department of Computer Science at Moorhead State University, Moorhead, Minnesota for his support.

LIST OF SYMBOLS

$\emptyset$	Empty set, a set with no element.
$ S $	If S is a set, $ S $ is the number of elements of the set.
$\underline{a}$	The cardinality of the set of natural numbers.
$\underline{c}$	The cardinality of the set of real numbers.
$\underline{r}$	A vector with all components equal to r .
I^+	The set of positive integers.
I^-	The set of negative integers.
R_e	The set of real numbers.
R^+	The set of non-negative real numbers.
$p \vee q$	p or q .
$p \wedge q$	p and q .
xRy	x is related to y by relation R .
$x \not R y$	x is not related to y by relation R .
$R[A]$	The set of R -related elements of A .
$[a]_R$	The set of all elements related to a by relation R .
iff	If and only if.
$a \implies b$	a implies b .
$a \iff b$	a if and only if b .
$\forall a$	For all a .
$\exists a$	There exists an a .
$\nexists a$	There does not exist an a .
$x \in A$	x is an element of A .
$x \notin A$	x is not an element of A .
$\supset$	Such that.

LIST OF SYMBOLS (continue)

$A \subset B$	A is a subset of B.
$\{a, b, \dots\}$	A set with elements a, b, ..etc.
$(a, b, \dots)$	A vector with components a, b, ..etc.
$[d_{ij}]$	A matrix with d_{ij} as its j^{th} element of its i^{th} row.
$[d^i_{jk}]$	A three dimensional matrix.

TABLE OF CONTENTS

Chapter	Page
I	INTRODUCTION.....
1.1	Five Major Problems..... 1
1.2	A Brief Review of Current Clustering Methods.... 5
1.3	Scope of This Research..... 10
1.4	Organization of This Thesis..... 11
II	GOAL ORIENTED CLUSTERING.....
2.1	Introduction..... 13
2.2	Goal Oriented Clustering System (GOCS)..... 14
2.3	Elements of the GOCS..... 16
III	NUMERIC REPRESENTATION OF NON-NUMERIC VARIABLES(NNV's) FOR CLUSTER ANALYSIS.....
3.1	Introduction..... 21
3.2	Category Relation..... 22
3.3	Representation of an NNV. 26
3.4	Representation of a Data Item..... 27
3.5	Representation of a Variable..... 31
3.6	Some Motivation for the Proposed Representation. 36
IV	ASSOCIATION MEASURES.....
4.1	Introduction..... 41
4.2	Association Measure Between Variables..... 42
4.2.1	Association Measure Between Numeric Variables. 43
4.2.2	Association Measure Between NNV's..... 44
4.3	Association Measure Between Data Items..... 48

TABLE OF CONTENTS

(continue)

Chapter		Page
V	CLUSTERING PROCEDURES.....	
	5.1 Introduction.....	54
	5.2 Geometric Representation of D and Δ	55
	5.2.1 Hypersphere Representation of Data Items....	55
	5.3 Clustering the set D by Transforming Δ to an N x N dissimilarity matrix.....	58
	5.4 General Centroid Method(GCM).....	62
	5.4.1 General Centroid Algorithm(GCA).....	64
	5.4.2 Analysis of the GCA.....	68
	5.4.3 Computational Limits and the Problem of Storage.....	70
	5.4.4 Some Special Cases.....	71
	5.5 Graph Representation of D and Δ	74
	5.6 k-clustering Based on Graph Representation of D and Δ	78
	5.6.1 k-clustering Algorithm.....	86
VI	SUMMARY AND CONCLUSION.....	
	6.1 Summary of this Research.....	92
	6.2 Conclusion.....	94
	6.3 Suggested Topics for Further Work	95
	BIBLIOGRAPHY.....	97

LIST OF FIGURES

Figure		Page
1.1	Dynamic Property of Clusters.....	4
1.2	Visual and Algorithmic Clusters.....	8
1.3	Visual and Algorithmic Clusters.....	8
1.4	Visual and Algorithmic Clusters.....-.....	8
2.1	Goal Oriented Clustering System.....	15
4.1	Unit Spheres of M_1 , M_2 and M_∞	50
5.1	Geometric Representation of D and Δ	56
5.2	Hypersphere Representation of the Data Items of Example 4.2.....	57
5.3	Bipartite, k -partite, Uniform k -partite, Uniform Weighted k -partite and Average Graph.....	76-77
5.4	Graph Representation of D and Δ of Example 4.2..	77
5.5	A Picture of Definition 5.9 for Weighted Graphs.	80

CHAPTER I

INTRODUCTION

Most nominal-scale data , especially those which are to define behavioral characteristics , are such that their nominal value is unknown. Throughout this entire thesis, nominal-scale data which have uncertain nominal values are referred to as Non-Numeric Variables(NNV's). This terminology makes it possible to classify variables into two categories. The first category contains all variables which can be represented by a single number regardless of whether the scale is ratio, interval, ordinal, or nominal. Variables of this class are called numeric variables. The second category contains those nominal-scale variables which have uncertain nominal values and as stated above are called NNV's.

The problem under consideration is to extend some of the existing clustering procedures so as to be able to cluster a set of N data items each of which is defined by n variables some of which are NNV's.

The major topics covered in this thesis are 1) numeric representation of a NNV, 2) association measures between NNV's and also between data items defined by NNV's and 3) clustering a dissimilarity N -block matrix of a set of N data items defined by both NNV's and numeric variables.

1.1 Five Major Problems:

The motivation for this research came from the design of a Television-Computer Educational (TCE) system. A major goal of the TCE system is to be able to act as an instructor and, at the

same time, be relatively inexpensive, efficient, and fast enough to handle a large number of students. During the design study of the TCE system, five major problems as given below were faced. The first two problems are investigated and totally solved in this research and other people are working on the last three problems.

1. The Problem of Representing Data Items: The data items of the TCE system are the students and each student is represented by a set of variables. Some of these variables are NNV's and they are important to the student clustering procedure. This type of variable must be represented numerically for clustering purposes. This problem is included in the scope of this research and is completed in chapter 3.

2. The Problem of Clustering: The number of students enrolled for a given course is large, the facilities to serve these students are limited and the students are from different backgrounds. Because of these reasons and the fact that the TCE system is to be efficient, it is necessary to divide the students into different groups. A clustering algorithm that performs this task is essential to the operation of the TCE system. This problem is also included in the scope of this research and is discussed in Chapters 4 and 5.

3. The Problem of Assigning Appropriate Actions to the Generated Clusters:

The TCE system should be designed such that it gives proper assignments to the students. The action sequence or sequence of assignments, tests, and homework corresponding to each group of students should be determined on the basis of the "label" of that group and modified according to certain

characteristics of the individual members of the group. In other words, the action sequence for a group cannot be pre-defined. Let C_i be the label of the i^{th} group and $A_i = A_{i1}, A_{i2}, \dots, A_{in_i}$ be the action sequence determined for this group. The action sequence A_i is called a pre-defined action sequence if it is not a function of the members of the i^{th} group.

4. The Problem of Applicability of A_i : Once the students are clustered and all action sequences are determined, it is time to determine whether or not the action sequence A_i is the best action sequence for the group C_i . The action sequence A_i is accepted as the best action sequence for C_i if it possesses the following properties:

- a. there are no conflicts in A_i ;
- b. all actions included in A_i are applicable to every individual member of C_i ;
- c. the application of A_i to the members of C_i would provide the expected changes for the group C_i as well as the individual members of C_i .

An algorithm is needed to test those three properties of every generated action sequence.

5. The Problem of Dynamic Clusters: The members of group C_i might move their positions from one point to another under the application of the action sequence A_i . This change of position is due to the changes in values of variables defining the students. This movement does not have to be in the same direction for all members of C_i . It is true that all members of C_i are under the effect of the same action sequence, but this does not mean that the progress or improvement of the

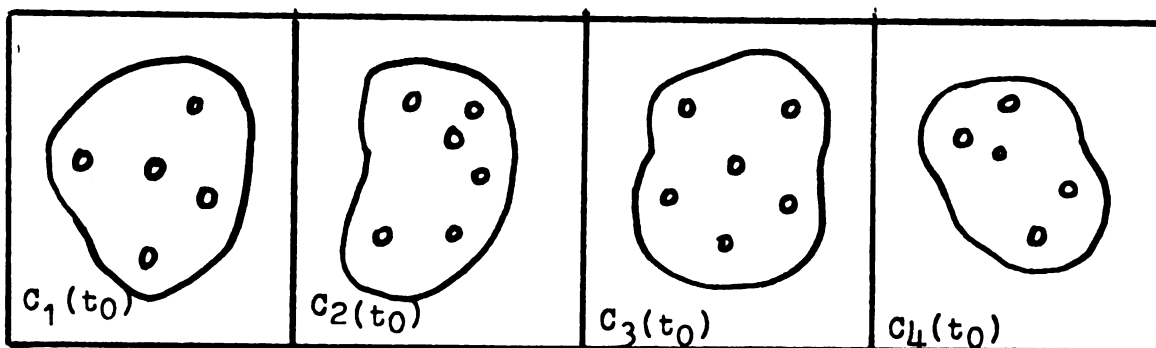
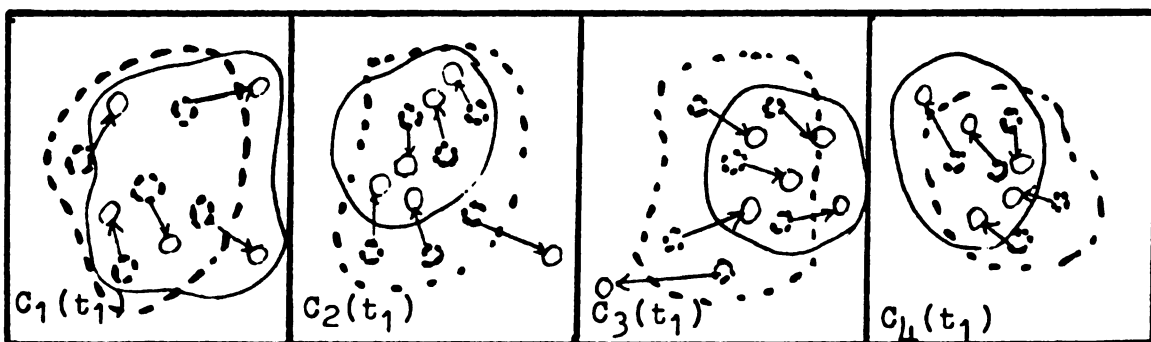
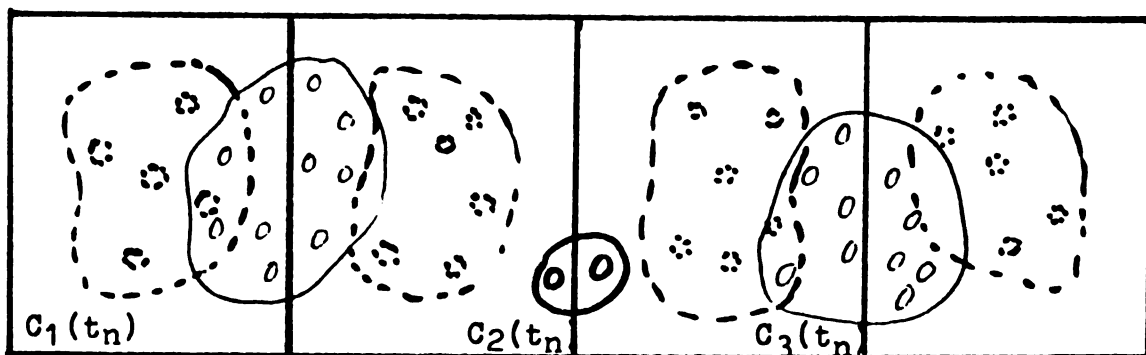
$t = t_0$

 $t = t_1$

 $t = t_n$


Figure 1-1.

Dynamic property of Clusters.

individuals will be the same. Figure 1-1 shows a set of four clusters at times $t_0, t_1, \dots, t_n$. The solid clusters are the positions of the dotted clusters at time $t = t_i$, $i=1, 2, \dots, n$. The movement of the elements of the clusters is called the dynamic property of the clusters. The dynamic property of a cluster could be studied under the following two separate cases.

1. The members of a cluster move such that they stay in the same cluster. Clusters $C_1(t_1)$ and $C_4(t_1)$ of Figure 1-1 are examples of this type of movement.

2. The members of a cluster move such that they explode the cluster; that is, at least one of the members does not remain a member of the same cluster. Clusters $C_2(t_1)$ and $C_3(t_1)$ of Figure 1-1 are examples of exploded clusters. In Figure 1-1 $C_1(t_0), \dots, C_4(t_0)$ are four initial clusters and $C_1(t_1), \dots, C_4(t_1)$ are four clusters after a period $t=t_1-t_0$. The application of the action sequence A_i to the cluster $C_i(t_0)$ causes a change in values of the variables defining the members of this cluster. The change in position of a member of a cluster is due to the change of variables defining this member. The enlarged clusters are special case of exploded clusters.

A system called Goal Oriented Clustering is designed in Chapter Two such that it is possible to solve all five major problems mentioned previously.

1.2 A Brief Review of Current Clustering Methods:

In this section a very brief description of some cluster analysis approaches is given. For those who are interested in a complete description, without referring to the original works, Anderberg [2] Chapters Six and Seven is a good

reference. Ling [41] gives a description of some of the current work along with some criticisms. Most of the current methods in clustering are included in one of the following categories:

1. Agglomerative Algorithms: Most of the widely used clustering algorithms belong to this family of algorithms. There are a large number of papers in hierarchical and non-hierarchical clustering employing agglomerative algorithms. For a complete description and references to the work in this area the reader is referred to Anderberg [2]. Very brief descriptions of some of the widely used agglomerative algorithms are given below.

a. The Single Linkage Algorithm: This algorithm is also known as the "nearest neighbor" or the "minimum" algorithm. It has been considered by Sneath [65], Sokal and Sneath [66], Johanson [35], Shepard and Willmott [64], Zahn [79] and others. This algorithm combines the two clusters whose distance is the minimum among all pairs of clusters. The distance between two clusters X_i and Y_j is given by

$$d(X_i, Y_j) = \text{Minimum}_{\substack{x \in X_i \\ y \in Y_j}} (d(x, y)) .$$

The merging process continues until a terminating criterion is satisfied or until all points belong to a single cluster.

b. The Complete Linkage Algorithm: This algorithm has been studied by Sokal and Sneath [66], Johnson [35] and others. It differs from the minimum algorithm only in the definition of the distance between two clusters. If X_i and Y_j are two clusters the distance between them is defined by

$$d(X_i, Y_j) = \text{Maximum}_{\substack{x \in X_i \\ y \in Y_j}} (d(x, y)).$$

c. The Centroid Algorithm: In this method the distance between two clusters is defined by $d(X_i, Y_j) = d(\bar{X}, \bar{Y})$, where X_i and Y_j are two clusters and $\bar{X}$, $\bar{Y}$ are the mean of these two clusters. The distance function d is the n -dimensional Euclidian distance function.

d. Minimizing Within-Group Sum of Square (WGSS) Algorithm: This method was introduced by Fisher [16] and applied by Ward [77]. It is also known as Ward's method. Ward's method is employed by Jensen [36], Vinod [75], Dubes [10], Fisher and Van Ness [15], Fukunaga and Koontz [19] and others. The WGSS method is applicable only to some configuration similar to Figure 1-2(a), where the clusters are compact and/or hyperellipsoidal and in either case they are well separated. In some cases, as in Figure 1-2(b) and (c), this method will not give an appropriate partition. Assuming the data items are in two-dimensional space, the optimal clusters of Figure 1-2 are the ones given in Figure 1-3. But by using the WGSS method the resultant cluster for Figure 1-2(a) is the same as in Figure 1-3(a); and the resultant clusters for Figure 1-2(b) and (c) are those given in Figure 1-4(b) and (c), respectively.

2. Divisive Algorithms: There exist a variety of techniques based on dividing the entire set of data items into subgroups. These methods are focused on finding the groups which are the most widely separated from each other. Divisive algorithms can be categorized into three different approaches.

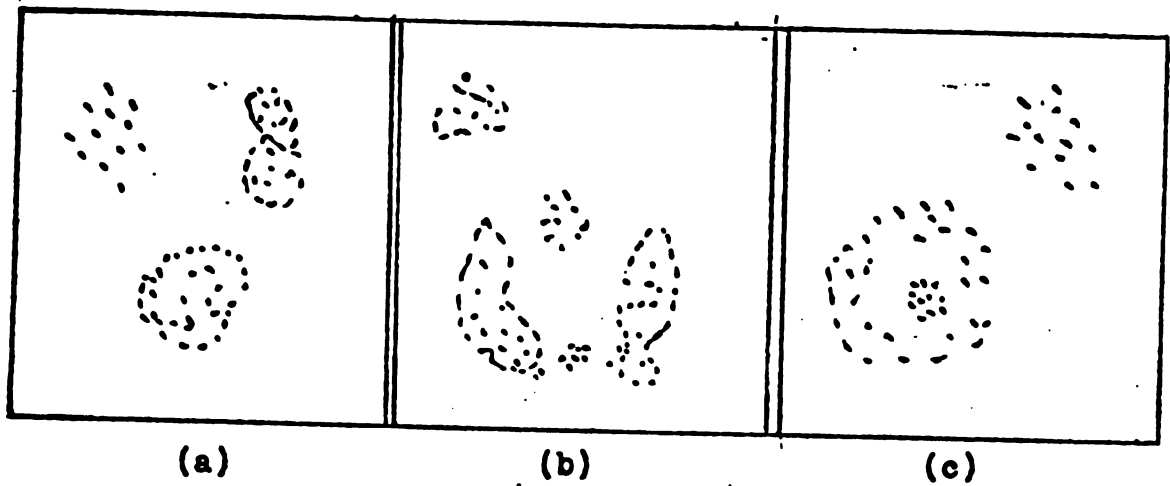


Figure 1-2.

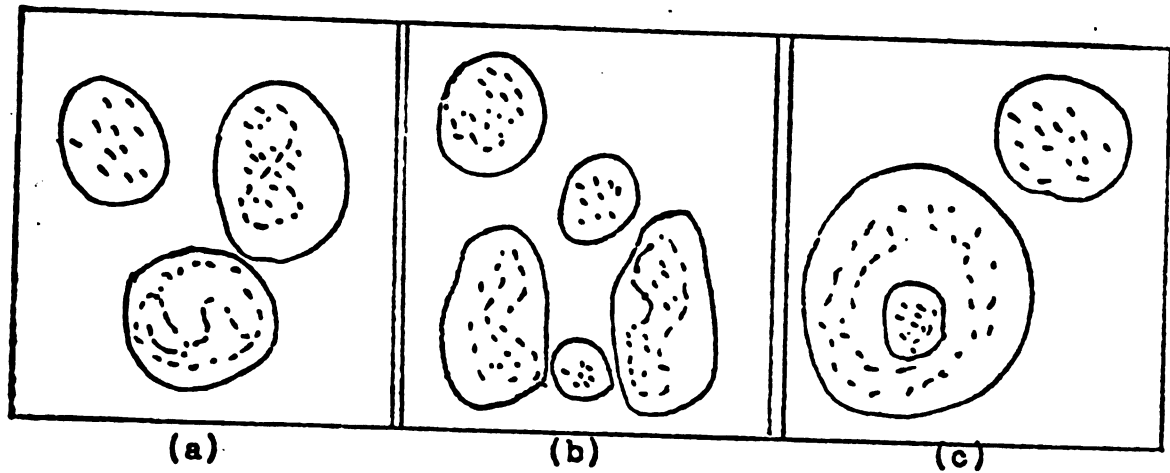


Figure 1-3.

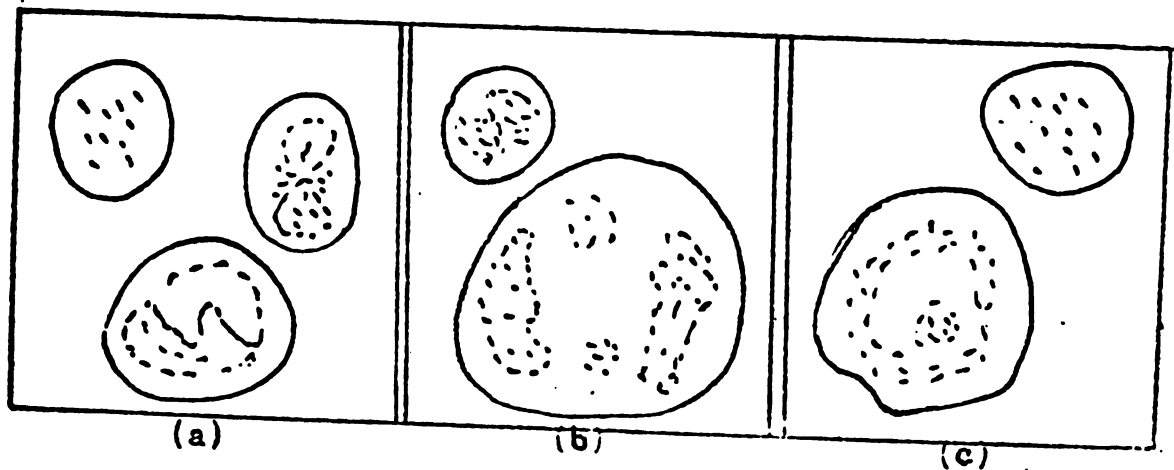


Figure 1-4.

Visual and Algorithmic Clusters.

The first approach is under the label "association analysis" and is discussed by Lance and Williams [43]. The second approach is proposed by Edwards and Cavalli-Sforza [12]. Scott and Symons [61] and Harding [28] have some work related to this approach. The third approach is based on the use of discriminant analysis. The idea is to start with some initial partition, compute a linear discriminant function and then iteratively reassign points and recompute discriminant functions so as to find the most strongly separated groups. Dubes [10] provides a computer program for this method.

Since the k -clustering method, introduced by Ling[41], is one of the two methods which are extended in Chapter Five of this thesis, some more words about this method are appropriate. In the k -clustering method a cluster is a well-defined entity which depends on the dissimilarity matrix and a parameter k . Given a set of data items D , a dissimilarity rank matrix R , and a positive integer k , the k -clustering algorithm will deliver all subsets of D that are k -clustered according to the definition of k -cluster. In this method the problem is not to define a cluster but the problem is to find the clusters. As soon as the dissimilarity matrix is determined no information about the individual data items is necessary and the dissimilarity matrix should be computed only once.

1.3 Scope of This Research:

Numeric variables are those which can be represented by a single number. For example, AGE and INCOME are two numeric variables. Non-numeric variables are those which cannot be represented by a single number. For example, BEHAVIOR and PERFORMANCE are two non-numeric variables. Every clustering procedure attempts to categorize data items defined by a set of n variables. Each data item consists of a vector of n dimensions (if all variables are numeric). Each component of this numeric vector is the value of one of the n variables for the given data item.

The first goal of this research was to find a meaningful and/or comparable numeric representation for NNV's. The approach used is explained in Chapter Three of this thesis.

The second goal of this research was to find an appropriate distance function between data items defined by both numeric variables and NNV's. The approach used is delineated in Chapter Four of this thesis.

The third goal of this research was to develop procedures capable of clustering data items defined by both numeric variables and NNV's. The form of the distance matrix for such data items is different from that for data items defined only by numeric variables so a new procedure was needed to cluster such data items. This problem has been approached in two different ways as explained in Chapter Five of this thesis.

The fourth goal of this research was to approach the design of a general system which provides the solutions to the five major problems given in section 1.1. The goal oriented

clustering system is a procedure which is designed in a manner which may fully solve the above five major problems. The goal oriented clustering system is defined in Chapter Two of this thesis.

1.4 Organization of This Thesis:

Chapter one has been an overview and description of a number of major problems to be either solved in this research or considered for further research.

Chapter two is a design study of a procedure called the goal oriented clustering system(GOCS). The description of the elements of the GOCS is given in this chapter.

