Summer 2005

Robust techniques and applications in fuzzy clustering

Amit Banerjee
New Jersey Institute of Technology

ABSTRACT

ROBUST TECHNIQUES AND APPLICATIONS IN FUZZY CLUSTERING

by
Amit Banerjee

This dissertation addresses issues central to fuzzy classification. The issue of sensitivity to noise and outliers of least squares minimization based clustering techniques, such as Fuzzy c-Means (FCM) and its variants is addressed. In this work, two novel and robust clustering schemes are presented and analyzed in detail. They approach the problem of robustness from different perspectives. The first scheme scales down the FCM memberships of data points based on the distance of the points from the cluster centers. Scaling done on outliers reduces their membership in *true* clusters. This scheme, known as the Mega-clustering, defines a conceptual mega-cluster which is a collective cluster of all data points but views outliers and *good* points differently (as opposed to the concept of Davé's Noise cluster). The scheme is presented and validated with experiments and similarities with Noise Clustering (NC) are also presented. The other scheme is based on the feasible solution algorithm that implements the Least Trimmed Squares (LTS) estimator. The LTS estimator is known to be resistant to noise and has a high breakdown point. The feasible solution approach also guarantees convergence of the solution set to a global optima. Experiments show the practicability of the proposed schemes in terms of computational requirements and in the attractiveness of their simplistic frameworks.

The issue of validation of clustering results has often received less attention than clustering itself. Fuzzy and non-fuzzy cluster validation schemes are reviewed and a novel methodology for cluster validity using a test for random position hypothesis is

developed. The random position hypothesis is tested against an alternative *clustered* hypothesis on every cluster produced by the partitioning algorithm. The Hopkins statistic is used as a basis to accept or reject the random position hypothesis, which is also the null hypothesis in this case. The Hopkins statistic is known to be a fair estimator of randomness in a data set. The concept is borrowed from the clustering tendency domain and its applicability to validating clusters is shown here.

A unique feature selection procedure for use with large molecular conformational datasets with high dimensionality is also developed. The intelligent feature extraction scheme not only helps in reducing dimensionality of the feature space but also helps in eliminating contentious issues such as the ones associated with labeling of symmetric atoms in the molecule. The feature vector is converted to a proximity matrix, and is used as an input to the relational fuzzy clustering (FRC) algorithm with very promising results. Results are also validated using several cluster validity measures from literature. Another application of fuzzy clustering considered here is image segmentation. Image analysis on extremely noisy images is carried out as a precursor to the development of an automated real time condition state monitoring system for underground pipelines. A two-stage FCM with intelligent feature selection is implemented as the segmentation procedure and results on a test image are presented. A conceptual framework for automated condition state assessment is also developed.

ROBUST TECHNIQUES AND APPLICATIONS IN FUZZY CLUSTERING

by

Amit Banerjee

A Dissertation
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy in Mechanical Engineering

Department of Mechanical Engineering

August 2005

APPROVAL PAGE

ROBUST TECHNIQUES AND APPLICATIONS IN FUZZY CLUSTERING

Amit Banerjee

Dr. Rajesh N. Davé, Dissertation Advisor — Date
Professor of Mechanical Engineering, NJIT

Dr. Carol A. Venanzi, Committee Member — Date
Distinguished Professor of Chemistry, NJIT

Dr. Jay N. Meegoda, Committee Member — Date
Professor of Civil and Environmental Engineering, NJIT

Dr. Zhiming Ji, Committee Member — Date
Associate Professor of Mechanical Engineering, NJIT

Dr. I. Joga Rao, Committee Member — Date
Assistant Professor of Mechanical Engineering, NJIT

BIOGRAPHICAL SKETCH

Author: Amit Banerjee

Degree: Doctor of Philosophy

Date: August 2005

Undergraduate and Graduate Education:

- Doctor of Philosophy in Mechanical Engineering,
  New Jersey Institute of Technology, Newark, NJ, 2005

- Master of Science in Product Design and Engineering,
  Indian Institute of Science, Bangalore, India, 1999

- Bachelor of Science in Mechanical Engineering,
  Regional Engineering College, Durgapur, India, 1996

Major: Mechanical Engineering

Publications:

Amit Banerjee,
"A dynamic game-theoretic approach to the modified prisoner's dilemma."
Logic Journal of the Interest Group in Pure and Applied Logics, to appear, 2005.