Chapter three provides a vector representation of NNV's. A relation R_i called a category relation, is defined and a partition is introduced on the range of every NNV X . Each NNV X is represented by an M -dimensional row vector and each data item D_i is represented by an $n \times M$ matrix.

Chapter four provides a number of appropriate distance functions to measure the dissimilarity (or similarity) between data items as well as between variables. An M -dimensional distance function is defined and some of the properties of this distance function are studied. The Minkowski metric is extended to an M -dimensional metric and is used to measure the dissimilarity between data items.

Chapter five provides a number of different procedures for clustering data items defined by NNV's. Two different approaches are provided for clustering a set of data items D . The first approach is to change the N -block dissimilarity matrix $\underline{D}$ to an $N \times N$ square matrix and then use one of the

current clustering methods to cluster the set D . The second approach is to develop a new clustering procedure to cluster the data items based on the N -block dissimilarity matrix Δ . A method similar to the centroid method is developed called the "General Centroid Method." Ling's [41] method is extended by using the graph representation of D and Δ . The theory for this extension is also provided in Chapter 5.

Chapter six is a summary of this research. A number of important topics for further research are given.

CHAPTER II

GOAL ORIENTED CLUSTERING

2.1 Introduction:

This chapter is a study of a new clustering system designed so as to be able to provide solutions for the five major problems given in chapter one. This system is fully described but only the clustering block of the system is completely developed. This system is called a Goal Oriented Clustering System (GOCS). The clustering block of the GOCS consist of three different procedures; 1) representation of data items defined by NNV's , 2) association measures between data items as well as between variables, and 3) a procedure to cluster an N-block dissimilarity matrix. These three procedures are discussed in Chapters 3, 4 and 5, respectively.

Through discussion with people from different areas of application, it was discovered that the five major problems mentioned in section 1.1 are not unique to the TCE system. People involved with research in areas such as behavioral science, psychology, sociology, biometry, biology, and education have problems very similar to those of the TCE system. The sociologists are vitally concerned about the overall behavior of every cluster after each period of time. They would like to see the relations between clusters as well as between the members of the cluster before and after application of the action sequences to the clusters. The psychologists would like to be able to represent their objects by NNV's

as well as numeric ones. The sociologists want to know the direction of any movement of their clusters. They use this information to predict some of the future events of society. The biologists want to know the interaction between clusters. They want to have full control over the dynamic property of the clusters.

2.2 Goal Oriented Clustering System (GOCS):

The GOCS consists of the three separate blocks given below.

1. Clustering Block: This block clusters the set of data items defined by a set of n variables some of which are NNV's. The input to the clustering block is a set of N data items each one of which is defined by an $n \times M$ matrix. The output of this block is a set of m clusters $C_1, C_2, \dots, C_m$.

2. Action block: This block assigns an action sequence to every generated cluster. These action sequences are based on the set of actions, the set of needs, the members of each cluster, and the cluster label. The input to the action block is a set of m clusters and the output is a set of m pairs $\langle C_i, A_i \rangle$, where C_i is a cluster and A_i is the action sequence corresponding to C_i .

3. Acceptor Block: This block tests the applicability of the action sequence A_i to the cluster C_i . Each A_i should be checked for the three general properties mentioned in section 1.1. The input to this block is a set of pairs $\langle C_i, A_i \rangle$ and the output is either "accepted" or "rejected."

GOAL ORIENTED CLUSTERING SYSTEM

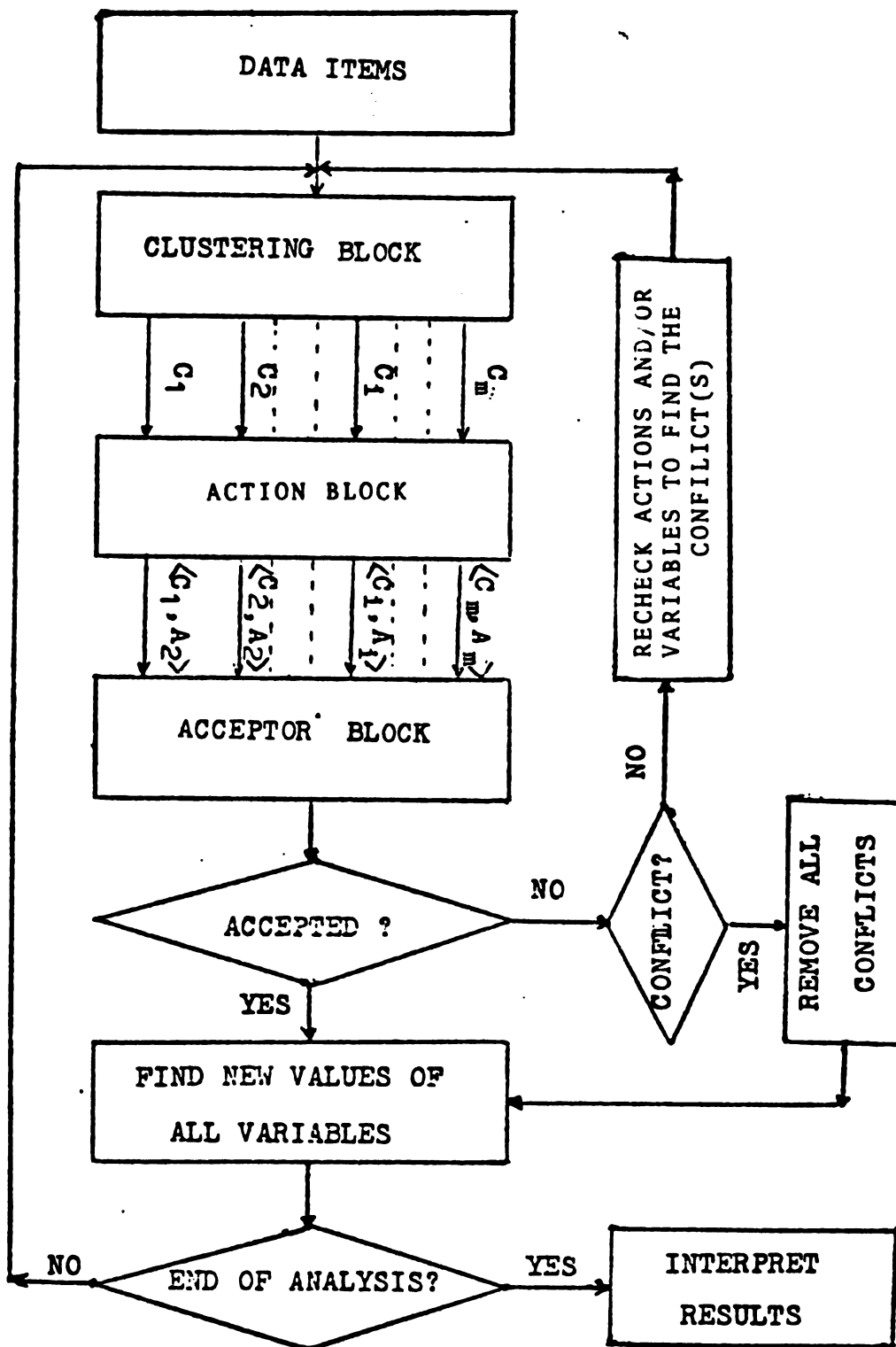


Figure 2-1.

The process of the GOCS is not limited to finding the most similar objects. It is to assign a set of actions to a set of data items in order to achieve a set of pre-defined goals. Figure 2-1 is a simple diagram of the GOCS. In order to take care of the dynamic property of a cluster, the GOCS is set up such that it could run periodically. The GOCS halts either by user request or when convergence occurs. The GOCS is said to converge if it generates only one pair $\langle C_1, A_1 \rangle$; that is, all data items are clustered into one group.

2.3 Elements of The GOCS:

The following is a list of elements needed to be defined and/or introduced for the GOCS before using it.

1. Data Item: Every member of a finite sample space $\mathcal{X}$ is called a data item. For example, let $\mathcal{X}$ be the set of all graduate students at Michigan State University. Everyone enrolled as a graduate student at MSU is a data item. To define the set of data items, it is sufficient to define $\mathcal{X}$. Every member of $\mathcal{X}$ is a data item. The set $D = \{D_1, D_2, \dots, D_N\}$ is called a set of N data items and is a subset of $\mathcal{X}$.

2. Variable: A data item D_i could be defined by a set of n variables and represented by a characteristic vector or by an $n \times M$ matrix (if a data item is defined by some NNV's, one might represent it by an $n \times M$ matrix as given in Chapter Three.) The components of this characteristic vector (or the row of the characteristic matrix) are the values of the n variables for a data item D_i . The variables which represent data items for the goal oriented clustering system are totally a function of the purposes of the clustering.

One might represent a graduate student of MSU by

$$D_i = (\text{Department, B.S. Major, Previous Institution})$$

for a purpose P_1 . For another purpose, say P_2 , one could represent the same student by

$$D_i = (\text{Department, GPA, Term Credits}).$$

Variables are the means to measure the distinction and the differences between data items.

3. What to Cluster: The set $D = \{D_1, D_2, \dots, D_N\}$ and the variables $V_1, V_2, \dots, V_n$ are chosen. Does one need to put "similar" data items in one group (data item clustering), does one want to see the "similarity" between variables (variable clustering) or does one want the simultaneous clustering of data items and variables? Answers to these questions must be decided in order to know what is to be clustered.

4. Variable Transformation: Every clustering procedure deals with a measure of association between variables. If the variables are not of the same scale of measurement it is necessary to homogenize the variables to have a unique scale of measurement.

5. Concept of Similarity: One of the well known definitions of clustering procedures says that a clustering procedure must generate an optimal partition on the set of data items so that the most "similar" data items are grouped together. The measure of "dissimilarity" between two elements of two different groups is to be maximum. Here, one sees the important role of similarity measure in the clustering procedure. The concept of similarity must be well defined and available to the clustering procedure even before choosing the clustering algorithm.

6. Method of Clustering: The term "cluster" is not an abstract term with a fixed mathematical definition. The term "cluster" in classification , pattern recognition, cluster analysis and other related areas is treated much like the term "point" in mathematics. Depending on the set of data items, the purpose of clustering, the set of actions, the set of needs and some other factors, one could choose the appropriate method of clustering.

7. Algorithm: Once the data items are defined by homogenized variables, the measure of similarity (or dissimilarity) is defined and the criterion of clustering is chosen, an algorithm is needed to specify the steps of the clustering procedure as well as a computer program to implement this algorithm. The choice of algorithm is important to the result of clustering. To choose the appropriate algorithm sufficient information must be known about the capacity of the available computer system, the size of the set of data items , the number of variables, and the method of clustering.

8. Number of Clusters: In some of the clustering methods, like non-hierarchical clustering, one needs to specify the number of clusters as a parameter to the system or as a parameter to the algorithm. In some other type of clustering the number of clusters does not have to be specified during the operation of the system. Since most of the existent algorithms are not defined so as to be able to decide the number of clusters, it seems reasonable that the analyst for the clustering system keep accurate and timely records on this parameter.

9. The Set of Needs: There is always at least one purpose for using any clustering procedure. When one classifies his groceries to cold, meat, frozen, canned food and dry food he might want to satisfy the following needs:

- a. To have easier access to his groceries;
- b. To put the cold bags in the cooler in the back of his car;
- c. To unload the cold foods before the canned foods.

The set of needs resulting in classification of groceries consists of those listed above. In general, whatever goals one wishes to reach by means of clustering are called the set of needs.

10. The Set of Actions: To satisfy the needs, a set of appropriate actions must be applied to the clusters in a certain order. Application of the set of actions to the generated clusters is intended to improve, in some sense, the behavior of the members of the cluster as well as the cluster itself. Since there is a unique sequence of actions corresponding to each cluster, one could represent a cluster by its action sequence.

11. Acceptor Block: In this new method of clustering the procedure of clustering is to generate a partition $\mathcal{P} = \{C_1, C_2, \dots, C_m\}$ on the set of data items. Also, the procedure of clustering is to associate an action sequence A_i with the cluster C_i . The action sequence corresponding to C_i should be based on the set of needs as well as on the members of C_i . The pair (C_i, A_i) identifies both the group of data items and the actions which the members of the group should do. In this method the pair (C_i, A_i) is called a cluster. Before applying the action sequence A_i to the group C_i , it is reasonable to test the applicability of A_i to the individual members of C_i as well as to the group C_i . An algorithm which tests the applicability of A_i for the group C_i is needed. The

block containing this algorithm is called the acceptor block.

12. Analysis of The Result: At the time one starts collecting data items for clustering analysis, his knowledge about the data items is non-existent or very small. By the time the procedure of clustering is done , at least the following information is available.

- a. the similarity of data items in terms of a set of given objects;
- b. the reaction of data items to the application of the action sequences;
- c. the effect of applying the action sequences to the individuals;
- d. the relation between data items of one group;
- e. the relation between groups;
- f. the measure of "improvement;"
- g. the convergence direction of the clusters as well as individual members of clusters.

Determining the degree of achievement of the goals, effectiveness of the procedure, efficiency of the algorithm and effectiveness of the application of the sequence of actions to the individuals are some of the factors which could be measured by analyzing the results of the GOCS.

The GOCS is the solution to the five major problems given in section 1.1. The GOCS is introduced in this research and the direction for additional work on the GOCS is given in Chapter six.

CHAPTER III

NUMERIC REPRESENTATION OF NNV FOR CLUSTER ANALYSIS

3.1 Introduction:

To cluster a set of data items $D = \{D_1, D_2, \dots, D_N\}$ it is necessary to represent each data item D_i by a set of n variables. Some of these variables are such that one can associate a single number to every one of them. This type of variable is called a numeric variable. For example age and income are two numeric variables. Some variables are such that they cannot be represented by a single number. That is, the meaning of this type of variable cannot be represented by a single number as it could be for age and income. This second type of variable is of nominal scale, but the reason that it is considered a different type is that the nominal category of this type of variable is not certain. For example, let X represent the behavior of a person P , from a population in which M different types of behavior have been recognized. Usually, there is not enough information available to define the type of behavior for person P as one of the M types of existing behavior. That is why X cannot be represented by a single number. This type of variable, which has been referred to as a NNV in Chapter One, has not previously been used in cluster analysis. In this chapter a numeric representation for NNV's is proposed in a manner that could be used for cluster analysis. Anderberg (2)

gives a complete listing of problems concerning numeric variables.

3.2 Category Relation:

Let X be a NNV and A be the range of X . It is assumed that the range of every NNV is a finite set. For example, assume X to be the behavior of an object Q and $A = \{A_1, A_2, \dots, A_M\}$ to be the range of X . $A_1, A_2, \dots$, and A_M are M different types of behavior. The definition of the elements of the set A as well as the number of these elements are dependent on the given problem.

Corresponding to each NNV X , and its range A , define a set A' which is the set of the original population related to X . Corresponding to each element A_i of the set A , define a set A'_i which is a subset of A' and contains those members of the original population which have the same value A_i for the NNV X . For example, if X is a NNV, say behavior, and $A_1, A_2, \dots, A_M$ are M different types of behavior then A'_i is the set of all members of the original population which have behavior of type A_i . The following paragraph describes some of the above notations.

Suppose the problem is to cluster a set of 1000 data items (persons) to give them a one-year industrial job training. Every individual data item is defined by the following six variables: V_1 =age, V_2 =income, V_3 = marital status, V_4 =background, V_5 = behavior, and V_6 = performance. The original population with respect to V_5 from which the given sample is taken is of size 100,000. Let A be the set of all types of behavior which have been identified in the original population. Based on the

problem under study (clustering for industrial job training) one can arbitrarily assume there are 4 different types of behavior, that is, $A = \{A_1, A_2, A_3, A_4\}$. Let n_i be the number of people with behavior of type A_i , that is, $|A'_i| = n_i$. For this example let $n_1 = 25,000$, $n_2 = 37,000$, $n_3 = 28,000$ and $n_4 = 10,000$. This means 25,000 people of this population have behavior of type A_1 , 37,000 have behavior of type A_2 and so on. It is clear that if one exchanges the problem given above with the problem "cluster a set of 1000 persons to give them a one-year job training in salesmanship" the definition of the different types of behavior might change as well as their number.

Definition 3.1: Let X be a NNV and A be the range of X . Let P and Q be two members of the set A' (the original population related to X .) Define a relation R on the set A' as follows: Let P and Q be two elements of A' . Then P is said to be related to Q if and only if they have the same category definition in terms of X . R is called the category relation.

It is easy to verify that R , the category relation defined by definition 3.1, is an equivalence relation.

Lemma 3.1: A category relation R induces a partition

$$\mathcal{T} = \{A'_1, A'_2, \dots, A'_M\} \text{ over the set } A'.$$

Proof: Let the equivalence classes of the equivalence relation R be $B_1, B_2, \dots, B_k$. It is sufficient to show that the partition $T = \{B_1, B_2, \dots, B_k\}$ is the same as the partition $\mathcal{T} = \{A'_1, A'_2, \dots, A'_M\}$. Let P and Q be any two elements of B_i .

This implies P and Q have the same category definition; and this further implies P and Q belong to a unique subset of A' , say A'_j . Therefore, B_i is a subset of A'_j . On the other hand let U be any member of A'_j other than P and Q , if such an element exists. This implies P and U have the same category definition which in turn implies P and U are related. Since the B_i 's are disjoint, P and U have to belong to B_i . Therefore, $B_i = A'_j$; and this is true for all $i = 1, 2, \dots, k$ and $j = 1, 2, \dots, M$. In the case in which B_i has less than two elements it is easy to show that there must exist a corresponding A'_j with only one element in it. QED.

Note 1: For a given NNV X with range A and population A' the partition $\mathcal{A} = \{A'_1, A'_2, \dots, A'_M\}$ as well as the category relation R are not unique. That is, $\mathcal{A}$ and R might change from one problem to another. This is true because the definition of the elements of the set A (categories of X) might change by the nature of the given problem.

Note 2: Depending on the nature of a given problem one might need to assign a set of weights to the elements of the set A , the range of a NNV X . These weights could be any real number in the interval $(0, 1]$. If no weight is assigned to an element A_i it is assumed that the weight of A_i is 1. The weights corresponding to a set A of M elements are given by a vector $W_A = (w_1, w_2, \dots, w_M)$ or by a diagonal matrix with weights on its diagonal.

Definition 3.2: Let $A = \{A_1, A_2, \dots, A_M\}$ be the range of a NNV X and A' be the finite population related to A . The density degree of an element A_i , denoted by $dd(A_i)$, is defined

by

$$dd(A_i) = \frac{|A'_i|}{|A'|} \quad \text{for } i=1,2,\dots,M.$$

If A' is an infinite set the density degree of A_i is defined by definition 3.3.

Definition 3.3: Let A and A' be as defined in definition 3.2 except that A' be an infinite set. Let k be the number of elements of partition $\mathcal{A} = \{A'_1, A'_2, \dots, A'_M\}$ which are infinite. Define $dd(A_i)$ as follows:

$$dd(A_i) = \begin{cases} 0 & \text{if } A'_i \text{ is a finite element of } \mathcal{A} \\ 1/k & \text{if } A'_i \text{ is an infinite element of } \mathcal{A}. \end{cases}$$

Definition 3.4: Let f be a function from $S \times S$ to the reals, where S is any set. Then f is called a distance function if:

- a. $f(X, Y) \geq 0$, $\forall X, Y \in S$;
- b. $f(X, Y) = 0$ iff $X=Y$ $\forall X, Y \in S$;
- c. $f(X, Y) = f(Y, X)$ $\forall X, Y \in S$.

If in addition to the properties given above one has

- d. $f(X, Y) + f(Y, Z) \geq f(X, Z)$ $\forall X, Y, Z \in S$

then f is called an Euclidian distance function or a metric.

If for a function f and a set S one has the properties (a) and (c) of definition 3.4 plus the fact that

$$f(X, X) = 0 \quad \forall X \in S$$

then f is called a psuedo distance. A psuedo distance which satisfies the triangle inequality is called a psuedo metric.

For example, one can define λ on $A' \times A'$ as follows:

$$\lambda(P, Q) = |dd(A_i) - dd(A_j)|$$

where $P \in A'_i$ and $Q \in A'_j$. The A_i 's and A_j 's are as defined in the beginning of this section. It is easy to verify that λ is a

psuedo metric.

3.3 Representation of a NNV:

Throughout the remainder of this chapter assume X is a NNV defined on the set A , A' is the population set corresponding to X and $\pi_A = \{A'_1, A'_2, \dots, A'_M\}$ is a partition on the set A' . Also, assume $W_A = (w_1, w_2, \dots, w_M)$ is a weight vector corresponding to the partition π_A .

Given a NNV X with range A and population A' and a data item Q there are three different possibilities as follows:

1. There is no information for the data item Q with respect to the NNV X . In this case it is assumed Q is uniformly distributed over A' . That is, $P(Q \in A'_i) = dd(A_i)$.

2. Q is uniformly distributed over Γ , a subset of π_A , with probability p . In this case

$$P(Q \in A'_i) = \begin{cases} (1-p) \frac{|A'_i|}{|D|} & \text{if } C'_i = \emptyset \\ p(dd(C_i)) & \text{if } C'_i \neq \emptyset, \end{cases}$$

where $\Gamma = \{A'_{s1}, A'_{s2}, \dots, A'_{sk}\}$, $B' = \bigcup_{i=1}^k A'_{si}$, $D = A' - B'$, $C'_i = A'_i \cap B'$ and $dd(C_i) = |C'_i| / |B'|$.

3. Q is distributed over Γ , a subset of π_A , with distribution function $(p_{s1}, p_{s2}, \dots, p_{sk})$ such that $\sum_{i=1}^k p_{si} = p$. In this case

$$P(Q \in A'_i) = \begin{cases} (1-p) \frac{|A'_i|}{|D|} & \text{if } C'_i = \emptyset \\ p_{sj} & \text{if } C'_i \neq \emptyset \text{ and } i=s_j, \end{cases}$$

where D , B' , C'_i and Γ are the same as defined in case 2.

Note that in cases 2 and 3 it is assumed that Q is uniformly distributed over $A' - B'$. If one has complete information about the data item Q then $A' - B'$ is an empty set.

Definition 3.5: For a given NNV X and a data item Q define

$$P_x = (P(Q \in A_1), P(Q \in A_2), \dots, P(Q \in A_M)).$$

The vector P_x is called the distribution vector of X for the data item Q .

If there is any weight vector corresponding to the range of a NNV X then one might represent X by the vector

$$V_x = (w_1 P(Q \in A_1'), w_2 P(Q \in A_2'), \dots, w_M P(Q \in A_M')).$$

The following example is to explain the notations P_x and V_x .

Example 3.1: Let X be a NNV with range $A = \{A_1, A_2, A_3\}$ and population of size 10. Let $\pi_A = \{A_1', A_2', A_3'\}$ be a partition over A' with $|A_1'| = 3$, $|A_2'| = 3$ and $|A_3'| = 4$ and $w_A = (1/3, 1, 1/4)$ be a weight vector corresponding to A (or π_A). Let $Q \in A'$ and assume it is known that $P(Q \in A_2') = 7/10$. In this example $\Gamma = A_2'$, $D = \{A_1', A_3'\}$, $|D| = 7$, $P(Q \in A_1') = (1 - 7/10)3/70 = 9/70$, and $P(Q \in A_3') = (1 - 7/10)4/70 = 12/70$. The distribution vector of X for the data item Q is given by $P_x = (9/70, 49/70, 12/70)$, and V_x is given by $V_x = (3/70, 49/70, 3/70)$.

3.4 Representation of a Data Item:

Let $D = \{D_1, D_2, \dots, D_N\}$ be a set of N data items and n be the number of variables defining each data item. If all n variables are numeric variables then each data item D_i can be represented by a row vector $D_i = (v_{i1}, v_{i2}, \dots, v_{in})$. The set of all data items can be represented by an $N \times n$ matrix, when all variables are numeric.

An alternative is to have s numeric variables and $n-s$ NNV's. Assume that the first s variables are numeric and the last $n-s$ variables are NNV's. Let π_{A_i} be a partition of order

n_i over the population of the NNV V_i , $i=s+1, s+2, \dots, n$. Define $M = \text{Maximum}(n_i)$. Represent each numeric variable by an M -dimensional row vector with the first component equal to the actual value of the variable and the next $M-1$ components equal to 0. For NNV V_i , if $n_i < M$, add $M-n_i$ zeroes to the end of both the distribution vector and the weight vector corresponding to V_i and denote them by V'_i and P'_{V_i} , respectively. The vector P'_{V_i} is called the augmented distribution vector of V_i .

$$P'_{V_i} = (p_{i1}, p_{i2}, \dots, p_{in_i}, \underbrace{0, 0, \dots, 0}_{M-n_i})$$

and

$$V'_i = (w_{i1}p_{i1}, w_{i2}p_{i2}, \dots, w_{in_i}p_{in_i}, 0, 0, \dots, 0),$$

for $i = s+1, s+2, \dots, n$, where $(p_{i1}, p_{i2}, \dots, p_{in_i})$ is the distribution vector of V_i and $(w_{i1}, w_{i2}, \dots, w_{in_i})$ is the weight vector of V_i . Using vector representation of NNV's, each data item can be represented by an $n \times M$ matrix as follows:

$$D^i = \begin{bmatrix} v_{11}^i & 0 & \dots & 0 \\ v_{21}^i & 0 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ p_{s+1,1}^i & p_{s+1,2}^i & \dots & p_{s+1,M}^i \\ p_{s+2,1}^i & p_{s+2,2}^i & \dots & p_{s+2,M}^i \\ \vdots & \vdots & \dots & \vdots \\ p_{n,1}^i & p_{n,2}^i & \dots & p_{n,M}^i \end{bmatrix}.$$

If the NNV's are represented by V'_i 's (there are weight vectors involved) then one can replace $p_{s+j,k}^i$ with $w_{s+j,k} p_{s+j,k}^i$ in D^i and denote the resultant matrix by D_w^i .

In Chapter Four it is shown how one can associate a dissimilarity measure to D_i and D_j using the matrices D_w^i and D_w^j .

The letter "D" with no subscript stands for the set of data items, with subscript "i" stands for the i^{th} data item, with superscript "i" stands for the matrix representation of the i^{th} data item, and with superscript "i" and subscript "w" stands for the matrix representation of the i^{th} data item when the weight vectors are effective.

It should be mentioned that the representation of a data item, defined by NNV's, proposed in this chapter is one of the possible ways to represent such data items. Of course, there are some other ways to represent a data item defined by NNV's. For example, an alternative method for representing a data item defined by s numeric variables and n -s NNV's is as follows: Suppose the first s variables are numeric and the last n -s are NNV's. Let the number of categories of the K^{th} NNV be n_k . The column vector

$$W^i = (v_1^i, v_2^i, \dots, v_s^i, p_{s+1,1}^i, \dots, p_{s+1,n_{s+1}}^i, \dots, p_{n,1}^i, \dots, p_{n,n_n}^i)^T$$

is an alternative form to represent the i^{th} data item.

In this alternative representation of data items, the components of the vector W^i all are treated as individual numeric variables when dealing with clustering algorithms. That is, W^i is treated as a data item of $s+n_{s+1}+n_{s+2}+\dots+n_n$ numeric variables. To see how this problem would affect the result of a clustering analysis one has to collect a set of real data and check the resulting clusters for being a well optimized set of clusters. Since the data items of NNV's are clustered for the first time in this thesis, the representations and clustering methods proposed in this thesis could be a base for comparison for the future work in this area.

Example 3.2: Let each data item D_i be defined by six variables, where U, V, W are three numeric variables and X, Y, Z are three NNV's. To be specific, let each data item be a person defined by U =age, V = income, W =marital status, X =behavior, Y = background, and Z = performance. X, Y , and Z are defined on some sets A, B , and C with population A', B' , and C' , respectively. The original population for all three NNV's is the same and it is assumed to be 10. That is, $|A'| = |B'| = |C'| = 10$. The corresponding partitions are:

$$\pi_A = \{A'_1, A'_2, A'_3\}, \pi_B = \{B'_1, B'_2, B'_3, B'_4, B'_5\} \text{ and } \pi_C = \{C'_1, C'_2\}$$

and the subpopulations are: $|A'_1| = |A'_2| = 3$, $|A'_3| = 4$,
 $|B'_1| = |B'_2| = 3$, $|B'_3| = 2$, $|B'_4| = |B'_5| = 1$, $|C'_1| = |C'_2| = 5$.

The weight vectors are:

$$W_A = (.1, .2, .3), W_B = (.3, .2, .1, .4, .5) \text{ and } W_C = (.1, .2).$$

Let assume for person D_i it is known that:

$$P(D_i \in A'_2) = 1/2, P(D_i \in B'_3) = 3/5 \text{ and } P(D_i \in C'_1) = 2/3.$$

The distribution vectors of X, Y , and Z for person D_i when computed are as follows:

$(3/14, 1/2, 4/14)$ for X , $(2/3, 1/3)$ for Z and $(6/70, 6/70,$

$42/70, 8/70, 8/70)$ for Y . The augmented weight and distribution vectors are:

$$W_A = (.1, .2, .3, 0, 0),$$

$$W_B = (.3, .2, .1, .4, .5),$$

$$W_C = (.1, .2, 0, 0, 0),$$

$$P_X = (3/14, 1/2, 4/14, 0, 0),$$

$$P_Y = (6/70, 6/70, 42/70, 8/70, 8/70),$$

$$P_Z = (2/3, 1/3, 0, 0, 0).$$

Let the person D_i be 34 years old ($U=34$) with \$15900 income

($V=15900$) and married($W=0$). The person D_i^* could be represented by the following 6×5 matrix. (The weight vectors are considered.)

$$D_w^i = \begin{bmatrix} 34 & 0 & 0 & 0 & 0 \\ 15900 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 3/140 & 2/20 & 12/140 & 0 & 0 \\ 18/700 & 12/700 & 42/700 & 32/700 & 40/700 \\ 2/30 & 2/30 & 0 & 0 & 0 \end{bmatrix}.$$

Let D_j be another person defined by the same six variables and represented by the following information:

$$U = 25, V = 12000, W = 1, P_x = (1/7, 3/5, 9/35),$$

$$P_y = (0, 2/55, 10/55, 40/55, 3/55) \text{ and } P_z = (3/5, 2/5).$$

The matrix representation of D_j using the weight vectors is:

$$D_w^j = \begin{bmatrix} 25 & 0 & 0 & 0 & 0 \\ 12000 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1/70 & 6/50 & 27/350 & 0 & 0 \\ 0 & 4/550 & 10/550 & 160/550 & 15/550 \\ 3/50 & 4/50 & 0 & 0 & 0 \end{bmatrix}.$$

In general, let $D = \{D_1, D_2, \dots, D_N\}$ be a set of N data items, n be the number of variables (consider numeric variables as NNV's with partition of order 1), and M be the maximum order of the partitions of the population of the variables. Define D^i , D_w^i , $\underline{D}$, and $\underline{D}_w$ as follows:

* Using the alternative way to represent D_i one has
 $D_w^i = (34, 15900, 0, 3/140, 2/20, 6/70, 18/700, 12/700, 42/700, 32/700, 40/700, 2/30, 2/30)^T$.

$$D^i = [p_{jk}^i] = \begin{bmatrix} p_{11}^i & p_{12}^i & \cdots & p_{1M}^i \\ p_{21}^i & p_{22}^i & \cdots & p_{2M}^i \\ \vdots & \vdots & \ddots & \vdots \\ p_{n1}^i & p_{n2}^i & \cdots & p_{nM}^i \end{bmatrix}$$

where p_{jk}^i is the k^{th} component of the augmented distribution vector of the j^{th} variable for the i^{th} data item;

$$D_w^i = [w_{jk} p_{jk}^i]$$

where w_{jk} is the k^{th} component of the weight vector corresponding to the j^{th} variable;

$$\underline{D} = [D^1 : D^2 : \cdots : D^N],$$

and

$$\underline{D}_w = [D_w^1 : D_w^2 : \cdots : D_w^N].$$

The matrix D^i is an augmented $n \times M$ matrix containing all information about the i^{th} data item, excluding the weight vectors; $\underline{D}$ is an N -block matrix containing all available information about the whole set of data items, excluding the weight vectors. D_w^i contains the same information as D^i plus the weight vectors and $\underline{D}_w$ contains the same information as $\underline{D}$ plus the weight vectors. Throughout the remainder of this chapter consider $M, N, n, D, D^i, D_w^i, \underline{D}$, and $\underline{D}_w$ as defined in this section.

3.5 Representation of a Variable:

Let $D = \{D_1, D_2, \dots, D_N\}$ be a set of N data items, each one represented by n variables (s numeric and $n-s$ NNV). A NNV X_i is defined on a set A_i with population set A_i^1 and π_{A_i} is a partition of order n_i on the set A_i^1 ;

$$\pi_{A_i} = \{A_{i1}^1, A_{i2}^1, \dots, A_{in_i}^1\}$$

for $i=s+1, s+2, \dots, n$. According to section 3.3 each NNV can be represented by its distribution vector $p_{x_i}^k = (p_{i1}^k, p_{i2}^k, \dots, p_{in_i}^k)$ or by its augmented distribution vector

$$p_{x_i}^k = (p_{i1}^k, p_{i2}^k, \dots, p_{in_i}^k, 0, 0, \dots, 0),$$

where p_{ij}^k is the j^{th} component of the distribution vector of the i^{th} variable for the k^{th} data item. Therefore, $p_{x_i}^k$ is the distribution vector of the i^{th} NNV for the k^{th} data item.

An alternative for representing the i^{th} NNV for the k^{th} data item is

$$v_{x_i}^k = p_{x_i}^k \cdot W_{A_i}$$

where W_{A_i} is the weight matrix corresponding to the NNV X_i and is given by

$$W_{A_i} = \begin{bmatrix} w_1 & 0 & 0 & \dots & 0 \\ 0 & w_2 & 0 & \dots & 0 \\ 0 & 0 & w_3 & \dots & 0 \\ \cdot & \cdot & \cdot & \dots & \cdot \\ \cdot & \cdot & \cdot & \dots & \cdot \\ 0 & 0 & 0 & \dots & w_{n_i} \end{bmatrix}.$$

The weight matrix W_{A_i} is the same as the weight vector $(w_1, w_2, \dots, w_{n_i})$ but in a different notation to make multiplication possible. The augmented weight matrix of X_i is given by

$$W'_{A_i} = \begin{bmatrix} w_1 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & w_2 & \dots & 0 & 0 & \dots & 0 \\ \cdot & \cdot & \dots & \cdot & \cdot & \dots & \cdot \\ \cdot & \cdot & \dots & \cdot & \cdot & \dots & \cdot \\ 0 & 0 & \dots & w_{n_i} & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & 0 & \dots & 0 \\ \cdot & \cdot & \dots & \cdot & \cdot & \dots & \cdot \\ \cdot & \cdot & \dots & \cdot & \cdot & \dots & \cdot \\ 0 & 0 & \dots & 0 & 0 & \dots & 0 \end{bmatrix}.$$

For a numeric variable , X_i , an N-dimensional vector

$$(x_{1i}, x_{2i}, \dots, x_{Ni})^T$$

will represent the set of N values of the variable X_i for N data items $D_1, D_2, \dots$, and D_N ; x_{ki} is the value of X_i for the k^{th} data item. If X_i is a NNV then the matrix

$$X^i = \begin{bmatrix} p_{11}^i & p_{12}^i & \dots & p_{1n_i}^i \\ p_{21}^i & p_{22}^i & \dots & p_{2n_i}^i \\ \vdots & \vdots & \dots & \vdots \\ \vdots & \vdots & \dots & \vdots \\ p_{N1}^i & p_{N2}^i & \dots & p_{Nn_i}^i \end{bmatrix}$$

will represent the set of N values of X_i corresponding to N different data items $D_1, D_2, \dots$, and D_N . The j^{th} row of the matrix X^i is the distribution vector of X_i for the j^{th} data item. One could use the augmented distribution vectors of X_i . In this case X^i is an $N \times M$ augmented matrix , where $M = \text{Maximum}(n_i)$. An alternative form for representing the set of all N values of a NNV X_i is

$$X_w^i = X_{A_i}^i ,$$

where X^i is the augmented $N \times M$ matrix as defined above and W_{A_i} is the augmented weight matrix corresponding to X_i .

The whole information about a set of N data items of n variables (NNV or Numeric) can be stored in a 3-dimensional array as follows:

$$\underline{X} = [p_{jk}^i] ,$$

where $i=1,2,\dots,n$ stands for number of variables, $j=1,2,\dots,N$ stands for number of data items , and $k=1,2,\dots,M$. Another form for representing $\underline{X}$ could be

$$\underline{X} = [x^1; x^2; \dots; x^n],$$

where each X^i is an augmented $N \times M$ matrix. One might use X_w^i to represent each variable. In this case if X_w^i and W_{A_i} are augmented matrices one could define X_w as follow:

$$X_w = [X_w^1; X_w^2; \dots; X_w^n].$$

In chapter four it is shown how one can associate a correlation (dissimilarity) measure to X_i and X_j using the matrices X_w^i and X_w^j .

The following example is to explain the notations and definitions given in this section.

Example 3.3: Three data items D_1, D_2 and D_3 are defined by two variables X and Y with ranges A and B and populations A' and B' , respectively. The partitions of A' and B' are:

$$\pi_A = \{A'_1, A'_2, A'_3\} \text{ and } \pi_B = \{B'_1, B'_2, B'_3, B'_4\},$$

respectively. In this example one has $n=2$, $N=3$ and $M=\text{Max}(3,4)=4$.

The following information is given for each data item:

for the first data item

$$P(D_1 \in A'_1) = P(D_1 \in A'_2) = P(D_1 \in A'_3) = 1/3 \text{ and}$$

$$P(D_1 \in B'_1) = P(D_1 \in B'_4) = 1/2, \quad P(D_1 \in B'_2) = P(D_1 \in B'_3) = 0;$$

for the second data item

$$P(D_2 \in A'_1) = P(D_2 \in A'_2) = 1/2, \quad P(D_2 \in A'_3) = 0, \text{ and}$$

$$P(D_2 \in B'_1) = P(D_2 \in B'_2) = P(D_2 \in B'_4) = 1/3, \quad P(D_2 \in B'_3) = 0;$$

for the third data item

$$P(D_3 \in A'_1) = P(D_3 \in A'_3) = 1/4, \quad P(D_3 \in A'_2) = 1/2 \text{ and}$$

$$P(D_3 \in B'_1) = P(D_3 \in B'_2) = P(D_3 \in B'_3) = P(D_3 \in B'_4) = 1/4.$$

The weight vectors are:

$$W_A = (.1, .2, .3) \text{ and } W_B = (.1, .2, .3, .4).$$

The augmented weight matrices are:

$$W_A = \begin{bmatrix} .1 & 0 & 0 & 0 \\ 0 & .2 & 0 & 0 \\ 0 & 0 & .3 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad W_B = \begin{bmatrix} .1 & 0 & 0 & 0 \\ 0 & .2 & 0 & 0 \\ 0 & 0 & .3 & 0 \\ 0 & 0 & 0 & .4 \end{bmatrix}.$$

The augmented distribution vectors are:

$$\begin{aligned} p_x^1 &= (1/3, 1/3, 1/3, 0), & p_y^1 &= (1/2, 0, 0, 1/2), \\ p_x^2 &= (1/2, 1/2, 0, 0), & p_y^2 &= (1/3, 1/3, 0, 1/3), \\ p_x^3 &= (1/4, 1/2, 1/4, 0), & p_y^3 &= (1/4, 1/4, 1/4, 1/4). \end{aligned}$$

The augmented matrices x^1 , x^2 , x_w^1 , and x_w^2 (assuming $X=X_1$ and $Y=X_2$) are:

$$x^1 = \begin{bmatrix} 1/3 & 1/3 & 1/3 & 0 \\ 1/2 & 1/2 & 0 & 0 \\ 1/4 & 1/2 & 1/4 & 0 \end{bmatrix}, \quad x^2 = \begin{bmatrix} 1/2 & 0 & 0 & 1/2 \\ 1/3 & 1/3 & 0 & 1/3 \\ 1/4 & 1/4 & 1/4 & 1/4 \end{bmatrix},$$

$$x_w^1 = \begin{bmatrix} 1/3 & 1/3 & 1/3 & 0 \\ 1/2 & 1/2 & 0 & 0 \\ 1/4 & 1/2 & 1/4 & 0 \end{bmatrix} \begin{bmatrix} .1 & 0 & 0 & 0 \\ 0 & .2 & 0 & 0 \\ 0 & 0 & .3 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 1/30 & 2/30 & 3/30 & 0 \\ 1/20 & 2/20 & 0 & 0 \\ 1/40 & 2/20 & 3/40 & 0 \end{bmatrix},$$

$$x_w^2 = \begin{bmatrix} 1/2 & 0 & 0 & 1/2 \\ 1/3 & 1/3 & 0 & 1/3 \\ 1/4 & 1/4 & 1/4 & 1/4 \end{bmatrix} \begin{bmatrix} .1 & 0 & 0 & 0 \\ 0 & .2 & 0 & 0 \\ 0 & 0 & .3 & 0 \\ 0 & 0 & 0 & .4 \end{bmatrix} = \begin{bmatrix} 1/20 & 0 & 0 & 4/20 \\ 1/30 & 2/30 & 0 & 4/30 \\ 1/40 & 2/40 & 3/40 & 4/40 \end{bmatrix}.$$

The two-block matrices $\underline{X}$ and $\underline{X}_w$ are:

$$\underline{X} = \left[\begin{bmatrix} 1/3 & 1/3 & 1/3 & 0 \\ 1/2 & 1/2 & 0 & 0 \\ 1/4 & 1/2 & 1/4 & 0 \end{bmatrix} : \begin{bmatrix} 1/2 & 0 & 0 & 1/2 \\ 1/3 & 1/3 & 0 & 1/3 \\ 1/4 & 1/4 & 1/4 & 1/4 \end{bmatrix} \right],$$

$$\underline{X}_w = \left[\begin{array}{cccc} \left[\begin{array}{cccc} 1/30 & 2/30 & 3/30 & 0 \end{array} \right] & \vdots & \left[\begin{array}{cccc} 1/20 & 0 & 0 & 4/20 \end{array} \right] \\ \left[\begin{array}{cccc} 1/20 & 2/20 & 0 & 0 \end{array} \right] & \vdots & \left[\begin{array}{cccc} 1/30 & 2/30 & 0 & 4/30 \end{array} \right] \\ \left[\begin{array}{cccc} 1/40 & 2/20 & 1/40 & 0 \end{array} \right] & \vdots & \left[\begin{array}{cccc} 1/40 & 2/40 & 3/40 & 4/40 \end{array} \right] \end{array} \right] .$$

The matrices D^1 , D^2 , D^3 , D_w^1 , D_w^2 , and D_w^3 are:

$$\begin{aligned} D^1 &= \begin{bmatrix} 1/3 & 1/3 & 1/3 & 0 \\ 1/2 & 0 & 0 & 1/2 \end{bmatrix}, & D_w^1 &= \begin{bmatrix} 1/30 & 2/30 & 3/30 & 0 \\ 1/20 & 0 & 0 & 4/20 \end{bmatrix} \\ D^2 &= \begin{bmatrix} 1/2 & 1/2 & 0 & 0 \\ 1/3 & 1/3 & 0 & 1/3 \end{bmatrix}, & D_w^2 &= \begin{bmatrix} 1/20 & 2/20 & 0 & 0 \\ 1/30 & 2/30 & 0 & 4/30 \end{bmatrix} \\ D^3 &= \begin{bmatrix} 1/4 & 1/2 & 1/4 & 0 \\ 1/4 & 1/4 & 1/4 & 1/4 \end{bmatrix}, & D_w^3 &= \begin{bmatrix} 1/40 & 2/20 & 3/40 & 0 \\ 1/40 & 2/40 & 3/40 & 4/40 \end{bmatrix}. \end{aligned}$$

The three-block matrices $\underline{D}$ and $\underline{D}_w$ are:

$$\begin{aligned} \underline{D} &= \left[\begin{array}{cccc} \left[\begin{array}{cccc} 1/3 & 1/3 & 1/3 & 0 \end{array} \right] & \vdots & \left[\begin{array}{cccc} 1/2 & 1/2 & 0 & 0 \end{array} \right] & \vdots & \left[\begin{array}{cccc} 1/4 & 1/2 & 1/4 & 0 \end{array} \right] \\ \left[\begin{array}{cccc} 1/2 & 0 & 0 & 1/2 \end{array} \right] & \vdots & \left[\begin{array}{cccc} 1/3 & 1/3 & 0 & 1/3 \end{array} \right] & \vdots & \left[\begin{array}{cccc} 1/4 & 1/4 & 1/4 & 1/4 \end{array} \right] \end{array} \right], \\ \underline{D}_w &= \left[\begin{array}{cccc} \left[\begin{array}{cccc} 1/30 & 2/30 & 3/30 & 0 \end{array} \right] & \vdots & \left[\begin{array}{cccc} 1/20 & 2/20 & 0 & 0 \end{array} \right] & \vdots & \left[\begin{array}{cccc} 1/40 & 2/20 & 3/40 & 0 \end{array} \right] \\ \left[\begin{array}{cccc} 1/20 & 0 & 0 & 4/20 \end{array} \right] & \vdots & \left[\begin{array}{cccc} 1/30 & 2/30 & 0 & 4/30 \end{array} \right] & \vdots & \left[\begin{array}{cccc} 1/40 & 2/40 & 3/40 & 4/40 \end{array} \right] \end{array} \right]. \end{aligned}$$

The elements of the k^{th} row of D^i (D_w^i) are the same as the elements of the i^{th} row of X^k (X_w^k). Therefore, to calculate $\underline{X}$ ($\underline{X}_w$) and $\underline{D}$ ($\underline{D}_w$) one has to calculate only X^k 's (X_w^k 's).

3.6 Some Motivation for the Proposed Representation:

The representations of a NNV (distribution vector) and a data item (an $n \times M$ matrix) are developed basically for cluster analysis of data items. That is, the representations proposed in this chapter are such that one could use them for the

purpose of clustering.

All changes in representation of variables start from introducing a distribution vector to represent the value of a variable for a given data item. Therefore, for simplicity, hereafter this new type of representation is referred to as "Distribution Vector Representation" (DVR). The previous representation of variables and data items is referred to as "Single Number Representation" (SNR) .

There are some advantages as well as disadvantages to the DVR method , concerning the cluster analysis problem. Some of the advantages are as follows:

1. Let X be a NNV with a range of M elements , that is , there are M nominal categories. It is known that the category definition of X for a given data item Q is not specifically defined. In the SNR method one either has to completely ignore this variable or make a rough guess in assigning a nominal value to X . In the first case, the question is; what happens if most of the variables defining the data item Q are NNV's such as in psychology and behavioral science? In the second case , the question is that of assigning a probability of $1/M$ to every category of the range of X . Comparing the distribution of X , in case of no available information , to the rough guess method one can easily see the difference. In the rough guess method , for N data items on the average N/M times one would be right. For example, let X be a NNV with range $A = \{A_1, A_2, A_3\}$, population 10,000 and the subpopulations $|A'_1| = 100$, $|A'_2| = 6000$ and $|A'_3| = 3900$.

Let a sample of size 450 be drawn from this population. With the rough guess method one would consider approximately 150 observations out of 450 to be of category one, in terms of NNV X. But it is known that even in the whole population there are only 100 observations of category one. This simple example shows the danger associated with the rough guess method.

2. The DVR method makes it possible to compare the values of a variable X for two data items. Let P and Q be two data items and let $P_X(P) = (p_1, p_2, \dots, p_M)$ be the DVR of a NNV X for the data item P and $P_X(Q) = (q_1, q_2, \dots, q_M)$ be the DVR of X for the data item Q. One can compare p_i to q_i for $i=1, 2, \dots, M$ as follows. If $p_i = q_i$ it means both P and Q have the same probability of being in category A_i . If $p_i > q_i$ then P has a larger probability of being in category A_i than Q does.