Milind Misra, Amit Banerjee, Rajesh N. Davé and Carol A. Venanzi,
"Novel feature extraction for fuzzy relational clustering of a flexible dopamine reuptake inhibitor,"
Journal of Chemical Information and Modeling, vol. 45(3), pp. 610-623, 2005.

Amit Banerjee and Rajesh N. Davé,
"The fuzzy mega-cluster: Robustifying FCM by scaling down memberships,"
Lecture Notes in Artificial Intelligence 3613, Eds. Wang, L. and Jin Y., Springer-Verlag, Berlin Heidelberg, pp. 444-453, 2005.

Jun Yang, Ales Sliva, Amit Banerjee, Rajesh N. Davé and Robert Pfeffer,
"Dry powder coating for improving the flowability of cohesive powders,"
Powder Technology, to appear, 2005.

Amit Banerjee and Rajesh N. Davé,
"Hopkins statistic as a measure of cluster validity,"
Manuscript in preparation, 2005.

Amit Banerjee, Jay N. Meegoda and Thomas M. Juliano,
"A conceptual framework for condition assessment of pipelines,"
Proceedings of the International Conference on Energy, Environment and Disasters (INCEED'05), Charlotte, NC, July 2005.

Amit Banerjee and Rajesh N. Davé,
"Validating clusters using the Hopkins statistic,"
Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE'04), Budapest, Hungary, pp. 149-153, July 2004.

Amit Banerjee,
"Static and dynamic fuzzy game-theoretic approaches to the modified prisoner's dilemma,"
Contributed paper at the 3rd Working Group Meeting on Soft Computing, European Research Consortium for Informatics and Mathematics (ERCIM), Vienna (Austria), July 2004.

Amit Banerjee and Rajesh N. Davé,
"The feasible solution algorithm for fuzzy least trimmed squares clustering,"
Proceedings of the 2004 Conference of the North American Fuzzy Information Processing Society (NAFIPS'04), Banff (Canada), pp. 222-227, June 2004.

Amit Banerjee, Milind Misra, Rajesh N. Davé and Carol A. Venanzi,
"Fuzzy clustering in drug design: Application to cocaine abuse,"
Proceedings of the 2004 Conference of the North American Fuzzy Information Processing Society (NAFIPS'04), Banff (Canada), pp. 308-313, June 2004.

Amit Banerjee,
"Analysis of political conspiracy games,"
Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE'03), St. Louis, MO, pp. 1043-1048, May 2003.

To

My parents and teachers

ACKNOWLEDGEMENT

I wish to express my sincere gratitude and appreciation to my advisor Dr. Rajesh N. Davé, for his help, guidance, and support throughout this research.

I thank Dr. Carol A. Venanzi and Dr. Jay N. Meegoda for giving me the opportunity to work closely with their research groups, an effort that has provided me with opportunities to learn and contribute to scientific research. I also extend my thanks to Dr. Zhiming Ji and Dr. I. Joga Rao for taking interest in my research and serving as committee members. Special thanks also go out to Dr. Thomas Juliano, Dr. Denis Blackmore and Dr. Manish C. Bhattacharjee.

I acknowledge the contributions made by Milind Misra towards the completion of this dissertation. I also acknowledge numerous referees all over the world whose scientific reviews have enriched my research contributions.

TABLE OF CONTENTS

TABLE OF CONTENTS
(Continued)

Chapter **Page**

TABLE OF CONTENTS
(Continued)

Chapter **Page**

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES
(Continued)

LIST OF FIGURES
(Continued)

CHAPTER 1

INTRODUCTION

1.1 Purpose of Research

Most of the traditional tools for formal modeling, reasoning and computing are crisp and deterministic in nature and are based on the conventional two-valued logic. However, most real world modeling and reasoning problems hardly present facts and events in such a dichotomous manner. Imprecision, uncertainty, inconsistency, vagueness, and incomplete knowledge are characteristics of a real world situation and proponents of many valued logics [1]-[3] have long argued the inefficacy of conventional dual logic to completely capture the essence of real situations. This fueled the need for alternative logic and truth representation systems of which fuzzy logic is one. Since its inception nearly 40 years ago, the theory of fuzzy sets and fuzzy logic has advanced in a multitude of ways, now encompassing many disciplines such as computer science and engineering, linguistics, social sciences, control engineering, artificial intelligence, decision theory and others. Specific applications of fuzzy set theory include expert systems and fuzzy controls [4], pattern recognition and classification [5]-[7], decision making [8], soft and granular computing [9], operations research [10], and approximate reasoning [11], [12]. This research focuses on one of the above mentioned applications – fuzzy clustering, and discusses related research issues, identifies a few fundamental problems, proposes solutions and methodologies. This work also demonstrates the applicability of fuzzy methodologies to real life classification problems.

Object and pattern classification is integral to most engineering tasks. Engineering applications in the past couple of decades have shown a marked tendency to be more knowledge driven with an emphasis on intelligent systems. Applications are widespread ranging from operations research, intelligent man-machine systems, automated manufacturing processes, quality control and diagnosis. Fuzzy logic based control mechanisms have also been used to design products such as household washing machines and vehicle suspension systems. Specific uses of fuzzy clustering have been proposed in improving acoustic properties of a room [13] and in construction simulation [14], among others. In the remainder of this section, the basics of the fuzzy set theory are presented followed by a brief discussion on fuzzy classification.

1.2 The Theory of Fuzzy Sets

Maiden publications in fuzzy set theory by Zadeh [15], and Goguen [16], [17] can be seen as attempts to generalize the classical set theory to include infinite levels of truth values – a concept known as *partial truth*. However, subsequent research and theoretical development has tried to establish fuzzy set theory as a formal theory independent of classical set theory [18], [19]. Fuzzy set theory states that an object belongs to a fuzzy set (as opposed to a classical set) with a grade of membership which has a value in the interval [0, 1]; a membership closer to zero would indicate a lower level of *belongingness* to the set as compared to a membership value closer to one. This apparently simple concept is more appropriate for capturing semantic, linguistic and real world vagueness than the classical concept of a set. Sugeno [20] defines *fuzziness* in a radically different way; a grade of fuzziness as proposed by Sugeno defines the degree of certainty of an

object belonging to a non-fuzzy subset. A piece of art could belong to a fuzzy subset, *old* (old is a vague predicate) with a certain degree of membership; at the same time it could belong to a non-fuzzy subset, *genuine* (a crisp concept) with a certain degree of certainty [19]. Although fuzzy logic could incorporate the two concepts of fuzziness because both the concepts deal with approximate rather than precise reasoning, the concept of fuzzy subsets of the universal set is what constitutes the theory of fuzzy sets.

Real world situations are very often vague and uncertain in a number of ways, of which stochastic vagueness is one aspect. Stochastic vagueness concerns the uncertainty about the future state of a system due to a lack of information and this type of vagueness has long been handled by probability theory and statistics [21]. In probability theory the events (elements of sets) and the statements concerning events are assumed to be well-defined. Fuzzy set theory attempts to handle the uncertainty and vagueness involved in the description of the semantic meanings of events, phenomena, and statements themselves and this vagueness of description is what best describes the word *fuzziness*. This is a way unifies the conceptually different definitions proposed by Zadeh and Sugeno. The comparison between probability theory and fuzzy set theory is inevitable because they are both concerned with modeling some type of uncertainty and both use the [0, 1] interval for their measures as the range of their respective functions. However, the similarity ends there, and it is important to note that they are two fundamentally different concepts. Zimmerman [22] states that comparison between the two is difficult because fuzzy set theory is not a uniquely-defined mathematical structure, such as Boolean algebra or dual logic, but is rather a very general family of theories and if anything, fuzzy set theory could be compared with other existing theories of multi-valued logic. Fuzzy

membership values indicate a degree of belongingness of an object to a fuzzy set while probability degrees denote a stochastic *chance* that an object might belong to a particular probabilistic set. Other approaches to modeling uncertainty and vagueness include Dempster-Shafer Theory of Evidence [23], Rough Sets [24], Consonant Belief Theory [25], and Possibility Theory, [19], [26] among others. For a discussion on the fundamental differences between probability theory and fuzzy set theory, the reader is referred to [27] and [28].