3. The DVR method can be applied to those nominal-scale variables which have certain category definitions. Let X be a nominal-scale variable and Q be a data item. Assume X has M categories $A_1, A_2, \dots, A_M$ and it is known that the category of Q is A_i . Consider $\Gamma = \{A_i\}$ and $p=1$ in case 2 of section 3.3. Note that in this case the subpopulation $|A_i|$ need not be known. In this case the distribution of X for the data item Q would be $(\underbrace{0, 0, \dots, 1}_{i-1}, \underbrace{0, 0, \dots, 0}_{M-i}, 0)$. These illustrate some of the advantages of the DVR method.

One of the disadvantages of the DVR method is the problem of storage and speed of computation. The amount of computer storage which is needed to operate a certain clustering algorithm when the data items are represented by the DVR method

is almost M times as much as when the data items are represented by the SNR method. This problem can be partially solved by using "The Storage Data Approach " methods; see Anderberg (2) Chapter Six. In this case the operation of clustering becomes significantly slow.

For some areas such as psychology and behavioral science, where a large number of NNV's are required to represent the data items the amount of time needed to solve the clustering problem when the DVR method is used is acceptable because of the advantages previously stated. In some areas where less NNV's (or even nominal-scale variables with certain category definition) are used to define the data items one should compare the advantages of using the DVR method in spite of its slowness. Note that slow speed computation will cause more computer time and greater cost. Also, note that the use of the DVR method will cause an increase in the number of computations as much as M times more than when the SNR method is used.

Hereafter, when the DVR method is chosen, it is assumed that a nominal-scale variable with a certain nominal value is a special case of a NNV as specified above.

CHAPTER IV

ASSOCIATION MEASURES

4.1 Introduction:

In clustering a set of N "objects" one usually deals with some type of numeric measure $\mathcal{M}$. The measure $\mathcal{M}$ is assumed to be an abstract number with ratio scale. It is also assumed that $\mathcal{M}$ is a mapping from $D \times D$ into a subset of real numbers, where D is the set of objects (data items).

Based on the nature of the measure $\mathcal{M}$ there are different terminologies available such as similarity measure, closeness measure, neighborhood distance and dissimilarity measure. It is assumed that $\mathcal{M}$ is a symmetric measure, that is,

$$\forall X, Y \in D \implies \mathcal{M}(X, Y) = \mathcal{M}(Y, X).$$

For N data items there are $\binom{N}{2} = N(N-1)/2$ different pairs.

Therefore, the function $\mathcal{M}$ is defined such that it has at most $N(N-1)/2$ possibly different values corresponding to the set D .

Let $D_1, D_2, \dots, D_N$ be N different elements of a set of data items.

Denote the value of the function $\mathcal{M}$ corresponding to the pair (D_i, D_j) by $\mathcal{M}(D_i, D_j) = d_{ij}$. The matrix S

$$S = \begin{bmatrix} d_{11} & d_{12} & \dots & d_{1N} \\ d_{21} & d_{22} & \dots & d_{2N} \\ \cdot & \cdot & \dots & \cdot \\ \cdot & \cdot & \dots & \cdot \\ d_{N1} & d_{N2} & \dots & d_{NN} \end{bmatrix}$$

is a symmetric matrix. Based on the nature of the measure $\mathcal{M}$ the matrix S might be called a similarity matrix, a dissimilarity

matrix or a distance matrix. If d_{ij} is the distance between D_i and D_j or if d_{ij} is the dissimilarity between D_i and D_j then all diagonal elements of the matrix d are zero, that is, $d_{ii} = 0$ for $i=1,2,\dots,N$.

Let S be a similarity matrix and t be a threshold given for the set of data items D . It is assumed that any two elements of the set D , say D_i and D_j , are similar if and only if $d_{ij} \geq t$. In most cases, as long as the magnitude of d_{ij} exceeds the threshold t , the value of d_{ij} is not important. In these cases it is more convenient to transform d_{ij} to a binary variable as follows:

$$d_{ij} = \begin{cases} 1 & \text{if } d_{ij} \geq t, \\ 0 & \text{if } d_{ij} < t. \end{cases}$$

If the d_{ij} 's are binary variables, S is called a binary matrix. To store a binary similarity matrix only $N(N-1)/2$ bits are needed, where S is an $N \times N$ matrix. If d is not a binary matrix $N(N-1)/2$ computer words are needed where each word may have as many as 60 bits. Therefore, transforming d_{ij} to a binary variable is a significant way to save machine memory.

4.2 Association Measure Between Variables:

This section is divided into two parts. In the first part some different types of measures between numeric variables are discussed. In the second part of this section some measures of association between NNV's are discussed.

Note that the association measure between variables is used for the purpose of clustering as well as feature reduction. In the case of using an association measure for the purpose of feature reduction, the term "correlation" is more meaningful for the measure between variables than the term

"dissimilarity." For example, in a set of N data items let $X = (x_1, x_2, \dots, x_N)$ represent N different values of the variable AGE and $Y = (y_1, y_2, \dots, y_N)$ represent N different values of the variable INCOME. Seeking a correlation measure between X and Y is more logical than seeking a similarity measure between them.

4.2.1 Association measure Between Numeric Variables:

It is assumed that all numeric variables involved in a clustering procedure are of interval scale (after solving the problem of mixed variables). Since this assumption has been made, it is sufficient to consider only the association measures between interval variables.

Let X and Y be two n -dimensional vectors given by

$$X = (x_1, x_2, \dots, x_n)^T \text{ and } Y = (y_1, y_2, \dots, y_n)^T.$$

a. The inner product of X and Y is defined by

$$\langle X, Y \rangle = X^T Y = Y^T X.$$

b. The Euclidian norm of the column vector X is defined by

$$\|X\| = \sqrt{\langle X, X \rangle}.$$

Define

$$d(X, Y) = \frac{X^T Y}{\|X\| \|Y\|}$$

and

$$r(X, Y) = \frac{\sum_{i=1}^n (x_i - \bar{X})(y_i - \bar{Y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{X})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{Y})^2}},$$

$$\text{where } \bar{X} = 1/n \sum_{i=1}^n x_i \text{ and } \bar{Y} = 1/n \sum_{i=1}^n y_i.$$

The measure $d(X, Y)$ is the cosine of the angle between the two column vectors X and Y ; and $r(X, Y)$ is called the moment correlation between X and Y . If X and Y are rank vectors then the

moment correlation[81] between them is defined by

$$\rho(X,Y) = 1 - 6 \frac{\sum_{i=1}^n (x_i - y_j)^2}{n(n^2 - 1)} .$$

The functions d , r and ρ all take values in the range $[-1,1]$. These three functions are some of the widely used functions for association measures between variables. Anderberg (2) chapter three is a good reference for further discussion about association measure between numeric variables.

4.2.2 Association Measure Between NNV's:

In chapter three, section 3.5, it was shown that a NNV can be represented by an $N \times M$ augmented matrix

$$x_w^i = [x_{jk}^i] = \begin{bmatrix} x_{11}^i & x_{12}^i & \dots & x_{1M}^i \\ x_{21}^i & x_{22}^i & \dots & x_{2M}^i \\ \vdots & \vdots & \dots & \vdots \\ x_{N1}^i & x_{N2}^i & \dots & x_{NM}^i \end{bmatrix}$$

where $x_{jk}^i = w_{ik} p_{ik}^j$, w_{ik} is the k^{th} component of the weight vector corresponding to the i^{th} variable, and p_{ik}^j is the k^{th} component of the distribution vector of the i^{th} variable for the j^{th} data item. With this representation in mind one might want to define some sort of association measure between NNV's X_i and X_j .

The problem of association measure between NNV's is approached in two different ways. In the first approach an average-association measure is suggested and in the second approach an exact association measure is given by using an M -dimensional distance function(to be defined.)

Definition 4.1: Let S be the set of all $n \times M$ matrices and V be the set of all non-negative n -dimensional vectors. Let $X = [x_{ij}]$ and $Y = [y_{ij}]$ be any two members of S . A function $\Gamma = (\gamma_1, \gamma_2, \dots, \gamma_M)$ from $S \times S$ into V is called an M -dimensional distance function if

- a. $\forall X, Y \in S \quad \Gamma(X, Y) = (e_1, e_2, \dots, e_M), \quad e_i \geq 0, \quad i=1, 2, \dots, M;$
- b. $\forall X, Y \in S \quad \Gamma(X, Y) = (0, 0, \dots, 0) \iff X=Y;$
- c. $\forall X, Y \in S \quad \Gamma(X, Y) = \Gamma(Y, X);$

where $e_i = \gamma_i(x_{\circ i}, y_{\circ i})$ for $i=1, 2, \dots, M$, γ_i is a distance function and $x_{\circ i}$ and $y_{\circ i}$ are the i^{th} columns of the matrices X and Y , respectively.

If all distance functions $\gamma_1, \gamma_2, \dots, \gamma_M$ satisfy the triangle inequality then Γ is called an M -dimensional Euclidian distance function or an M -dimensional metric. The norm of $\Gamma(X, Y)$ is denoted by

$$\|\Gamma(X, Y)\| = \sqrt{\sum_{i=1}^M e_i^2}.$$

If one or more of the γ_i 's is a psuedo distance then Γ is called a psuedo distance.

Definition 4.1 can be modified to map the set S into the set of non-negative M -dimensional column vectors. In this case one has $\beta = (\beta_1, \beta_2, \dots, \beta_n)^T$ and $\beta(X, Y) = (b_1, b_2, \dots, b_n)^T$, where $b_i = \beta_i(x_{i\circ}, y_{i\circ})$ and $x_{i\circ}$ and $y_{i\circ}$ are the i^{th} rows of the matrices X and Y , respectively.

Definition 4.2: Let $A = [a_{ij}]$ be an $N \times M$ matrix. The row vector $V = (v_1, v_2, \dots, v_M)$ with

$$v_j = \frac{1}{N} \sum_{i=1}^N a_{ij}$$

is called the average row vector of the matrix A. The column vector $U = (u_1, u_2, \dots, u_N)^T$ with

$$u_i = \frac{1}{M} \sum_{j=1}^N a_{ij}$$

is called the average column vector of the matrix A.

Remark: To be more meaningful about the measures between NNV's, the association measure is defined only between those variables which have ranges of the same size (or partitions of equal order or the same number of categories.) For example, in Example 3.3 there is no association measure defined between variables X and Y because the range of X has three elements and the range of Y has four elements.

Definition 4.3: Let X and Y be two M-dimensional vectors $X = (x_1, x_2, \dots, x_M)$ and $Y = (y_1, y_2, \dots, y_M)$. Define

$$\sigma(X, Y) = 1/M \sum_{i=1}^M |x_i - y_i|$$

The value of the function σ is called the mean deviation measure of X and Y. Define $\hat{\sigma}(X, Y) = \text{Minimum}_p (\sigma(X_p, Y_p))$, where X_p and Y_p are different permutations of X and Y. The value of the function $\hat{\sigma}$ is called the minimum mean deviation measure of X and Y.

It is easy to verify that σ is a metric. The first three properties of a metric are the immediate results of the definition of σ and the triangle inequality property of σ reduces to M triangle inequalities like the following one:

$$|x_i - y_i| + |y_i - z_i| \geq |x_i - z_i|$$

The function σ could be considered as a dissimilarity measure between two NNV's X_i and X_j when they are represented by the average row vectors $\bar{X}^i$ and $\bar{X}^j$. When all components of X

and Y are between zero and one, the range of σ is $[0,1]$.

It must be mentioned that, when one uses either the average row vectors representing NNV's or the β function for finding an association measure between NNV's, one must be sure that the function defining the association measure must be invariant under any permutation of the distribution vectors representing NNV's.

An alternative for finding an association measure between NNV's could be introduced by the N -dimensional distance function β . To use β one needs to specify $\beta_1, \beta_2, \dots, \beta_N$. For example, let $\beta_1 = \beta_2 = \dots = \beta_N = \hat{\sigma}$. This implies

$$\beta(X^i, X^j) = (\hat{\sigma}(x_{1\circ}^i, x_{1\circ}^j), \hat{\sigma}(x_{2\circ}^i, x_{2\circ}^j), \dots, \hat{\sigma}(x_{N\circ}^i, x_{N\circ}^j))^T$$

where $x_{k\circ}^i = (x_{k1}^i, x_{k2}^i, \dots, x_{kM}^i)$.

When β is used to compute the association measure between NNV's, it is usually assumed that $\beta_1 = \beta_2 = \dots = \beta_N$. This assumption makes it possible to compare the components of the distance vector $\beta(X, Y) = (e_1, e_2, \dots, e_N)^T$.

Since this research deals with association measures between data items, the discussion about the association measures between NNV's is closed at this point. The association measure between X_i and X_j when they are represented by x_w^i and x_w^j should be computed in the same way as when they were represented by x^i and x^j .

4.3 Association Measure Between Data Items:

A set of data items, D , is given by its three-dimensional (N-block) matrix

$$\underline{D}_w = [D_w^1 \vdots D_w^2 \vdots \dots \vdots D_w^N]$$

or

$$\underline{D} = [D^1 \vdots D^2 \vdots \dots \vdots D^N].$$

The problem is to find some sort of association measure between data items. In this section a few M-dimensional distance functions are discussed. These distance functions as well as many others can be used as measures between data items with matrix representation. The algorithm for working with an M-dimensional distance function is set up so that it is independent of the type of the distance function. Therefore, if later on during the practical use of the measures given in this section, one finds a more effective and/or practical measure he has only to change the formula for calculating the distance matrix (to be defined later in this chapter).

Definition 4.4: Let $X = (x_1, x_2, \dots, x_n)^T$ and $Y = (y_1, y_2, \dots, y_n)^T$ be two column vectors of the set of all column vectors V . The function M_p from $V \times V$ into the set of reals defined by

$$M_p(X, Y) = \left[\sum_{i=1}^n |x_i - y_i|^p \right]^{1/p}$$

is called the Minkowski metric for all $p \geq 1$.

The limit of M_p when p approaches infinity is called the Chebyshev metric and is defined by

$$\lim_{p \rightarrow \infty} M_p(X, Y) = M_\infty(X, Y) = \max_i |x_i - y_i|.$$

For each value of p a different metric could be obtained, but any metric for $p > 2$ is of little practical use for the area of cluster analysis. For $p = 1$ the metric is

$$M_1(X,Y) = \sum_{i=1}^n |x_i - y_i| .$$

For $p = 2$ the metric is

$$M_2(X,Y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} ,$$

M_2 is the familiar Euclidian distance function.

The set of all points for which $M_p = 1$ is called the unit sphere of the metric M_p . The unit spheres for M_1 , M_2 and M_∞ for a two-dimensional and three-dimensional space are given in Figure 4-1.

Let $\gamma_1 = \gamma_2 = \dots = \gamma_n = M_p$. The association measure between the elements of the set of data items, D , is given by

$$\Gamma(D_w^k, D_w^s) = (e_1, e_2, \dots, e_n)^T,$$

where

$$e_i = \left[\sum_{j=1}^M |f_{ij}^k - f_{ij}^s|^p \right]^{1/p} .$$

Recall that $D_w^k = [p_{ij}^k w_{ij}]$, $i=1,2,\dots,n$, $j=1,2,\dots,M$, $k=1,2,\dots,N$, p_{ij}^k is the j^{th} component of the augmented distribution vector of the i^{th} variable for the k^{th} data item, and w_{ij} is the j^{th} component of the weight vector of the i^{th} variable. For simplicity, it is assumed that $p_{ij}^k w_{ij} = f_{ij}^k$. The special case of M_p , M_2 , is used for the purpose of the association measure between data items D_w^k and D_w^s . In this case $M_2(D_w^k, D_w^s)$ is a dissimilarity vector between D_w^k and D_w^s and it is given by $M_2(D_w^k, D_w^s) = (e_1, e_2, \dots, e_n)^T$, where

$$e_i = \left[\sum_{j=1}^M |f_{ij}^k - f_{ij}^s|^2 \right]^{1/2}.$$

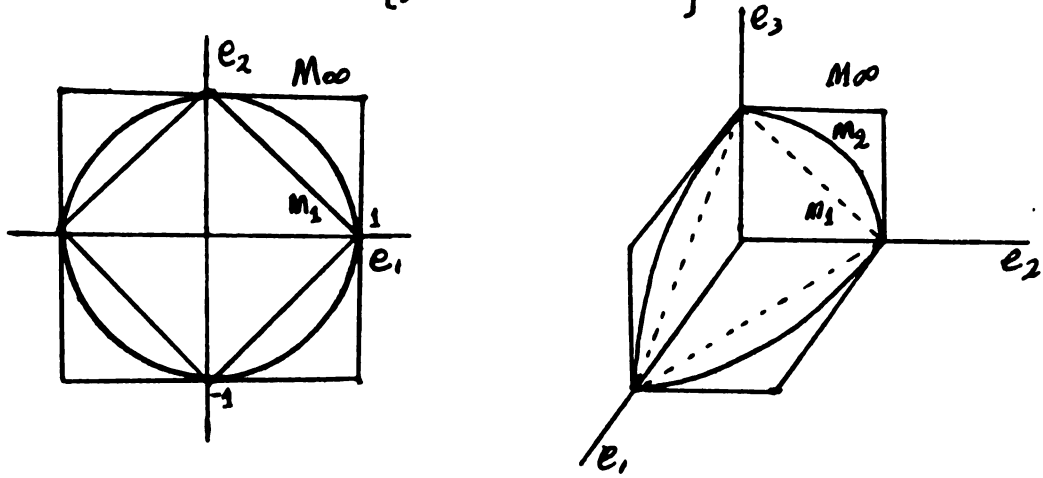


Figure 4-1. Unit spheres of M_1, M_2 and M_∞ .

Note : When a set of data items is defined by both NNV's and numeric variables the magnitude of each of the numeric variable should be normalized so that it can be compared to a NNV. One appropriate factor to normalize a numeric variable can be defined by

$$f_X = 1/M_X,$$

where M_X is the maximum of all N values of the variable X corresponding to the N different data items $D_1, D_2, \dots, D_N$. Hereafter, it is assumed that every numeric variable which is involved in definition of a data item is normalized by some factor f_X .

Definition 4.5: A $p \times q$ matrix A is called a blockwise symmetric matrix if there exists an $r \times r$ block partition $T = [T_{ij}]$ on A such that $T_{ij} = T_{ji}^T$ for $i=1, 2, \dots, r$ and $j=1, 2, \dots, r$. Each T_{ij} is a submatrix of the matrix A .

1

Definition 4.6: Let an M-dimensional distance function

Γ be defined on a set of data items. Let n be the number of variables and D_W^i be the matrix representing a data item D_i ; $\Gamma(D_W^k, D_W^s) = (d_{s1}^k, d_{s2}^k, \dots, d_{sM}^k) = (d_{k1}^s, d_{k2}^s, \dots, d_{kM}^s) = \Gamma(D_W^s, D_W^k)$, where $d_{sj}^k = \gamma_j(f_{.j}^k, f_{.j}^s)$, $f_{.j}^k$ is the j^{th} column of the matrix D_W^k , and γ_j is the j^{th} distance function of Γ . Since there are N data items, if one calculates Γ for all possible pairs of data items the result is an $N \times (N \times M)$ matrix Δ ,

$$\Delta = \begin{bmatrix} d_{11}^1 & d_{12}^1 & \dots & d_{1M}^1 & | & d_{11}^2 & d_{12}^2 & \dots & d_{1M}^2 & | & \dots & | & d_{11}^N & d_{12}^N & \dots & d_{1M}^N \\ d_{21}^1 & d_{22}^1 & \dots & d_{2M}^1 & | & d_{21}^2 & d_{22}^2 & \dots & d_{2M}^2 & | & \dots & | & d_{21}^N & d_{22}^N & \dots & d_{2M}^N \\ \vdots & \vdots & \ddots & \vdots & | & \vdots & \vdots & \ddots & \vdots & | & \ddots & | & \vdots & \vdots & \ddots & \vdots \\ d_{N1}^1 & d_{N2}^1 & \dots & d_{NM}^1 & | & d_{N1}^2 & d_{N2}^2 & \dots & d_{NM}^2 & | & \dots & | & d_{N1}^N & d_{N2}^N & \dots & d_{NM}^N \end{bmatrix}.$$

The matrix Δ is called an N-block distance matrix.

Any N-block distance matrix is a blockwise symmetric matrix. To explain definitions 4.5 and 4.6 one might consider the $\underline{D}_W$ of Example 3.3.

$$\underline{D}_W = \left[\begin{bmatrix} 1/30 & 2/30 & 3/30 & 0 \\ 1/20 & 0 & 0 & 4/20 \end{bmatrix} ; \begin{bmatrix} 1/20 & 2/20 & 0 & 0 \\ 1/30 & 2/30 & 0 & 4/30 \end{bmatrix} ; \begin{bmatrix} 1/40 & 4/40 & 3/40 & 0 \\ 1/40 & 2/40 & 3/40 & 4/40 \end{bmatrix} \right].$$

Using $\gamma_1 = \gamma_2 = \gamma_3 = M_2$ one has

$$\Gamma(D_W^1, D_W^2) = (.024 \ .047 \ .1 \ .066) = \Gamma(D_W^2, D_W^1),$$

$$\Gamma(D_W^1, D_W^3) = (.026 \ .066 \ .078 \ .10) = \Gamma(D_W^3, D_W^1),$$

$$\Gamma(D_W^2, D_W^3) = (.026 \ .017 \ .105 \ .033) = \Gamma(D_W^3, D_W^2),$$

and

$$\Delta = \begin{bmatrix} 0 & 0 & 0 & 0 & .024 & .047 & .10 & .066 & .026 & .066 & .078 & .10 \\ .024 & .047 & .10 & .066 & 0 & 0 & 0 & 0 & .026 & .017 & .105 & .033 \\ .026 & .066 & .078 & .10 & .026 & .017 & .105 & .033 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

As one can see Δ is a blockwise symmetric matrix. Each block is a 4-dimensional vector.

If the distance functions (γ_j 's) are all the same then the components of each row of the matrix Δ within each block are comparable. Let d_{jk}^i be the k^{th} component of the i^{th} block and j^{th} row of Δ . Then d_{jk}^i is the association measure between the data items D_i and D_j based on the k^{th} elements of the ranges of the NNV's. In case of $\gamma_1 = \gamma_2 = \dots = \gamma_M = M_2$ the d_{jk}^i 's are the dissimilarity measures between D_i and D_j . The matrix Δ is a general case of a dissimilarity matrix. If it happens that $M=1$ then Δ will be the same $N \times N$ dissimilarity matrix as would occur by using the SNR method. The problem of storage and computation which was discussed in chapter three will be apparent here by looking at Δ .

Definition 4.6 could be modified for the n -dimensional function β . In this case the form of Δ_β is an $(n \times N) \times N$ matrix. The following is the transposed form of Δ_β .

$$\Delta_\beta^T = \begin{bmatrix} b_{11}^1 & b_{12}^1 & \dots & b_{1n}^1 & b_{11}^2 & b_{12}^2 & \dots & b_{1n}^2 & \dots & b_{11}^N & b_{12}^N & \dots & b_{1n}^N \\ b_{21}^1 & b_{22}^1 & \dots & b_{2n}^1 & b_{21}^2 & b_{22}^2 & \dots & b_{2n}^2 & \dots & b_{21}^N & b_{22}^N & \dots & b_{2n}^N \\ \vdots & \vdots & \dots & \vdots & \vdots & \vdots & \dots & \vdots & \dots & \vdots & \vdots & \dots & \vdots \\ \vdots & \vdots & \dots & \vdots & \vdots & \vdots & \dots & \vdots & \dots & \vdots & \vdots & \dots & \vdots \\ b_{N1}^1 & b_{N2}^1 & \dots & b_{Nn}^1 & b_{N1}^2 & b_{N2}^2 & \dots & b_{Nn}^2 & \dots & b_{N1}^N & b_{N2}^N & \dots & b_{Nn}^N \end{bmatrix}.$$

As one can see Δ_j is also an N-block distance matrix. Each block of Δ_j is a column vector of dimension n. The element b_{jk}^i is the k^{th} component of the association vector between data items D_i and D_j . In other words, b_{jk}^i is the association measure between D_i and D_j in terms of the k^{th} variable.

In Chapter Five the M-dimensional distance function $\sqrt{}$ is used for computing an N-block dissimilarity matrix for a set of N data items. But one could use the n-dimensional function β or any other appropriate function to compute an N-block dissimilarity measure for a set of N data items. Note that $\sqrt{}$ is not invariant under the permutation of the categories of the NNV's. If one is to use $\sqrt{}$ one should hurris-tically fix the order of the categories of the NNV's.

To summarize this chapter, let D_i and D_j be two data items represented by two $n \times M$ matrices D_w^i and D_w^j , respectively. To find the dissimilarity measure between D_i and D_j , two approaches are given. The first approach is to define (choose) $\gamma_1, \gamma_2, \dots, \gamma_M$ and compute $\sqrt{\gamma(D_w^i, D_w^j)}$. In this case the dissimilarity between D_i and D_j is given by an M-dimensional row vector. The k^{th} component of this vector is the dissimilarity between the k^{th} columns of the matrices D_w^i and D_w^j . The second approach is to choose $\beta_1, \beta_2, \dots, \beta_n$ and compute $\beta(D_w^i, D_w^j)$. In this case the dissimilarity between D_i and D_j is given by a column vector. The k^{th} component of this vector is the dissimilarity measure between the k^{th} rows of the matrices D_w^i and D_w^j . In this case the dissimilarity vector is of dimension n.

CHAPTER V

CLUSTERING PROCEDURES

5.1 Introduction:

In developing the clustering block of the GOC system several changes in the representation of the data items and variables were desirable. The following is a list of some of the major changes, given in Chapters Three and Four.

- a. The representation of a data item changes from an n -dimensional vector to an $n \times M$ matrix;
- b. The similarity (dissimilarity) measure between data items changes from a single number to an M -dimensional vector;
- c. The similarity (dissimilarity) matrix changes from an $N \times N$ matrix to an N -block matrix.

Because of these three major changes the data items defined by NNV's need a new type of clustering procedure. In this chapter two different clustering procedures are extended. One can employ these extended procedures for clustering a set of data items defined by NNV's as well as numeric variables. Since most of the comparisons in this chapter are between vectors of the same dimension, the statement given below will help to clarify the ordering relation between vectors of the same dimension.

Statement: Between every two vectors $V = (v_1, v_2, \dots, v_r)$ and $U = (u_1, u_2, \dots, u_r)$ at least one of the following relations exist:

1. $V = U$ iff $v_i = u_i$ for $i=1,2,\dots,r$;
2. $V \geq U$ iff $v_i \geq u_i$ for $i=1,2,\dots,r$;
3. $V < U$ iff $v_i < u_i$ for $i=1,2,\dots,r$;
4. U and V are not comparable.

5.2 Geometric Representation of D and Δ :

Since D^i is a matrix and Δ is a set of Euclidian distances, a geometric representation of D and Δ would help to clarify the notations used in sections 3.4 and 3.5.

Consider two data items D_i and D_j . One has $D_w^i = [f_{kr}^i]$ and $D_w^j = [f_{kr}^j]$, $k=1,2,\dots,n$, $r=1,2,\dots,M$. The distance between D_i and D_j is given by

$$\rho(D_w^i, D_w^j) = (d_{j1}^i, d_{j2}^i, \dots, d_{jM}^i) = (d_{i1}^j, d_{i2}^j, \dots, d_{iM}^j),$$

where $d_{jr}^i = d_{ir}^j$ for $r=1,2,\dots,M$ and

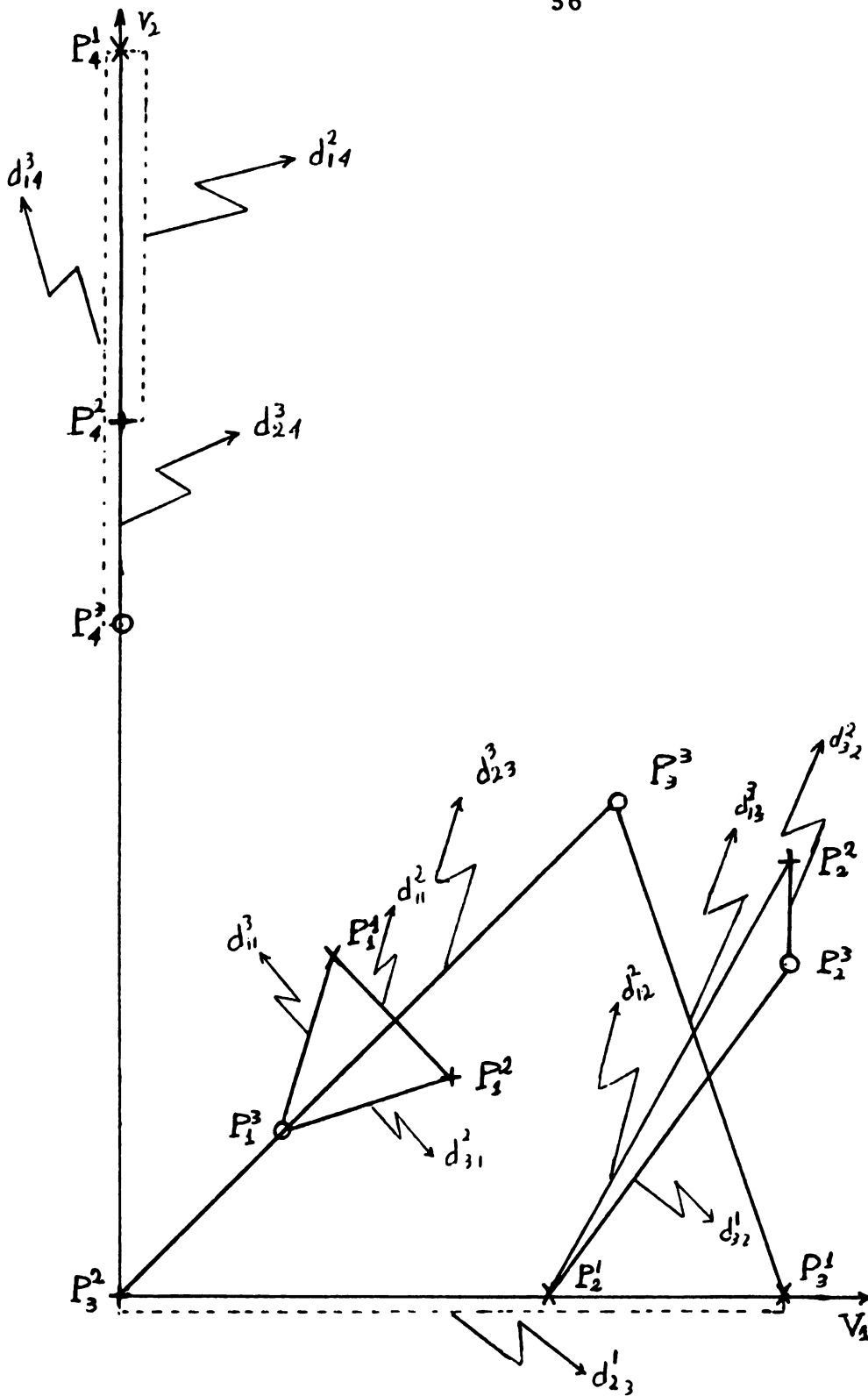
$$d_{jr}^i = \left[\sum_{k=1}^n (d_{kr}^i - d_{kr}^j)^2 \right]^{1/2}.$$

Each data item D_i could be represented by M points in an n -dimensional space. Call these M points $P_1^i, P_2^i, \dots, P_M^i$. The M components of the vector $\rho(D_w^i, D_w^j)$, the j^{th} block of the i^{th} row of Δ , are the Euclidian distances between points P_k^i and P_k^j for $k=1,2,\dots,M$.

Figure 5.1 is the geometric representation of D and Δ of Example 3.3. There were three data items of two variables and M was 4. Therefore, each data item is represented by four points in a 2-dimensional space. In Figure 5.1 a set of four X's represents D_1 , a set of four +'s represents D_2 , and a set of four 0's represents D_3 . The lines between corresponding points are the elements of Δ , as they are shown by arrows.

5.2.1 Hypersphere Representation of D_i :

Let $P_1^i, P_2^i, \dots, P_M^i$ be M points in an n -dimensional space. Let $D_w^i = [f_{jk}^i]$ be the matrix representation of D_i . One could



$$\begin{aligned}
 P_1^1 &= (1/30, 1/20) , P_2^1 = (2/30, 0) , P_3^1 = (1/10, 0) , P_4^1 = (0, 2/10) \\
 P_1^2 &= (1/20, 1/30) , P_2^2 = (1/10, 2/30) , P_3^2 = (0, 0) , P_4^2 = (0, 4/30) \\
 P_1^3 &= (1/40, 1/40) , P_2^3 = (1/10, 1/20) , P_3^3 = (3/40, 3/40) , P_4^3 = (0, 1/10) .
 \end{aligned}$$

Figure 5-1.

Geometric Representation of D and Δ of Example 3.3.

have $P_k^i = (f_{1k}^i, f_{2k}^i, \dots, f_{nk}^i)$ for $i=1,2,\dots,N$ and $k=1,2,\dots,M$.

As one can see P_k^i is a point in an n -dimensional space.

Define $\mu_j^i = 1/M \sum_{k=1}^M f_{jk}^i$, then the vector

$$\mu^i = (\mu_1^i, \mu_2^i, \dots, \mu_n^i)$$

is a row mean vector of the matrix D_w^i . One can represent the data item D_i by the smallest hypersphere S_i containing all the points P_k^i 's with center at μ^i . Figure 5-2 is the hypersphere representation of the data items in Example 3.3. Hypersphere (S_i, μ^i) is called the hypersphere representation of the data item D_i .

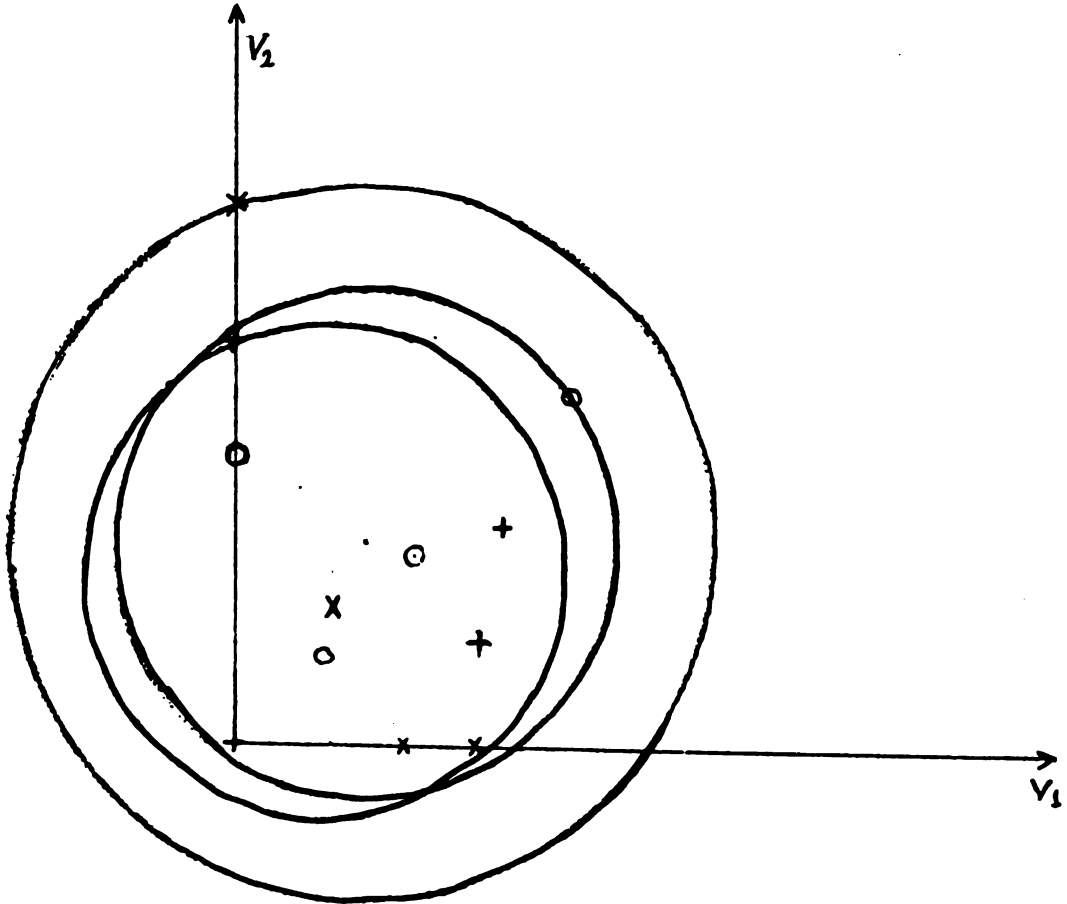


Figure 5-2.

Hypersphere representation of the data items of Example 3.3.

5.3 Clustering the Set D by Transforming Δ to an NxN Dissimilarity Matrix:

The problem of transforming Δ to an NxN dissimilarity matrix is approached in two different ways. The first approach is by means of a threshold vector, T , and the second approach is without use of a threshold vector.

A. Using a Threshold Vector: In this approach it is assumed that a threshold vector $T = (t_1, t_2, \dots, t_M)$ is given. Two data items D_i and D_j are said to be similar if all components of the vector $\cap(D_w^i, D_w^j)$ are less than or equal to the corresponding components of the given threshold vector T . That is, $\cap(D_w^i, D_w^j) \leq T$. As long as it is known that $d_{jk}^i \leq t_k$ or $d_{jk}^i > t_k$ the magnitude of the d_{jk}^i is not important to the procedure of clustering. Considering this fact, it is possible to convert Δ to a binary matrix $\Delta_b = [b_{jk}^i]$ as follows:

$$b_{jk}^i = \begin{cases} 0 & \text{if } d_{jk}^i \leq t_k \\ 1 & \text{if } d_{jk}^i > t_k \end{cases}$$

for $i=1, 2, \dots, N$, $j=1, 2, \dots, N$ and $k=1, 2, \dots, M$.

To transform Δ_b to an NxN binary matrix D_b , one way is to define $D_b = [b_{ij}]$ as follows:

$$b_{ij} = \begin{cases} 1 & \text{if } \sum_{k=1}^M b_{jk}^i > M/2 \\ 0 & \text{if } \sum_{k=1}^M b_{jk}^i \leq M/2 \end{cases}$$

This means that if the number of 1's in the j^{th} block of the i^{th} row of Δ_b is greater than the number of 0's in the j^{th} block of the i^{th} row of Δ_b then $b_{ij} = 1$. Otherwise $b_{ij} = 0$.

By this method of transforming Δ_b to D_b , two data items are called similar if they are similar in more than half of the categories. An alternative approach to transform Δ_b to an $N \times N$ binary matrix D_b which is more reasonable, but idealistic, is as follows:

$$b_{ij} = \begin{cases} 1 & \text{if there exist one or more "1" in the } j^{\text{th}} \text{ block of the } i^{\text{th}} \text{ row of } \Delta_b, \\ 0 & \text{if otherwise.} \end{cases}$$

In this alternative two data items are said to be similar if they are similar in all the cases.

B. Without Using a Threshold: In this approach it is assumed there is no threshold vector given. To transform Δ to an $N \times N$ dissimilarity matrix one could define $\mathcal{A} = [a_{ij}]$ as below:

$$a_{ij} = \sum_{k=1}^M d_{jk}^i$$

for $i=1,2,\dots,N$ and $j=1,2,\dots,N$. Define $\mathcal{A}' = [a'_{ij}]$, where

$a'_{ij} = 1/M a_{ij}$. The matrix $\mathcal{A}'$ is called the average dissimilarity matrix. It is an $N \times N$ symmetric matrix. The element a'_{ij} is the average dissimilarity measure between data items D_i and D_j .

For example, the dissimilarity matrix corresponding to the Example 3.3 is

$$\Delta = \begin{bmatrix} 0 & 0 & 0 & 0 & .024 & .047 & .10 & .066 & .026 & .066 & .078 & .10 \\ .024 & .047 & .10 & .066 & 0 & 0 & 0 & 0 & .026 & .017 & .105 & .033 \\ .026 & .066 & .078 & .10 & .026 & .017 & .105 & .033 & 0 & 0 & 0 & 0 \\ .026 & .066 & .078 & .10 & .026 & .017 & .105 & .033 & 0 & 0 & 0 & 0 \end{bmatrix},$$

and the average dissimilarity matrix corresponding to this Δ is

$$\mathcal{A}' = 1/4 \Delta = 1/4 \begin{bmatrix} 0 & .247 & .270 \\ .247 & 0 & .181 \\ .270 & .181 & 0 \end{bmatrix}.$$

One can use A or A' as the dissimilarity matrix for the set of data items D and cluster the set D based on the matrix A or A' . There are a number of methods available for clustering a set of data items based on an $N \times N$ dissimilarity matrix. (See section 1.2 for references.) Also, the algorithm given in the next section could be used to cluster any set of data items which is represented by an $N \times N$ dissimilarity matrix.

Note that some clustering algorithms require recalculation of the dissimilarity matrix in each iteration of the algorithm. If one uses such an algorithm for clustering purposes he must note that to recalculate the $N \times N$ matrix A it is necessary to recalculate Δ .

The problem with using the matrix $A(A')$ as the dissimilarity matrix is that there is a chance of generating invalid clusters in some special cases. This is because of summing (or averaging) the dissimilarity measures over the categories of the NNV's. Consider a particular case when a set of data items consists of four data items and is represented by a 4-block dissimilarity matrix

$$\Delta = \begin{bmatrix} 0 & 0 & 0 & 0 & | & 0 & .25 & 0 & .75 & | & .25 & 0 & .75 & 0 & | & .25 & .26 & .27 & .30 \\ & & & & | & 0 & 0 & 0 & 0 & | & 0 & .75 & .25 & 0 & | & .29 & .30 & .30 & .31 \\ & & & & | & & & & & | & 0 & 0 & 0 & 0 & | & .26 & .25 & .28 & .29 \\ & & & & | & & & & & | & & & & & | & 0 & 0 & 0 & 0 \end{bmatrix}.$$

The average matrix corresponding to Δ is

$$A' = \begin{bmatrix} 0 & .25 & .25 & .27 \\ & 0 & .25 & .30 \\ & & 0 & .27 \\ & & & 0 \end{bmatrix}.$$

By looking at Δ one can see the dissimilarity relation between the pair of data items (D_1, D_2) is totally different from the one between the pairs (D_1, D_2) and (D_2, D_3) . But the average matrix A' says the dissimilarity relations between the pairs (D_1, D_2) , (D_1, D_3) and (D_2, D_3) are exactly the same. In other words, it is very likely for a clustering algorithm to put D_1, D_2 and D_3 into one group as three similar data items (using A or A' as dissimilarity matrix). Because of this problem, the following necessary condition which is only based on some experiments is suggested for the use of an average dissimilarity matrix (or the matrix of the sum of the dissimilarity measures) for clustering a set of data items.

Necessary Condition: Calculate the standard deviation of each $1 \times M$ submatrix of Δ . There are $N(N-1)/2$ standard deviations. Let σ_{ij} be the standard deviation of the j^{th} block in the i^{th} row and μ_{ij} the mean of this block.

$$\mu_{ij} = 1/M \sum_{k=1}^M d_{jk}^i$$

$$\sigma_{ij} = \sqrt{1/M \sum_{k=1}^M (d_{jk}^i)^2 - \mu_{ij}^2}$$

$$\lambda_{ij} = 1/M \max_k |d_{jk}^i - \mu_{ij}|,$$

for $i=1, 2, \dots, N$ and $j=1, 2, \dots, N$.

Do not use the matrix A or A' for the purpose of clustering if there exist one or more σ_{ij} such that $\sigma_{ij} > \lambda_{ij}$ for $i=1, 2, \dots, N$ and $j=1, 2, \dots, N$. If it should happen that for some cases one has $\sigma_{ij} < \lambda_{ij}$ for all i 's and j 's then one might use A or A' as a dissimilarity matrix for clustering purposes.

• 1
2
3
4

5

6
7
8

• 9
10
11

12
13

14
15

16
17
18

19
20

5.4 General Centroid Method (GCM):

In the general centroid method it is assumed that the dissimilarity between each pair of data items is given by an M -dimensional row vector. The i^{th} component of this vector is the dissimilarity between the pair of data items associated with the i^{th} categories of the NNV's. It is also assumed that each data item D_i is represented by an $n \times M$ matrix D_w^i . The dissimilarity matrix of these data items is an N -block matrix. Note that if there is no weight vector corresponding to the range of a particular NNV then the weight vector for this NNV is assumed to be a vector with all components equal to 1.

In this method it is assumed that either a fixed threshold vector $T = (t_1, t_2, \dots, t_M)$ is given for the clustering purposes or a vector T will be calculated in each iteration of the algorithm.

If there is no fixed threshold vector given then one can define an appropriate formula to calculate a threshold vector T when it is necessary to be calculated. The following is a list of some appropriate formulas for computing T :

1. Let $\Delta_r = [d_{jk}^i]$, $i=1,2,\dots,N_r$, $j=1,2,\dots,N_r$ be the N_r -block dissimilarity matrix at the r^{th} iteration of the algorithm. Define

$$t_k = \frac{2}{N_r(N_r-1)} \sum_{i=1}^{N_r} \sum_{j>i}^{N_r} d_{jk}^i, \quad k=1,2,\dots,M.$$

Then $T=(t_1, t_2, \dots, t_M)$ is a threshold vector computed for the r^{th} iteration of the algorithm.

2. Define t_k for $k=1,2,\dots,M$ to be the median of the numbers:

$d_{2k}^1, d_{3k}^1, \dots, d_{N_r k}^1, d_{3k}^2, d_{4k}^2, \dots, d_{N_r k}^2, \dots, d_{i+1k}^i, \dots, d_{N_r k}^i, \dots, d_{N_r k}^{N_r-1}$.

The vector $T=(t_1, t_2, \dots, t_M)$ is a threshold vector computed for the r^{th} iteration of the algorithm.

3. For $0 < q < 1$ define t_k to be the smallest number such that $q*(N_r(N_r-1))/2$ elements of the numbers

$d_{2k}^1, d_{3k}^1, \dots, d_{N_r k}^1, d_{3k}^2, d_{4k}^2, \dots, d_{N_r k}^2, \dots, d_{i+1k}^i, \dots, d_{N_r k}^i, \dots, d_{N_r k}^{N_r-1}$

are less than or equal to t_k and $(1-q)(N_r(N_r-1))/2$ of them are greater than t_k . A special case of this could be for $q = 1$. In

this case t_k would be defined as $t_k = \text{Maximum}_{\substack{1 \leq i < N_r \\ j > i}} (d_{jk}^i)$

The initial groups of the data items consist of N groups, one data item per group. Assume G_i and G_j are two groups of data items. These groups are said to be similar if and only if

$$\rho(G_w^i, G_w^j) < T.$$

If during the operation of the algorithm it was decided to combine two groups G_i and G_j into one group, G_k , then the matrix representation of this new group is given by

$$G_w^k = (G_w^i + G_w^j)/2,$$

where G_w^i and G_w^j are the matrix representations of the groups G_i and G_j , respectively. At the end of each iteration of the algorithm, if there are some new groups generated, then the following items must be updated and/or recalculated:

1. the total number of groups;
2. the matrices representing the new groups;
3. the dissimilarity N_r -block matrix.

5.4.1 General Centroid Algorithm (GCA):

The algorithm for the GCM based on an N-block dissimilarity matrix consists of seven steps. Note that a formula for computing a threshold vector for the algorithm and a formula for computing the dissimilarity measure between data items are given as a part of the input to the algorithm. The steps of the algorithm are as follows:

Step 1-- Compute the dissimilarity N-block matrix Δ .
Compute the threshold vector T if it is not given as input.