1.3 Fuzzy Clustering

Classification and pattern recognition are fields where fuzzy set theory is widely used. Classification deals with assigning objects to different classes on the premise that similar objects would be classified into the same class, and dissimilar objects into different classes. If the classes are not predefined, the process is known as "cluster analysis" or simply "clustering" and the classes are themselves called clusters. Other types of pattern classification include statistical classification techniques such as singular value decomposition, feature and component analysis, neural network based classification etc. A large part of this research deals with a wide range of issues involved in fuzzy clustering such as feature selection, clustering algorithms, robust clustering, and cluster validation.

The most common application of clustering methods is to partition a data-set into clusters or classes, where similar data are assigned to the same cluster and dissimilar data belong to different clusters. In real applications there is very often no sharp boundary between clusters so that fuzzy clustering is better suited for such classifications. Membership degrees between zero and one are used in fuzzy clustering instead of crisp

assignments of the data to clusters. Data may be broadly divided into two categories – object data and relational data; most image processing and pattern recognition applications make use of object data, where individual data entities are explicitly expressed in terms of their features. In relational data on the other hand, the relationship (in most cases, a similarity or a proximity relation) between the data entities is known without the explicit knowledge of the constitutive feature set. Relational clustering finds use in social sciences, taxonomy, and computational chemistry among others. Cluster analysis address three basic areas - clustering tendency, the *art* and *science* of clustering and lastly, cluster validity. Given an unlabeled data-set, clustering tendency is determining whether or not to look for substructure. Once it is decided that the data-set has a structure (indicated by a possible presence of natural groups), a model needs to be chosen whose measure of mathematical similarity may capture the structure in the data. Different models and algorithms produce different partitions of the data and so choosing a relevant model is very important. Finally, once clusters are obtained, they need to be validated – the best clustering solution has to be picked.

All clustering schemes can be broadly classified into two classes – hierarchical and partitional. Perhaps the only instance of a fuzzy hierarchical clustering technique is the approach presented in [29]; otherwise all fuzzy clustering techniques are partitional in nature and almost all of them are based on the fuzzy variant of the K-Means [30] algorithm, called the Fuzzy c-Means [7]. Fuzzy c-Means (FCM) produces a fuzzy partition of the data by optimizing (minimizing) an objective function similar to the K-Means least squares error criterion. Unlike K-Means, where every data point belongs to exactly one cluster, in a fuzzy partition all data points belong to all the clusters with

varying degrees of membership in [0, 1]. An alternative optimization scheme is used to implement FCM, which minimizes a fuzzy extension of the least squares error criteria. A cluster in FCM is defined in terms of a prototype, which in most cases is a cluster center having the same dimensions as the data-set. It has been shown that FCM converges to a local minima solution. The terminology, to be followed later on in this document, is stated here.

n = total number of feature vectors or data points,

c = number of clusters to be detected,

$\mathbf{x} = \{x_k \mid 1 \leq k \leq n\}$ is the given data-set where each data point x_k is defined by p features,

$x_k = \{x_{1k}, x_{2k}, .., x_{pk}\}$,

$\mathbf{v} = \{v_i \mid 1 \leq i \leq c\}$ is set of the prototype (cluster centers) to be detected.

The functional to be minimized, as it follows from the least squares criterion, is defined as

$$J_m = \sum_{i=1}^{c} \sum_{k=1}^{n} u_{ik}^m d_{ik}^2 . \tag{1.1}$$

The Partition Matrix is defined as $\mathbf{U} = [u_{ik}]_{cxn}$ such that

$$\begin{aligned} u_{ik} &> 0, \\ \sum_{i=1}^{c} u_{ik} &= 1, \\ 0 < \sum_{k=1}^{n} u_{ik} &< 1, \end{aligned} \tag{1.2}$$

where m is the weighting exponent known as the *fuzzifier*, and d_{ik} is the distance of data point x_k from the cluster prototype v_i. It can be the Euclidean distance, $d_{ik}^2 = \|x_k - v_i\|^2$ or

any other distance norm. The functional J_m is minimized by alternating between **U** and **v**, by initializing either one of them initially. The alternating optimization equations are

$$u_{ik} = \frac{\left(\frac{1}{d_{ik}^2}\right)^{\frac{1}{m-1}}}{\sum_{j=1}^{c}\left(\frac{1}{d_{jk}^2}\right)^{\frac{1}{m-1}}}, \text{ and} \tag{1.3}$$