Step 2-- If there is no threshold given go to step 3.
Otherwise consider those pairs of the total $\binom{N}{2}$ pairs of the groups of data items which satisfy the condition $\sqrt{'}(G_w^i, G_w^j) \leq T$, where G_i and G_j are two groups of data items and G_w^i and G_w^j are their matrix representations. Let

$Q_1 = (G_{i_1}, G_{j_1})$, $Q_2 = (G_{i_2}, G_{j_2})$, ..., $Q_c = (G_{i_c}, G_{j_c})$
be those pairs which satisfy the above condition. Every one of these pairs is a candidate pair for being combined into one group. If $c = 0$ then go to step 7, otherwise go to step 3.

Step 3-- Assign a number q_k to each pair $Q_k = (G_{i_k}, G_{j_k})$ as follows:

$$q_k = \sum_{r=1}^M d_{jk_r}^{i_k},$$

where $d_{jk_r}^{i_k}$ is the r^{th} component of the dissimilarity vector between groups G_{i_k} and G_{j_k} . It could happen, although very seldom, that $q_k = q_{k'}$, for some $k \neq k'$. If this should happen replace q_k by q'_k as follow:

$$q'_k = \begin{cases} q_k + \frac{1}{10^{\nu+1}} & \text{if } S_{i_k} + S_{j_k} > S_{i_{k'}} + S_{j_{k'}}, \\ q_k - \frac{1}{10^{\nu+1}} & \text{if } S_{i_k} + S_{j_k} < S_{i_{k'}} + S_{j_{k'}}, \end{cases},$$

where S_r is the sum of the elements in the r^{th} row of the matrix Δ , and ν is the maximum number of digits to the right of the decimal point in the q_k 's after rounding them. If $S_{i_k} + S_{j_k}$ is equal to $S_{i_k} + S_{j_k}$, then one should reconsider the matrices representing the data items in the pairs Q_k and $Q_{k'}$.

Step 4-- Rank the pairs $Q_1, Q_2, \dots, Q_c$ according to their weights, q_k 's. Sort these pairs according to their ranks.

Step 5-- Mark the pair with the smallest weight (or rank). If $Q_k = (G_{i_k}, G_{j_k})$ is a marked pair, then G_{i_k} and G_{j_k} should be combined into one group. If no threshold vector is given go to step 6. Otherwise delete all pairs which have either G_{i_k} or G_{j_k} as one of their elements from the list of sorted pairs. Repeat step 5 until every remainder pair in the sorted list is marked.

Step 6-- Go to step seven if either all data items are assigned into one group or if the number of groups is not greater than a previously defined number, provided such a number is given. Otherwise, calculate the matrices for the new groups, set the new number of groups to N , and go back to step 1.

Step 7-- Print all the information about the generated groups (clusters) and stop the algorithm.

Example 5.1: Let $D = \{D_1, D_2, \dots, D_6\}$ be a set of six data items to be clustered. The initial groups are $G_1 = \{D_1\}$, $G_2 = \{D_2\}$, $G_3 = \{D_3\}$, $G_4 = \{D_4\}$, $G_5 = \{D_5\}$, $G_6 = \{D_6\}$ and the matrices representing these groups are:

$$G_W^1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1/3 & 2/3 & 0 & 0 \\ 1/3 & 0 & 1/3 & 1/3 \end{bmatrix}, G_W^2 = \begin{bmatrix} 2/3 & 0 & 0 & 0 \\ 1/4 & 3/4 & 0 & 0 \\ 1/4 & 1/8 & 1/4 & 1/4 \end{bmatrix}, G_W^3 = \begin{bmatrix} 5/6 & 0 & 0 & 0 \\ 1/3 & 1/3 & 1/6 & 0 \\ 3/4 & 0 & 1/4 & 0 \end{bmatrix}$$

$$G_W^4 = \begin{bmatrix} 1/5 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1/2 & 0 & 1/2 & 0 \end{bmatrix}, G_W^5 = \begin{bmatrix} 3/5 & 0 & 0 & 0 \\ 0 & 1/2 & 1/4 & 0 \\ 1/2 & 0 & 0 & 1/2 \end{bmatrix}, G_W^6 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1/2 & 1/2 & 0 & 0 \\ 1/2 & 1/4 & 0 & 0 \end{bmatrix}$$

The distance functions $\chi_1, \chi_2, \dots, \chi_M$ all are the same and they are the Euclidian distances in an n-dimensional space. Since Δ is a blockwise symmetric matrix, one need only store $N(N-1)/2$ blocks of Δ . In this example N is 6 and M is 4, therefore, Δ can be stored in an array of two dimensions. The first dimension is equal to $N(N-1)/2 = 15$ and the second dimension is equal to $M=4$. Call this array $\text{DELTA}(15,4)$.

Applying step 1 of the GCA: the initial dissimilarity matrix is computed and stored in $\text{DELTA}(I,J)$ as follow:

$i \backslash j$	1	2	3	4
1	.35	.15	.08	.08
2	.45	.33	.19	.33
3	.88	.33	.17	.33
4	.55	.17	.42	.17
5	.24	.30	.33	.33
6	.57	.44	.17	.25
7	.59	.28	.25	.25
8	.36	.28	.35	.25
9	.49	.28	.25	.25
10	.76	.66	.30	.00

$I \backslash J$	1	2	3	4
11	.48	.17	.26	.50
12	.33	.30	.30	.00
13	.40	.50	.56	.50
14	.94	.56	.50	.00
15	.64	.25	.25	.50

DELTA(I,J)

Let a fixed threshold vector $T = (.50, .45, .40, .30)$ be given. Applying step 2 of the GCA: $Q_1=(G_1, G_2)$, $Q_2=(G_2, G_6)$, $Q_3=(G_3, G_6)$, and $Q_4=(G_2, G_5)$. Applying step 3 of the GCA: $q_1=.66, q_2=1.27$, $q_3=.93$, and $q_4=1.24$. Applying step 4 of the GCA: Q_1 and Q_2 are marked and Q_2 and Q_4 are deleted from the list. Applying step 6 of the GCA: the matrix representation of the new groups H_1 and H_2 are:

$$H_W^1 = \begin{bmatrix} 5/6 & 0 & 0 & 0 \\ 7/24 & 17/24 & 0 & 0 \\ 7/24 & 1/16 & 7/24 & 7/24 \end{bmatrix} = (G_W^1 + G_W^2)/2,$$

$$H_W^2 = \begin{bmatrix} 11/12 & 0 & 0 & 0 \\ 5/12 & 5/12 & 1/12 & 0 \\ 5/8 & 1/8 & 1/8 & 0 \end{bmatrix} = (G_W^3 + G_W^6)/2.$$

The number of new groups, N , is 4 and the groups are:

$$G_1=H_1=\{D_1, D_2\}, \quad G_2=\{D_3, D_6\}, \quad G_3=\{D_4\}, \quad G_4=\{D_5\}.$$

The second time through the algorithm, the new matrix Δ is given by $\text{DELTA}(I, J)$ as below:

$I \backslash J$	1	2	3	4
1	.37	.30	.19	.29
2	.73	.30	.21	.29
3	.43	.22	.38	.21

$I \backslash J$	1	2	3	4
4	.84	.60	.37	.00
5	.54	.15	.21	.50
6	.40	.50	.56	.50

$\text{DELTA}(I, J)$.

There is no change in the threshold T . The candidate pairs are: $Q_1=(G_1, G_2)$ and $Q_2=(G_1, G_4)$. The marked pair is Q_1 and the matrix representation of this new group is given by:

$$H_W^1 = \begin{bmatrix} 21/24 & 0 & 0 & 0 \\ 17/48 & 27/48 & 1/24 & 0 \\ 11/24 & 3/32 & 5/24 & 7/48 \end{bmatrix}, \quad = (G_1 + G_2)/2.$$

The groups are $G_1 = H_1 = \{D_1, D_2, D_3, D_6\}$, $G_2 = \{D_4\}$ and $G_3 = \{D_5\}$.

The new number of groups is 3. The third time through the algorithm c would be zero and the procedure of clustering has to stop at step 7 of the algorithm. Corresponding to the given threshold vector T the final clusters are the groups generated at the second iteration of the algorithm.

5.4.2 Analysis of the General Centroid Algorithm:

The five important features of an algorithm should be demonstrated for the GCA in order to call it an algorithm.

These five features are as follows:

1. Finiteness: The GCA is a finite procedure and it will stop after $(\frac{N}{2}) - 1$ iterations at most. The maximum number of times that step 2 can be executed before going to step 7 is $N(N-1)/2$. Each time through the algorithm either there is no candidate pair or there is at least one candidate pair. If there is no candidate pair the algorithm will halt. The worst case occurs when every time through step 2 there is only one candidate pair. Each time through step 5 at least one group will be marked. The initial number of groups is $N(N-1)/2$ and each time through step 5 the number of groups reduces at least by one. Therefore, after at most $N(N-1)/2 - 1$ iterations the number of groups is one and the GCA will halt at step 7. Each individual step of the GCA is finite and there is no infinite loop in any of the seven steps of the GCA. The only step which is

repeated in each iteration is step 5. Let there be c pairs included in the sorted list of step 4. The maximum number of times that step 5 might be repeated is c ; and this happens only if all c candidate pairs are disjoint.

2. Definiteness: Step 1 is the calculation of the dissimilarity matrix Δ and the threshold vector T . Step 2 is the comparison of at most $\binom{N}{2}$ vectors of dimension M with the vector T . The result of the comparison is equal, less than, greater than, or not comparable, and one should keep record of those comparisons which result in less than or equal. Step 3 is the calculation of the sum of the dissimilarity measures for each candidate pair of groups. Step 4 is the ranking and sorting of a set of c pairs which have c different weights. Steps 5, 6, and 7 are very well defined and their definiteness is obvious.

3. Input: There are obviously some inputs to the GCA such as, a set of N matrices each one representing a single data item, a threshold vector T (or a formula for calculating T), a formula for calculating the dissimilarity measures between each pair of data items, and N , M , and n .

4. Output: The output of the GCA is a set of M_0 clusters along with all relevant information about each cluster.

5. Effectiveness: All of the operations which are to be performed in the GCA are sufficiently basic that they can be done exactly and in a practical (not mathematical) finite length of time by a person familiar with the operations using pencil and paper. The GCA uses some basic operations

such as summation, subtraction, multiplication, comparison, and square root.

5.4.3 Computational Limit and the Problem of Storage:

Given N and M , the dissimilarity matrix Δ needs $\frac{M}{2} (N(N-1))$ computer words to be stored. A normal size problem might have as many as 400 data items and M usually is between 3 and 10. For $M=5$ and $N=200$ the amount of memory needed to store Δ is 99,500 words which is beyond the capacity of the average size existing computers. Because of this problem the Stored Matrix Approach is used for programming the GCA. The dissimilarity matrix Δ will be stored on disc or on some other storage device and it will be read as needed. The amount of memory needed for programming the GCA consists of two two-dimensional arrays $DATA1(n,M)$ and $DATA2(n,M)$, a one-dimensional array $DISV(M)$, and possibly some other small arrays as they are needed for programming purposes.

The calculation limit of the GCA is very much problem-dependent and is especially dependent on the threshold vector T . For fixed N , n , and M one might change T so that it makes the same problem twice as long. The number of basic computations which are required for the initial iteration of the GCA could be given as below.

Given the Euclidian distance as the dissimilarity measure between data items and a fixed threshold vector T , the number of basic computations needed in step 1 are:

$(N(N-1)/2)[n+(M-1)(n-s)]$ subtractions (s is the number of numeric variables), the same number of multiplications,

$M*(N(N-1))/2$ square roots and a large number of input/output operations. At step 2 there are $M*(N(N-1))/2$ comparisons plus some input/output operations. At step 3 there are $M * c$ summations and perhaps some additional summations when two weights become equal. Step 4 is a sorting assignment for a list of c elements. (there are tables available for computational time of a binary sort for different number of data.) Computation at step 5 is a search through a sorted list. This search might be done as many times as : $2(c-1)+2(c-2)+\dots+2= 2(c-1)(c-2)/2 = (c-1)(c-2)$. Steps 6 and 7 do not require many computations. As mentioned before, the number of computations at the non-initial iterations of the GCA are dependent on the threshold vector T . Any estimation of the computation limits of the GCA at non-initial iterations without considering the effect of T would be a rough guess.

5.4.4 Some Special Cases:

As a special case consider $M=1$, that is, there is no NNV. In other words, consider the case where the dissimilarity matrix is an $N \times N$ matrix. In this case T is a threshold number. Step 3 of the GCA is not needed for this special case except for tie breaking schemes. The other steps of the GCA stay the same for this special case.

As another special case consider the situation where neither a threshold vector is given nor a formula for finding it. In this case step 2 of the GCA is redundant, that is, all $N(N-1)/2$ pairs are candidate pairs. For this special case,

in each iteration of the algorithm there are only two groups and always two groups to be combined into one group. Therefore, in each iteration of the algorithm the number of clusters reduces by one.

A third special case occurs when the dissimilarity matrix is an $N \times N$ matrix and there is no threshold vector given. The GCA as modified for the above two special cases can be used to cluster this third special case. Let D be a set of data items and A or A' as defined in section 5.3.B. The GCA can be used to cluster the set D with the dissimilarity matrix A . As an example, consider the data items of Example 5.1. The matrix A corresponding to these data items is as follows: (Only the elements of the upper triangle are printed.)

$$A = \begin{bmatrix} 0 & .66 & 1.30 & 1.72 & 1.30 & 1.20 \\ & 0 & 1.59 & 1.37 & 1.24 & 1.27 \\ & & 0 & 1.73 & 1.41 & .93 \\ & & & 0 & 1.96 & 2.00 \\ & & & & 0 & 1.64 \\ & & & & & 0 \end{bmatrix}.$$

At step 5 the pair (D_1, D_2) is marked and at step 6 the number of new groups is 5. The new matrix A corresponding to these five groups $G_1 = \{D_1, D_2\}$, $G_2 = \{D_3\}$, $G_3 = \{D_4\}$, $G_4 = \{D_5\}$ and $G_5 = \{D_6\}$ is:

$$A = \begin{bmatrix} 0 & 1.39 & 1.17 & 1.24 & 1.19 \\ & 0 & 1.73 & 1.41 & .93 \\ & & 0 & 1.96 & 2.00 \\ & & & 0 & 1.64 \\ & & & & 0 \end{bmatrix}$$

At this time the pair (G_3, G_6) is marked. The number of new groups is 4 and the new matrix A corresponding to these four groups

$G_1 = \{D_1, D_2\}$, $G_2 = \{D_3, D_6\}$, $G_3 = \{D_4\}$ and $G_4 = \{D_5\}$ is:

$$A = \begin{bmatrix} 0 & 1.15 & 1.47 & 1.24 \\ & 0 & 1.72 & 1.40 \\ & & 0 & 1.96 \\ & & & 0 \end{bmatrix}.$$

At this time the pair (G_1, G_2) is marked. The number of new groups

is 3 and the new matrix A for these groups $G_1 = \{G_1, G_2\} =$

$\{D_1, D_2, D_3, D_6\}$, $G_2 = \{D_4\}$ and $G_3 = \{D_5\}$ is :

$$A = \begin{bmatrix} 0 & 1.65 & 1.21 \\ & 0 & 1.96 \\ & & 0 \end{bmatrix}.$$

At this time the number of groups is 2 and the groups are:

$G_1 = \{D_1, D_2, D_3, D_5, D_6\}$ and $G_2 = \{D_4\}$.

Next time there will be only one group and the algorithm will stop at step 7.

The remainder of this chapter is a different approach to the problem of clustering a set of N data item with an N-block dissimilarity matrix Δ . In the GCM the dissimilarity matrix Δ is to be recalculated every time repeating step 1 of the GCA. The approach discussed in the following sections has the advantage of calculating Δ only once for the whole clustering procedure.

5.5 Graph Representation of D and Δ :

In this section a new type of graph is introduced and some of the properties of this graph are discussed. This new type of graph is used to represent a set of data items D along with its dissimilarity N-block matrix Δ .

Definition 5.1: A graph $G = \langle A \cup B, P \rangle$ is called a bipartite graph [81] if $A \cap B = \emptyset$ and every member of P has one element in A and the other element in B. A and B are called sources of G and the number of elements (points or nodes) in A is called the size of the source A. If every node of A is connected to every node of B then G is called a complete bipartite graph.

Definition 5.2: A graph $G = \langle \bigcup_{i=1}^N S_i, P \rangle$ is called an N-partite graph if $S_i \cap S_j = \emptyset$ and each member of P has one element in S_i and the other element in S_j for $i \neq j$, $i=1,2,\dots,N$, and $j=1,2,\dots,N$.

Definition 5.3: An N-partite graph $G = \langle \bigcup_{i=1}^N S_i, P \rangle$ is called a weighted N-partite graph (or an N-partite weighted graph) if there is a weight corresponding to every edge of the N-partite graph G.

The mean of all weights of the edges of a source is the weight of that source.

Definition 5.4: An N-partite graph $G = \langle \bigcup_{i=1}^N S_i, P \rangle$ is called a uniform N-partite graph if:

1. the number of nodes in every source is the same, that is, $|S_i| = M$, $i=1,2,\dots,N$;
2. each node has exactly N-1 branches (edges); and
3. there are exactly M edges between every two sources.

Proposition 5.1: The number of edges of a uniform N -partite graph with source size M is $M(N(N-1))/2$.

Proof: There are $\binom{N}{2} = N(N-1)/2$ pairs of sources and by (3) in definition 5.4 there are M edges corresponding to each pair of sources. QED.

In Figure 5-3, (a) is a bipartite graph and (b) is a complete bipartite graph. A complete bipartite graph is usually denoted by $K_{r,s}$, where r and s are the size of its two sources. In Figure 5-3, (c) is a 3-partite graph, (d) is a weighted 4-partite graph, (e) is a uniform 4-partite graph and (e') is a uniform weighted 4-partite graph.

Definition 5.5: Let $G = \langle \bigcup_{i=1}^N G_i, P \rangle$ be a uniform weighted N -partite graph and $G_i = \{g_{i1}, g_{i2}, \dots, g_{iM}\}$ and $G_j = \{g_{j1}, g_{j2}, \dots, g_{jM}\}$ be two sources of G . Let $w(g_{ik}, g_{jk}) = w_k$ be the weight of the edge (g_{ik}, g_{jk}) . The weight

$$W(G_i, G_j) = 1/M \sum_{k=1}^M w(g_{ik}, g_{jk})$$

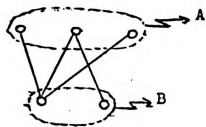
is called the average weight between two sources G_i and G_j .

Definition 5.6: Let $G = \langle \bigcup_{i=1}^N G_i, P \rangle$ be a uniform weighted N -partite graph. Construct a complete weighted graph G_a , as follows:

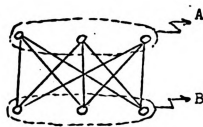
1. The nodes of G_a are the sources of G ;
2. The weight of each edge of G_a is the average weight of its corresponding nodes.

The graph G_a is called an average weighted graph corresponding to the graph G .

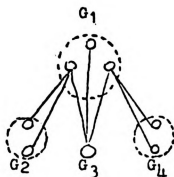
Note that there are $N-1$ average weights corresponding to each source of a uniform N -partite weighted graph G . But the total number of average weights of G is $N(N-1)/2$.



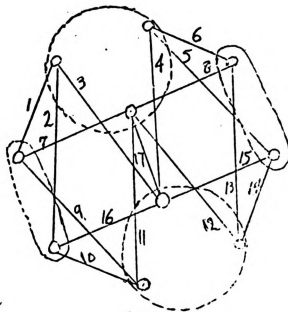
(a)



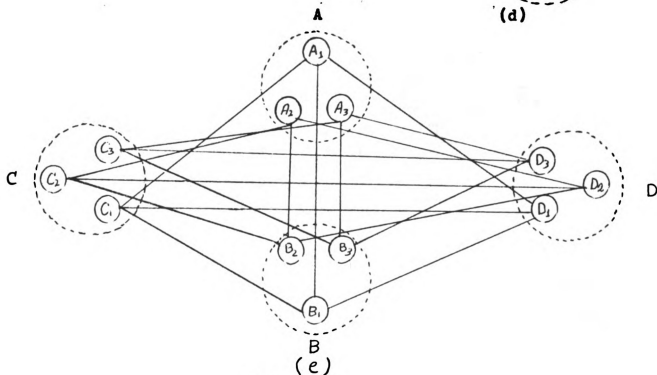
(b)



(c)



(d)



(e)

Figure 5-3.

Bipartite, k-partite, Uniform k-partite, Uniform weighted k-partite and Average Graph.

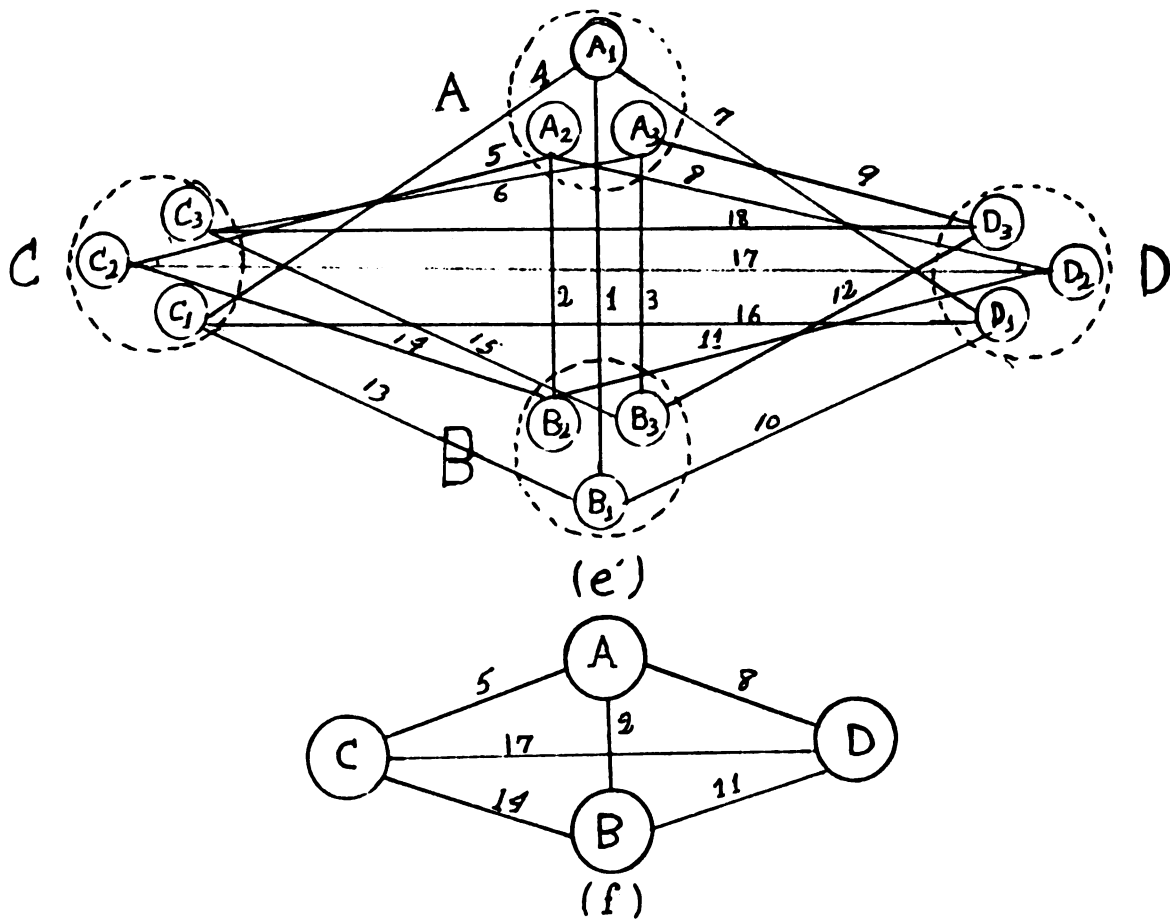


Figure 5-3 (continued)

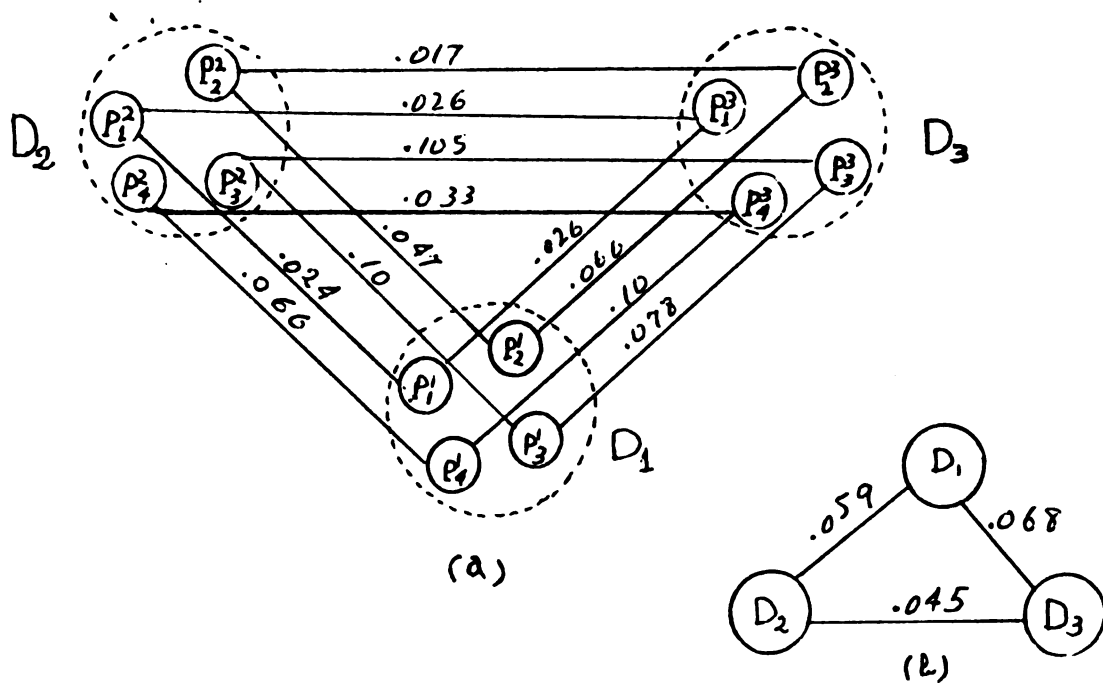


Figure 5-4.

Graph representation of D and Δ .

Figure 5-3 (f) is the average graph of Figure 5-3 (e').

Let $D = \{D_1, D_2, \dots, D_N\}$ be a set of N data items. Let D_i be represented by M points $P_1^i, P_2^i, \dots, P_M^i$ and Δ be the dissimilarity N -block matrix of the set D . Considering D and Δ as defined above, the following theorem can be stated.

Theorem 5.1: Let $G = \langle \bigcup_{i=1}^N D_i, P \rangle$, where
 $P = \{ (P_k^i, P_k^j) \mid i \neq j, i=1,2,\dots,N, j=1,2,\dots,N, k=1,2,\dots,M \}$.

Then G is a uniform N -partite graph.

Proof: By definition of P there are no edges between points in D_i for $i=1,2,\dots,N$. Each point of D_i has only one branch or edge to every one of the sources $D_1, D_2, \dots, D_{i-1}, D_{i+1}, \dots, D_N$. And the fact that every source has exactly M points (nodes) completes the proof of the theorem. QED.

The set of data items D could be represented by the graph G of theorem 5.1. Let d_{jk}^i be the weight of the edge (P_k^i, P_k^j) of the graph G . Then G is a uniform N -partite weighted graph and is the graph representation of both D and Δ . For example, the graph representation of D and Δ of Example 3.3 is given in Figure 5.4 (a). Figure 5-4 (b) is the average graph of Example 3.3.

5.6 k-clustering Based on Graph Representation of D and Δ :

The term "k-cluster" was introduced by Ling [41]. He proposed a method of clustering based on an $N \times N$ dissimilarity matrix. The method was called the k-clustering method. In this section the k-clustering terminology is used for a clustering method which deals with an N -block dissimilarity matrix Δ . This method is based on graphical representations of D and Δ . Most of the terminologies used in this section are taken from Ling

but are defined for the UNWG's.

Let D be a set of N data items and Δ be the N -block dissimilarity matrix corresponding to D . There is a one-to-one correspondence between P of theorem 5.1 and Δ . Therefore, it is possible to replace P by Δ in theorem 5.1. Corresponding to each d_{jk}^i for $i \neq j$ there is one edge between two sources D_i and D_j . Therefore, throughout this section

$$G = \left\langle \bigcup_{i=1}^N D_i, \Delta \right\rangle$$

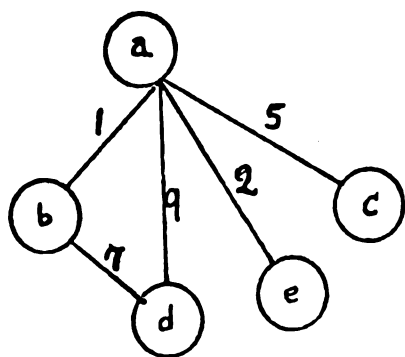
is a uniform N -partite weighted graph representing D and Δ .

Definition 5.7: Let $G_1 = \langle A, P \rangle$ and $G_2 = \langle B, Q \rangle$ be two graphs:

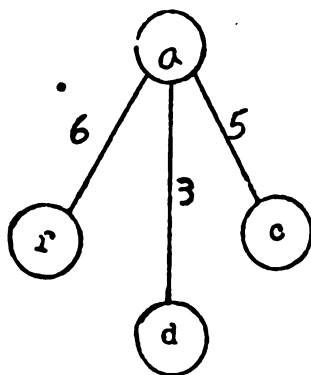
- a. $G_1 = \emptyset \iff A = \emptyset$;
- b. $G_1 \cap G_2 = \langle A \cap B, P \cap Q \rangle$;
- c. $G_1 \cup G_2 = \langle A \cup B, P \cup Q \rangle$;
- d. $H \subset G_1 \iff H = \langle C, R \rangle, C \subset A \text{ and } R \subset P$.

Note that $P \cap Q$ is the set of all edges which have their nodes in $A \cap B$. Similarly, $P \cup Q$ is the union of all edges of both graphs, G_1 and G_2 . In order to extend definition 5.7 to weighted graphs one can define the set of edges of a weighted graph G_1 as follows: (a, b, w) is an edge of a weighted graph G_1 only if (a, b) is an edge of G_1 with a weight equal to w . For example, the weighted graphs of Figure 5-5 could be represented as follows: $G_1 = \langle A, P \rangle$, $A = \{a, b, c, d, e\}$, $P = \{(a, b, 1), (a, c, 5), (a, e, 2), (b, d, 7), (a, d, 9)\}$ and $G_2 = \langle B, Q \rangle$, $B = \{a, c, d, f, g, h\}$, $Q = \{(a, c, 5), (a, d, 3), (a, f, 6), (f, g, 0), (h, g, 10), (c, h, 1), (d, h, 4)\}$.

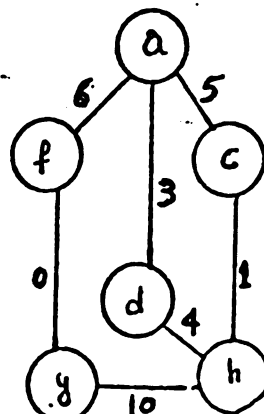
If $G_1 = \langle A, P \rangle$ and $G_2 = \langle B, Q \rangle$ are two weighted graphs then part (c) of definition 5.7 is well defined only if $A \cap B = \emptyset$.



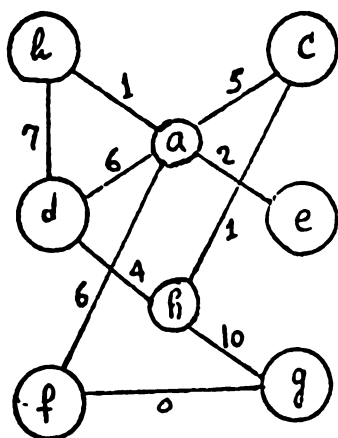
$$G_1 = \langle A, P \rangle$$



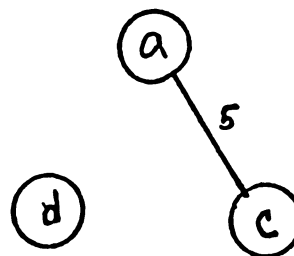
$$H = \langle C, R \rangle \subset G_2$$



$$G_2 = \langle B, Q \rangle$$



$$G_3 = G_1 \cup G_2$$



$$G_4 = G_1 \cap G_2$$

Figure 5-5.

A picture of definition 5.7 for weighted graphs.

If $A \cap B \neq \emptyset$ and (a,b) be an edge in both P and Q but with different weights, say w_1 and w_2 , then the union of these two edges is defined as: $(a,b,w_1) \cup (a,b,w_2) = (a,b, \frac{w_1+w_2}{2})$. For example, the nodes and the edges of $G_3 = G_1 \cup G_2$ and $G_4 = G_1 \cap G_2$ of Figure 5-5 are as follows:

$$A \cup B = \{a, b, c, d, e, f, g, h\};$$

$$P \cup Q = \{(a,b,1), (a,c,5), (a,d,6), (a,e,2), (a,f,6), (b,d,7), (f,g,0), (c,h,1), (d,h,4), (h,g,10)\};$$

$$A \cap B = \{a, c, d\}, \quad P \cap Q = \{(a,c,5)\}.$$

Definition 5.8: Let $A = \{A_1, A_2, \dots, A_r\}$ and $B = \{B_1, B_2, \dots, B_s\}$ be two sets of sources. Let all sources be of the same size, say M . The union and intersection of A and B are defined as below.

$$A_i \oplus B_j = \begin{cases} A_i \cup B_j & \text{if } A_i \cap B_j = \emptyset \\ A_i & \text{if } A_i = B_j \\ \text{Undefined} & \text{otherwise} \end{cases}$$

$$A_i \odot B_j = \begin{cases} A_i & \text{if } A_i = B_j \\ \emptyset & \text{if } A_i \neq B_j \end{cases}$$

$$A \cup B = \bigcup_{i=1}^r \left[\bigcup_{j=1}^s (A_i \oplus B_j) \right]$$

and

$$A \cap B = \bigcup_{i=1}^r \left[\bigcup_{j=1}^s (A_i \odot B_j) \right].$$

Note that if there is one or more undefined elements in $A \cup B$ then $A \cup B$ is not defined. With this definition in mind one can apply definition 5.7 to the UkWG's as well as to the regular graphs.

Definition 5.9: Let G be a UNWG. A uniform k -partite weighted subgraph (UkWSG) of G , say $H = \langle X, Q \rangle$, is called an r -chained UkWSG of G (or simply an r -chained subgraph) if for every two sources of X such as A and B there exist m sources, $m \geq 1$, in X so that the chain $A = A_1, A_2, \dots, A_m = B$ satisfies

the condition

$$W(A_i, A_{i+1}) \leq \underline{r} \quad \text{for } i=1, 2, \dots, m-1. \quad (5-2)$$

The set of all weights of the edges between sources A_i and A_j is denoted by $W(A_i, A_j) = (d_{j1}^i, d_{j2}^i, \dots, d_{jM}^i)$.

In definition 5.9 r is any positive number such that

$$\text{Minimum}_{i,j,k} (d_{jk}^i) \leq r \leq \text{Maximum}_{i,j,k} (d_{jk}^i) .$$

The chain $A=A_1, A_2, \dots, A_m=B$ is called the r -chain of the sources A and B and is denoted by $rc(A, B)$. For simplicity, denote (5-2) by

$$rc(A, B) \leq \underline{r} . \quad (5-3)$$

Proposition 5.2: Every graph $H = \langle A, Q \rangle$ with $Q = \emptyset$ is an r -chained Uk_W for any positive integer r and for $k=1, 2, \dots, |A|$.

Proof: It is an immediate result of definition 5.9.

Proposition 5.3: Let G_1 be a Uk_1W r -chained graph and G_2 be a Uk_2W r -chained graph and let M_1 be the source size of G_1 and M_2 be the source size of G_2 .

a. $G_3 = G_1 \cap G_2$ is a Uk_3W r -chained graph for $k_3 = \text{Min}(k_1, k_2)$ provided that $M_1 = M_2$.

b. $G_4 = G_1 \cup G_2$ is not necessarily a Uk_4W r -chained graph.

Proof: If $M_1 \neq M_2$ then G_3 is an empty graph and G_4 is undefined. Now let $M_1 = M_2$. Since $G_3 \subset G_1$ (and $G_3 \subset G_2$), G_3 is a subgraph of a UkW r -chained graph. Therefore, G_3 is a Uk_3W r -chained graph. To show that G_4 is not necessarily a Uk_4W r -chained graph, let G_1 and G_2 be two graphs such that $G_1 \cap G_2 \neq \emptyset$ and there exists at least one source A_1 in G_1 and one source A_2 in G_2 such that $W(A_1, X) > \underline{r}$ for every source X of G_2 and $W(A_2, Y) > \underline{r}$ for every source Y of G_1 . There is no r -chain for A_1 and A_2 , two sources of G_4 , satisfying (5-3). In other words, $rc(A_1, A_2) > \underline{r}$ for all r -chains in G_4 . QED.

Proposition 5.4: Let $G_1 = \langle A_1, P_1 \rangle$ be a Uk_1W r -chained subgraph and $G_2 = \langle A_2, P_2 \rangle$ be a Uk_2W r -chained subgraph of G . If $G_1 \cap G_2 \neq \emptyset$ then $G_1 \cup G_2$ is a Uk_3W r -chained subgraph of G .

Proof: Let $G_1 \cap G_2 = \langle A_3, P_3 \rangle$ and $A_3 \neq \emptyset$. Given any two sources X and Y of $G_1 \cup G_2$ there are two cases:

1. X and Y are both sources of G_1 (or G_2). In this case $rc(X, Y) \leq r$ by the fact that G_1 (or G_2) is r -chained.

2. X is a source of G_1 and Y is a source of G_2 . In this case let Z be a source of $G_1 \cap G_2$. One has $rc(X, Z) \leq r$ and $rc(Y, Z) \leq r$ by the fact that G_1 and G_2 are both r -chained graphs. Let $X = X_1, X_2, \dots, X_m = Z$ and $Z = Y_1, Y_2, \dots, Y_n = Y$ be two chains connecting X to Z and Z to Y , respectively. The chain

$$X = X_1, X_2, \dots, X_m = Z = Y_1, Y_2, \dots, Y_n = Y$$

will connect X to Y and one has

$$\{rc(X, Z) \leq r \text{ and } rc(Z, Y) \leq r\} \Rightarrow rc(X, Y) \leq r.$$

therefore, $G_1 \cup G_2$ is r -chained. QED.

Definition 5.10: A $UsWSG$ of G , say $H = \langle X, Q \rangle$, is called (k, r) -bounded, $k > 0$, if for every source A in X there exist a $UkWSG$ $F = \langle \bigcup_{i=1}^k T_i, P \rangle$ of H such that $W(A, T_i) \leq r$, $i = 1, 2, \dots, k$.

Note that a (k, r) -bounded $UkWSG$ is not necessarily an r -chained $UkWSG$.

Definition 5.11: A (k, r) -bounded $UkWSG$ which is r -chained is called a (k, r) -chained $UsWSG$.

Proposition 5.5: The finite union of (k, r) -bounded Uk_iWSG 's is a (k, r) -bounded $UsWSG$.

Proof: It is an immediate result of definition 5.10.

Proposition 5.6: The finite union of (k, r) -chained Uk_iWSG 's is not necessarily a (k, r) -chained $UsWSG$, but the intersection is.

A UsWSG of G is called a maximal (k,r) -bounded $(r\text{-chained})$ UsWSG if it is (k,r) -bounded $(r\text{-chained})$ and it is not a proper subgraph of any other (k,r) -bounded $(r\text{-chained})$ UtWSG.

Theorem 5.2: Given a positive number r and $G = \langle \bigcup_{i=1}^N D_i, \Delta \rangle$, the class of all maximal r -chained Uk_i WSG's of G is a partition on G .

Proof: It is sufficient to show that any two maximal r -chained Uk_i WSG of G , say G_1 and G_2 , are disjoint. Let $G_1 \cap G_2 \neq \emptyset$. By proposition 5.4 $G_3 = G_1 \cup G_2$ is an r -chained Uk_3 WSG of G . On the other hand the fact that $G_1 \subsetneq G_3$ (or $G_2 \subsetneq G_3$) is in contradiction with the hypothesis that G_1 (or G_2) is a maximal r -chained Uk_i WSG of G . Therefore, $G_1 \cap G_2$ has to be an empty graph. QED.

Definition 5.13: Given $G = \langle \bigcup_{i=1}^N D_i, \Delta \rangle$, a Uk_i WSG of G is called a (k,r) -cluster if r is the minimum value for which G_i is (k,s) -chained for some s and G_i is maximal (k,r) -chained.

A subgraph of G which is a (k,r) -cluster for some r is called a k -cluster.

Proposition 5.7: Not every UsWSG of G , $s > 1$, is a (k,r) -cluster for some r .

Proof: It is an immediate result of definition 5.13.

Proposition 5.8: Let S be a (k,r) -cluster and S' be a (k,r') -cluster of G .

a. $r=r' \implies$ either $S = S'$ or $S \cap S' = \emptyset$.

b. $r' > r \implies$ either $S \subset S'$ or $S \cap S' = \emptyset$.

Proof: a) Let $S \neq S'$ and $S \cap S' = X \neq \emptyset$. This implies $S \cup S'$ is (k,r) -chained, by proposition 5.4, and this implies S is not maximal (k,r) -chained which is a contradiction. Therefore, if $S \neq S'$ then S and S' must be disjoint.

b) Let $S \not\subset S'$ and $S \cap S' = X \neq \emptyset$. If $S=S'$ then this implies $r=r'$ which contradicts the fact that $r' > r$. If $S \supset S'$ then $S \cup S'$ is a (k,r) -chained and this implies S (or S') is not the maximal (k,r) -chained UpWSG of G which is a contradiction. Therefore, If $S \not\subset S'$ then S and S' must be disjoint. QED.

Theorem 5.3: If T is a nonempty maximal (k,r) -bounded UpWSG for a fixed integer k then T is uniquely partitioned into m (k,r') -clusters for some $r' < r$.

Proof: Let $X_1, X_2, \dots, X_p$ be p sources of T . The fact that T is (k,r) -bounded implies, for each X_i , there exists some UkWSG T_i of T with exactly k sources (excluding X_i) such that

$$W(X_i, U) \leq \underline{r} \text{ for all } U \text{ being a source of } T_i. \quad (5-4)$$

If there are more than one such T_i let r'_i be the minimal value of r satisfying (5-4) and T'_i be the corresponding UkWSG of T . Let $\bigcap_{i=1}^q T'_{s_i} \neq \emptyset$ for $s_i \in (1, 2, \dots, p)$. Then

$$C_s = \langle \bigcup_{i=1}^q X_{s_i}, Q \rangle$$

is a maximal (k, r') -cluster, where $r' = \text{Max}(r'_{s_1}, r'_{s_2}, \dots, r'_{s_q})$ and q is the number of all such T'_i 's. The uniqueness of T'_i 's implies the uniqueness of the C_s . QED.

It is clear from the above definitions, propositions and theorems that the clustering problem is only the problem of finding the maximal (k,r) -bounded Uk_iWSG's of G . The set of k -clusters corresponding to each (k,r) -bounded Uk_iWSG of G is a well defined entity and it can be found by theorem 5.3.

Remark: Given an N -block dissimilarity matrix $\Delta = [d_{jk}^i]$, assign a rank matrix $R = [r_{jk}^i]$ to Δ as follow: Rank d_{jk}^i 's from 1 to $(N(N-1)/2) * M$ according to their magnitude and set r_{jk}^i to be the rank of d_{jk}^i . The scheme of handling ties by assigning

to the tied observations the average of their ranks should not be used in ranking d_{jk}^i 's, because it could raise some of the terms to fractional ranks which are not desirable for computational reasons. One way to break the ties is to assign the same rank to all tied observations and skip the unused ranks. An alternative is to add some random noise to the tied observations to make them unequal. Some of the reasons for replacing Δ by R in the k -clustering algorithm are as follows:

1. The use of R simplifies the computations required to find all k -clusters of a set of N data items;
2. The use of R makes it possible to save computer memory by storing each rank in at most 12 bits instead of storing it in a complete computer word;
3. The use of R makes the clustering result invariant under any monotonic transformation of Δ .

5.6.1 K-clustering Algorithm:

As mentioned before, Ling[41] uses the term "k-cluster" for his method. Here in the last two sections some definitions and theorems were presented in order to make it possible that one can generalize the idea of k -clustering to cluster a set of data items which are represented by $n \times M$ matrices (not by n -dimensional vectors.) The algorithm for the k -clustering method based on graph representations of D and Δ consists of the following steps:

Step 1-- Choose a positive integer k , set $r = M \binom{k+1}{2}$, and replace Δ by R .

Step 2-- Find B_r , the maximal (k, r) -bounded UsWSG of G .

Step 3-- If $B_r \neq \emptyset$ find all $m, m \geq 1$, k -clusters

corresponding to B_r using theorem 5.3. If $B_r = \emptyset$ go to step 5.

Step 4-- If a new cluster is found, save all relevant information about this cluster.

Step 5-- Increase r by M units (or if it is appropriate, increase it as it is specified in the following remark) and repeat the algorithm from step 2 until:

- a. the number of clusters is less than or equal to some previously defined number of clusters, if such a number is given.
- b. the largest cluster, D , is reached, that is, every data item is assigned into one group.

Remark: Let r^* be the previous r used to find maximal (k, r) -bounded UsWSG S and $I(r^*)$ be the set of all ranks of clusters already found. Instead of increasing r by M , one can set the next r to be the smallest integer greater than r^* and not in $I(r^*)$. The reason is that no integer in $I(r^*)$ can generate a new k -cluster.

Steps 1, 4 and 5 of the above algorithm are clearly well defined. To show the definiteness of the steps 2 and 3, and indeed the complete algorithm, the following computational steps of the algorithm are specified.

Step c(1)-- Read the fixed parameters N, M, n and k . Compute the initial value of r as $r_0 = M \binom{k+1}{2}$. Compute all $N(N-1)/2$ dissimilarity M -dimensional vectors corresponding to the $N(N-1)/2$ pairs of data items. The following is the order of pairs for which the dissimilarity vectors are computed:

$(D_1, D_2), (D_1, D_3), \dots, (D_1, D_N), (D_2, D_3), (D_2, D_4), \dots, (D_2, D_N), \dots,$
 $(D_i, D_{i+1}), \dots, (D_i, D_N), \dots, (D_{N-1}, D_N).$

Store the ranks in an array called $RANKS(N(N-1)/2, M)$. The ranks are stored as they are calculated. Therefore, $RANKS(I, J)$ is the J^{th} component of the dissimilarity vector between D_t and D_s where t is the integer solution of the inequality

$$(t-1)(N-t/2) < I < t(N-(t+1)/2),$$

and $s = I - N(t-1) + t(t+1)/2$.

Step c(2)-- To find the maximal (k, r) -bounded UsWSG B_r of G , first let all sources of G be included in B_r . At each iteration each source of B_r is checked once to decide whether it should remain in B_r or not. The decision rule is as follows: If D_i is a source of B_r and there exist k other sources, say $D_{j_1}, D_{j_2}, \dots, D_{j_k}$, in B_r such that $rc(D_i, D_{j_s}) \leq \underline{r}$, $s=1, 2, \dots, k$ then D_i remains in B_r . Otherwise it is deleted from the list of sources of B_r . At the completion of such a pass, if some sources of B_r were deleted from the list of the sources of B_r then at least one more pass is required to ensure that B_r is (k, r) -bounded. This iterative process terminates when no source is deleted from B_r during a complete pass, or when the number of remainder sources in B_r is less than $k+1$. If the number of sources of B_r is less than $k+1$ then B_r is an empty UsWSG. If $B_r \neq \emptyset$ then it is the maximal (k, r) -bounded UsWSG of G .

Step c(3)-- If $B_r = \emptyset$ set new r to the maximum of $r + M$ and the smallest integer not in $I(r^*)$ and greater than r^* and go to step c(2). Otherwise partition B_r into m , $m \geq 1$, maximal r -chained UsWSG's (k -clusters) $C_1, C_2, \dots, C_m$. The process of partitioning B_r into m k -clusters is as follow: Consider each source of B_r to be a node of a graph. There is a path between two nodes, A and B , of this graph if and only if $rc(A, B) \leq \underline{r}$.

The disjoint subgraphs of this graph are the k -clusters of B_r . If the number of clusters is the same as that for the previous r^* and B_r is identical to B_{r^*} then no new cluster is found. In this case update the value of r and go to step c(2). If there is only one cluster obtained and it contains all D_i 's then go to step c(6). Otherwise go to step c(4).

Step c(4) -- Compare the clusters $C_1, C_2, \dots, C_m$ of step c(3) to the previous clusters. As soon as a new cluster is found go to step c(5).

Step c(5) -- Save the elements, the size, and the r corresponding to this new cluster and modify $I(r)$.

Step c(6) -- Mark those previously obtained clusters which are a proper subcluster of the new cluster. Once a cluster is marked it will not be compared to any other cluster in subsequent iterations of the clustering procedure.

Step c(7) -- If the new cluster contains all data items then print the results of clustering and stop the procedure. If the total number of disjoint clusters at this point is less than or equal to a previously defined number of clusters, provided such a number is given, then print the clustering results and stop the procedure. Otherwise update the value of r and go to step c(2).

As an example of the k -clustering algorithm, one can consider the data items of Example 5.1. The rank matrix for these data items is stored in array $RANKS(I, J)$ as follows:

I \ J	1	2	3	4
1	38	6	4	4
2	43	30	12	30
3	59	35	9	30
4	52	7	41	9
5	13	26	30	30

I \ J	1	2	3	4
6	51	42	7	14
7	55	23	14	14
8	39	23	37	14
9	45	23	14	14
10	58	57	26	1

I \ J	1	2	3	4
11	44	9	22	46
12	36	29	26	1
13	40	46	53	46
14	60	53	46	1
15	56	14	14	46

RANKS(I,J).

Given $k=2$, the first maximal $(2,r)$ -bounded UsWSG of $G = \langle \bigcup_{i=1}^6 D_i, R \rangle$ is a maximal $(2,44)$ -bounded U4WSG with sources equal D_1, D_3, D_6 . This U3WSG is partitioned into only one 2-cluster, namely $C_1 = \{D_1, D_3, D_6\}$. The maximal $(2,48)$ -bounded U5WSG of G has D_1, D_2, D_3, D_5, D_6 as its sources and is partitioned into one 2-cluster, namely $C_2 = \{D_1, D_2, D_3, D_5, D_6\}$. Since C_1 is a subcluster of C_2 , C_1 is marked. The last maximal $(2,r)$ -bounded UsWSG of G is a $(2,56)$ -bounded U6WSG and is G itself. Since this last maximal $(2,55)$ -bounded U6WSG is partitioned into only one cluster, namely the set D itself, the algorithm will halt at step 5 (Or step C(7).)

The GCM and the k -clustering method are two different approaches. Each one has its own advantages as well as disadvantages. But there are two important reasons that one might prefer the GCM. These reasons are 1) for a large number of data items the GCA is significantly faster than the k -clustering algorithm, and 2) the GCA recomputes the dissimilarity matrix each time a new cluster is obtained, therefore, more optimized clusters are expected from the GCA.

One of the most important advantages of the k -clustering algorithm is that it can generate different clusters by changing the parameter k .

It should be mentioned that , due to the lack of a set of real data , the two algorithms given in this chapter have not been tested by a set of real data. But they have been programmed and tested by a set of generated data.

To summarize this chapter, a set of N different data items , $D = \{D_1, D_2, \dots, D_N\}$, each one of them represented by an $n \times M$ matrix along with a dissimilarity matrix, Δ , are given. Two different approaches are developed for clustering the set D . The first approach is to transform Δ to an $N \times N$ dissimilarity matrix and use one of the existing clustering algorithms to cluster the set D . The second approach is to use Δ as it is. For this second approach two different clustering algorithms are developed. One is the generalization of the Centroid Algorithm and the other one is based on graphical representations of D and Δ . For extending Ling's[41] method, some theoretical properties of the graph representation of D and Δ are discovered in this chapter.

CHAPTER VI

SUMMARY AND CONCLUSION

6.1 Summary of this Desertation:

Chapter one is an overview which includes brief descriptions of a number of major problems. These major problems are 1) the problem of representation, 2) the problem of clustering, 3) the problem of assignment, 4) the problem of applicability of assignments and 5) the problem of dynamic clusters. Chapter two is the design study of a general system called goal oriented clustering. This system is designed so as to provide some solution to the above major problems. The goal oriented clustering procedure(GOCP) consists of three separate blocks. A list of elements of the GOCP is given in chapter two along with a brief description of each element.

Chapter three is concerned with variables. Since numeric variables have been studied for a long time by most of the authors working in the area of clustering, only a reference is given for the problem related to the numeric variables. The numeric representation of NNV's is the most important step of GOCP. Two different representations of NNV's are provided in chapter three. Also, in this chapter some appropriate measures corresponding to the numeric representation of NNV's are defined. Each data item is represented by an $n \times M$ matrix and the set of all data items is represented by an N -block matrix. Finally , some motivation for all these proposed representations is given at the end of chapter three.

In chapter four an M-dimensional distance function $\mathcal{P} = (\mathcal{Y}_1, \mathcal{Y}_2, \dots, \mathcal{Y}_M)$ is defined and a very special form of this distance function is used for association measure between variables. Also, in this chapter the Minkowski metric is discussed and extended to an M-dimensional Minkowski metric. The extended Minkowski metric is used for measuring the dissimilarity between data items. A general N-block distance matrix is defined and is used to represent the dissimilarity between data items.

The following are some of the most important topics discussed in chapter five .

1. Geometric Representation of D and Δ : In this section an attempt is made to interpret the matrices D and Δ by some geometrical means.

2. General Centroid Method: Two approaches are provided for clustering a set of data items D with a dissimilarity matrix Δ . The first approach is to transform Δ to an $N \times N$ dissimilarity matrix and then use one of the existing clustering methods to cluster the set D. The other approach is to work with Δ as it is. The General Centroid Method is proposed for this second approach. Some special cases of GCM are discussed and an algorithm for the general case of the GCM is provided.

3. Graph Representation of D and Δ : In this section two new types of graphs are introduced. These new graphs are called N-partite Weighted and Uniform N-partite Weighted Graphs. The definitions and properties of these graphs are discussed and D and Δ are represented by a uniform N-partite weighted graph.

4. k-clustering Based on Graph Representation of D and Δ : Chapter five is closed by providing the necessary mathematical theory for clustering a set of N data items when they are represented by a UNWG $G = \langle \bigcup_{i=1}^N D_i, \Delta \rangle$. A number of theorems and definitions are provided in this chapter dealing with graph representations of D and Δ .

6.2 Conclusion:

One of the major results of this thesis is the development of a method of representing NNV's as part of a characteristic vector of a data item. NNV's had never before been considered as part of characteristic vector of a data item. On the other hand the contribution of this type of variable to the result of any cluster analysis is apparent. Through use of the GOCS one can represent an observation by any type of variable. One can include behavioral performance as well as age in the list of variables.

A most important result of this thesis is the development of a complete solution to the clustering problem of the GOCS. The clustering criteria proposed in this research are generalized modes of some previous clustering criteria. The representation of variables has been generalized to M-dimensional vectors, the representation of data items has been generalized to an $n \times M$ matrix, and the distance measure between data items has been generalized to an N-block matrix. The generalization is such that if no NNV is involved then the representations are the same as they were with the previous criteria. Two clustering algorithms are developed. The first

algorithm is a generalization of the Centroid Algorithm(GCA). The GCA ,if there is no threshold vector given, will generate only one new cluster in each iteration. That is, the GCA combines two of the most similar groups and then compares this new group with all the other groups to find the next most similar group. The second algorithm is the generalization of the k-clustering Algorithm. The definitions and the theorems developed for generalizing the k-clustering Algorithm are one of the important parts of this thesis.

The M-dimensional distance function introduced in Chapter Four could be used as well for factor analysis, multivariate analysis in case of three dimensional matrices, and pattern recognition. The clustering criteria introduced in Chapter Five could also be used for clustering the data items with no NNV, especially the one based on graph representation of D and Δ . The advantage of this method is that the cluster in this method is a well defined entity and is dependent on the dissimilarity matrix and the parameter k . The definition of uniform k -partite graph could be extended for pure graph theoretic reasons.

6.3 Suggested Topics for Further Work: The following is a list of suggested topics for further work. in the area of GOC.

1. Implementing a General Action Block: At this time the clustering routine of the GOCP is completed but the action routine of the GOCP is not implemented as a general computer program. The action block should be designed such that it accepts as input a set of M_0 clusters, a set of needs associate with the problem, a set of actions, and a set of control

parameters which are dependent on the given problem. The output of the action block should be a set of action sequences.

2. Implementing a General Acceptor Block: This block should be designed as a subblock of the action block. The inputs to this block are, possibly, a set of actions , a set of needs , a set of action sequences , and a set of parameters defining the nature of the given problem and specifying the relations between actions. The output of this routine is either "accepted" or "rejected."

3. Studying the Dynamic Property of a Cluster: This subject was mentioned as one of the five major problems in chapter one. As far as the author knows there has been no work related to this subject and it is perhaps a most interesting subject to be studied.

4. Developing a Different Approach: Solving the clustering problem of the GOCS in a different approach and comparing the result of that approach to what is done in this thesis could be an interesting topic for further research in this area.

Bibliography

- (1) Aitken, M.A. and Hume, N.W., "Correlation in a Singly Truncated Bivariate Normal Distribution III. Correlation between Ranks and Variate Values." *Biometrika* Vol. 53, parts 1/2 pp. 278-281, 1966.
- (2) Anderberg, R.M., "Cluster Analysis for Application." National Technical Information Service, 1972. (Published in 1973)
- (3) Babu, C.C. and Chan, W.C., "An Algorithm for Pattern Classification Using Eigenvectors."
- (4) Bailey, D.E. and Tryan, R.C., "Clustering Analysis." McGraw-Hill, 1970.
- (5) Ball, G.H. and Hall, D.J., "A Clustering Technique for Summarizing Multivariate Data." *Behavioral Science*, 12, pp. 153-155, 1967.
- (6) Ball, G.H. and Hall, D.J., "ISODATA, A Novel Method of Data Analysis and Pattern Classification." Stanford Research Inst., AD 699616, 1965.
- (7) Bartels, P.H., Bahr, G.F., Calhoun, D.W. and Wied, G.L., "Cell Recognition by Neighbourhood Grouping Technique in TICAS." *Acta Cytological* Vol. 14, No. 6, pp. 313-324, 1970.
- (8) Canover, W.J., "Practical Nonparametric Statistics." John Wiley & Sons, Inc., 1971.
- (9) Diday, E., "Optimization in Non-Hierarchical Clustering." *Pattern Recognition*, vol. 6, No. 1, pp. 17-34, 1974.
- (10) Dubes, R.C., "Information Compression, Structure Analysis and Decision Making with a Correlation Matrix." Michigan State University (1970).
- (11) Dunn, J.C. "A Graph Theoretic Analysis of Pattern Classification via Tamura's Fuzzy Relation." *IEEE Trans. on System, Man and Cybernetics*, pp. 310-313, May 1974.
- (12) Edwards, A. W. F. and L. L. Cavalli-Sforza, "A Method for Cluster Analysis," *Biometrics*, Vol. 21, No. 2, pp. 362-375, 1965.
- (13) Eigen, D.J., Northouse, R.A. and Fromm, F.R., "Cluster Analysis Based on Dimensional Information with Application to Feature Selection and Classification." *IEEE Trans. on System Man, Cybernetics*, pp. 284-294, May 1974.
- (14) Fisher, J.L. and Zudun, J. "On the Method of Theory of Clustering." *Multivariate Behavioral Research*, 4, pp. 235-250, 1969.

- (15) Fisher, L. and VanNess, J.W., "Admissible Clustering Procedures." *Biometrika* Vol. 58, pp. 99-104, 1971.
- (16) Fisher, W.D., "On Grouping for Maximum Homogeneity." *Journal of the American Statistical Association*, No. 53, pp. 789-798, 1958.
- (17) Friedman, H.P. and Rubin, J., "On Some Invariant Criteria for Grouping Data." *Journal of American Statistical Association*, Vol. 62, pp. 1159-1178, 1967.
- (18) Friedman, J.H. and Tukey, J.W., "A Projection Pursuit Algorithm for Exploratory Data Analysis." *IEEE Trans. Comp.* C-23, pp. 881-890, Sept. 1974.
- (19) Fukunaga, K. and Koontz, W.L.F., "A Nonparametric Valley-Seeking Technique for Cluster Analysis." *IEEE Com. Tran.* C-21, 2, pp. 171-179, Feb. 1972.
- (20) Fukunaga, K. and Olsen, D.R., "Representation of Non-Linear Data Surface." *IEEE Comp. Trans.* Vol. 3, C-22, No. 10, Oct. 1973, pp. 915-923.
- (21) Fukunaga, K. and Koontz, W.L.G., "A Non-Linear Feature Exhibition Algorithm Using Distance Transformation." *IEEE Comp. Trans.* Vol. C-21, No. 1, Jan. 1972, pp. 56-63.
- (22) Fukunaga, K. and Koontz, W.L.G., "Asymptatic Analysis of a Non-Parametric Clustering Technique." *IEEE Trans. on Computer*, Vol. C-2, No. 9, pp. 967-974, Sept. 1972.
- (23) Fukunaga, K., "Introduction to Statistical Pattern Recognition." Academic Press, 1972.
- (24) Fukunaga, K. and Olsen, D.R., "An Algorithm for Finding Intrinsic Dimensionality of Data." *IEEE Trans. on Computer*, Vol. C-20, No. 2, Feb. 1971.
- (25) Fukunaga, K. and Kessell, D.L., "Estimation of Classification Error." *IEEE Comp. Trans.* Vol. C-20, No. 12, pp. 1521-1527, Dec. 1971.
- (26) Gimilin, D.R. and Ferrell, D.R., "A K-K' Error Correction Procedure for Nonparametric Imperfectly Supervised Learning." *IEEE Trans. SMS*, pp. 304-306, May 1974.
- (27) Gower, J.C., "Comparison of Some Methods of Cluster Analysis." *Biometrics*, No. 23, pp. 623-637, 1967.
- (28) Harding, E. F., "The Probabilities of Rooted Tree-Shapes Generated by Random Bifurcation," *Advances in Applied Probability*, Vol. 3,
- (29) Hajek, J., "Nonparametric Statistics." Holden-Day, Inc., 1969.
- (30) Hellman, M.E., "The Nearest Neighbor Classification Rule with a Reject Option." *IEEE Trans.* Vol. SSC-6, pp. 179-185, July 1970.

- (31) Harry T. Hsu, "An Algorithm for Finding a Numeral Equivalent Graph of a Digraph." *Journal of ACM*, Vol. 22, No. 1, Jan. 1975.
- (32) Hasselblad, V., "Estimation of Parameters of a Mixture of Normal Distributions." *Technometrics*, Vol. 8, pp. 431-444, 1966.
- (33) Jance, G. N. and Williams, W. T., "Note on the Classification of Multi-Level Data." *Computer Journal*, No. 9, pp. 381-382, 1967a.
- (34) Jarvis, R. A. and Patric, E. A., "Clustering Using a Similarity Measure Based on Shared Nearest Neighbor." *IEEE Trans.* Vol. C-22, pp. 1025-1034, Nov., 1973.
- (35) Johnson, S. C., "Heirarchical Clustering Schemes." *Psychametrika*, Vol. 32, No. 3, Sept. 1967, pp. 241-254.
- (36) Jensen, R. E., "A Dynamic Programming Algorithm for Cluster Analysis," *Operation Research*, Vol. 17, No. 6, pp. 1034-1057, (1969).
- (37) Kruskal, J. B., "Multidimensional Scaling by Optimizing Goodness to Fit to a Non-metric Hypothesis." *Psychametrika*, 29, pp. 1-27, (1964a).
- (38) Labovitz, S., "In Reference of Assigning Numbers to Ranks." *American Sociological Review*, Vol. 36, No. 3, pp. 520-521, 1971.
- (39) Labovitz, S., "The Assignment of Numbers to Rank Order Categories." *American Sociological Review*, Vol. 35, No. 3, pp. 515-524, 1970.
- (40) Labovitz, S., "Some Observations on Measurement and Statistics." *Social Forces*, Vol. 46, No. 2, pp. 151-160, 1967.
- (41) Ling, R. F., "Clustering Analysis." Ph.D. dissertation, Technical Report No. 18, Dept. of Statistics, Yale University, 1971.
- (42) Lingoes, J. C., "An IBM 7090 Program for Guttman-Lingoes Smallest Space Analysis--I." *Behavioral Sciences*, 10, pp. 183-184, 1966.
- (43) Lance, G. N. and Williams, W. T., "Computer Program for Hierarchical Polythetic Classification (Similarity Analysis)." *Computer Journal*, Vol. 9, No. 2, pp. 60-64, 1966.
- (44) Lance, G. N. and Williams, W. T., "Computer Program for Monothetic Classification (Association Analysis)." *Computer Journal*, Vol. A, No. 3, pp. 246-249, 1965.
- (45) Lee, Y. H. and Rosenfeld, A., "A Clustering Heuristic for Line-Drawing Analysis." *IEEE Trans. Comp.* Vol. C-21, No. 8, pp. 904-911, Aug. 1972.
- (46) McQuitty, L. L., "Hierarchical Linkage Analysis for the Isolation of Types." *Educational and Psychological Measurement*, Vol. 202, No. 1, pp. 55-67, 1960.

- (47) Meisel, W. S., "Computer-Oriented Approaches to Pattern Recognition." Academic Press, 1972.
- (48) Meisel, W. S. and Michalopoulos, D. A., "A Partitioning Algorithm with Application in Pattern Classification and the Optimization of Decision Trees." pp. 93-102, IEEE Trans. on Compt. Vol. C-22, No. 1, Jan. 1973.
- (49) Minsky, M., "Semantic Information Processing." MIT Press, 1967.
- (50) Morrison, D. F., "Multivariate Statistical Methods." McGraw-Hill, 1967.
- (51) Mucciardi, A. N. and Gose, E. E., "An Automatic Clustering Algorithm and Its Properties in High Dimensional Space." IEEE Trans. Syst. Man. Syb., Vol. SMC-2, pp. 247-254, April 1972.
- (52) Nagy, G., "State of the Art of Pattern Recognition." Proceeding of the IEEE, Vol. 56, No. 5, pp. 830-862, 1968.
- (53) Patric, E. A., "Fundamental of Pattern Recognition." Prentice-Hall, Inc., 1972.
- (54) Parks, J. M., "FORTRAN IV Program for Q-Mode Cluster Analysis and Distance Function with Printed Dendrogram." The University of Kansas, 1970.
- (55) Rao, V. A., "A Multimodel Distribution Based Clustering Algorithm." Ph.D. Thesis at Michigan State University, Dept. of Computer Science.
- (56) Ruspini, E. E., "A New Approach to Clustering." Infor. and Control, No. 15, pp. 22-32, (1969).
- (57) Ruspini, E. H., "Numerical Methods for Fuzzy Clustering." Infor. Science, No. 2, pp. 319-350, 1970.
- (58) Sawrey, W. L., Keller, L. and Conger, J. J., "An Objective Method of Grouping Profiles by Distance Functions and Its Relation to Factor Analysis." Educational and Psychological Measurement, Vol. 20, No. 4, pp. 651-673, 1960.
- (59) Schweitzer, S. and Schweitzer, D. G., "Comment on the Pearson r in Random Numbers and Precise Functional Scale Transformation." American Sociological Review, Vol. 36, No. 3, pp. 517-518, 1971.
- (60) Scott, A. J. and Symons, M. J., "Clustering Methods Based on Likelihood Ratio Criteria." Biometrics, Vol. 27, No. 2, pp. 387-398, 1971a.
- (61) Scott, A. J. and Symons, M. J., "On the Edwards and Cavalli-Sforza Method of Clustering Analysis." Biometric Vol. 27, No. 1, pp. 217-219, 1971a.

- (62) Shepard, R. N., "The Analysis of Proximities, Multi-dimensional Scaling with an Unknown Distance Function." *Psychometrika* 27, pp. 125-139 and 219-2146, (1962).
- (63) Shepard, R. N., "Representation of Structure in Similarity Data: Problems and Prospects." *Psychometrika* Vol. 39, No. 4, pp. 373-421, Dec. 1974.
- (64) Shepard, M. J. and Willmontt, A. J., "Cluster Analysis on the Atlas Computer." *Computer Journal*, 11, pp. 57-62, 1968.
- (65) Sneath, P. H. A., "The Application of Computer to Texonomy." *Journal of General Microbiology*, Vol. 17, pp. 201-206, 1957.
- (66) Sokal, R. R. and Sneath, P. H. A., "Principles of Numerical Taxonomy." W. H. Freeman and Company, San Francisco, 1963.
- (67) Sokal, R. R. and Sneath, P. H. A., "Numeric Taxonomy." W. H. Freeman, San Francisco, 1973.
- (68) Sokal, R. R. and Michener, C. D., "A Statistical Method for Evaluating Systematic Relationships." *University of Kansas Science Bulletin*, No. 38, pp. 1409-1438, 1958.
- (69) Sokal, R. R., "Classification: Purposes, Principles, Progress, Prospects." *Science*, Vol. 185, No. 4, 57, pp. 1115-1123, Sept. 1974.
- (70) Stanat, D. F. and Das, S. K., "A Modified Training Procedure for Linear Threshold Device." *IEEE Trans. Com.* Vol. C-21, No. 4, pp. 390-397, April 1972.
- (71) Stuart, A., "The Correlation Between Variate Values and Ranks in Samples from a Continuous Distribution." *The British Journal of Statistical Psychology*, Vol. VII, Part I, pp. 37-44, 1954.
- (72) Tamura, S., Higuchi, S. and Taneka, K., "Pattern Classification Based on Fuzzy Relation." *IEEE Trans. System, Man and Cypernetics*, pp. 61-66, June 1971.
- (73) Uhr, L., "Pattern Recognition Learning and Thought." Prentice-Hall, 1973.
- (74) Ullmann, J. R., "Pattern Recognition Techniques." Butterworth, 1973.
- (75) Vinod, H. D., "Integer Programming and the Theory of Grouping." *Journal of the Ame. Stat. ASS.* Vol. 64, pp. 506-519, (1969).
- (76) Ward, J. H., Jr., "Heirarchical Grouping to Optimize an Objective Function." *Journal of the American Statistical Assoc.*, Vol. 58, pp. 236-244, 1963.
- (77) Wishart, D., "An Algorithm for Hierarchical Classification." *Biometries*, 25, pp. 165-170, 1969a.

- (78) Wolfe, J. H., "Pattern Clustering by Multivariate Mixture Analysis." Multivariate Behavior Research, Vol. 5, No. 3, pp. 329-350, 1970.
- (79) Zahn, C. T., "Graph-Theoretical Method for Detecting and Describing Gestal Clusters." IEEE Trans. Computer, Vol. C-20, pp. 69-89, Jan. 1971.
- (80) Zadeh, L. A., "Fuzzy Sets." Information and Control, Vol. 8, pp. 338-353, 1965.
- (81) Berztiss, A. T., "Data Structure Theory and Practice." Academic Press, 1975.