$$v_i = \frac{\sum_{k=1}^{n} u_{ik}^m x_k}{\sum_{k=1}^{n} u_{ik}^m}. \tag{1.4}$$

There are several shortcomings in the FCM scheme. Being a hill descent algorithm, it only guarantees a local minimum. This is somewhat mitigated by performing FCM not just once but many times with different initializations of **U** (or **v**) which increases the chances of finding the global minima at least once. However, with a reasonably good initialization, the local minima partition is also the global minima solution. Fitting data using the minimization of a least squares criterion has been well-known to be affected drastically by the presence of noise and outliers in the data. As a reason, FCM and many of its derivatives are not robust against outliers. A lot of effort has gone into deriving robust versions of FCM, and non-FCM-based robust fuzzy clustering techniques. These will be discussed in later chapters, along with their drawbacks and proposed remedies.

1.4 Organization of the Dissertation

The dissertation is organized into six chapters based on content and information flow. It deals primarily with the development of theories and methods for robust fuzzy clustering and cluster validation. Also explored are related topics such as feature extraction and image segmentation with emphasis on specific applications.

Attempts made to robustify FCM for use with noisy data are discussed in brief in Chapter 2. Two novel robust fuzzy clustering schemes are proposed in this chapter. The first scheme introduces the concept of a *mega-cluster* and the resulting clustering scheme, called the fuzzy mega-clustering, is described in detail. Chapter 2 also details another robust clustering scheme, which achieves robustness by implementing the Least Trimmed Squares Estimator instead of FCM's least squares estimator.

Another issue closely related to clustering is cluster validity. It involves quantifying the *goodness* of a partition so that in the absence of *a priori* information about the number of clusters, various partitions (partitions for varying number of clusters) could be compared and ranked according to their relative *goodness*. The purpose of any clustering application is to uncover natural groups or natural substructure in data; however, FCM or all of its variants implicitly assume that the number of clusters is known beforehand. The literature in the field of cluster validity is reviewed and a novel cluster validity technique is proposed in Chapter 3.

At the core of any clustering problem is the issue of proper and judicious feature selection. Features are attributes based on which objects (data points in classification or patterns in pattern recognition) can be compared and thenceforth clustered. It is of utmost importance that correct features are selected and compared. At the raw level, the

number of features in a data-set is the same as its dimensionality. Clustering algorithms are known to deteriorate in performance if the dimensionality is high (almost comparable to the number of objects to be classified), a phenomenon known as the *curse of dimensionality*. A judicious process of feature selection involves dimensionality reduction and/or feature recombination. An application-dependent novel feature selection and dimensionality reduction procedure is described in Chapter 4, with a focus on relational clustering in computational chemistry.

Chapter 5 focuses on an image processing application where fuzzy clustering is used as an image segmentation tool. The application constituted a major part of a larger project – designing a framework for automatic condition state assessment for underground pipelines. Video inspection data was converted into individual image frames and analyzed, to locate damage to the interior surface of pipelines and the detailed defect information was used to construct a guideline that can allow monitoring agencies to accurately rate the condition state of the pipeline on a 4-point rating scale.

Chapter 6 presents the overall summary of the dissertation and the direction of future research in cluster analysis and discusses possibility of new applications.

CHAPTER 2

ROBUST FUZZY CLUSTERING

2.1 Introduction

Most real world data-sets are noisy. Noisy data has to be separated from *good* data in order to achieve meaningful clustering, but since this is impossible in most cases, algorithms have been developed that are robust against noise to the extent that clustering results are acceptable. In image processing, noise due to statistical distribution of the measuring/scanning/recording instrument is usually of no concern; however, completely arbitrary noise points that do not belong to the pattern or class being searched for are of real significance and have to be addressed. A few methods have been proposed,

- Identify such noisy points and remove them before applying the clustering algorithm to the data [31]. In many cases it may be extremely difficult or impossible to do.

- In the *c*-Means type of algorithms, each feature point must be assigned to one of the clusters. Hence, even noisy points must be allotted to some good cluster(s). Jolion and Rosenfeld [32] proposed a method wherein each feature point is given a weight proportional to the density of points in its vicinity. This approach works well when there is a uniformly-distributed background noise.

- Weiss [33] proposed separating the data of interest from random noise by utilizing the principle of maximum likelihood. This technique works well for fitting a single line to good data points amongst a noisy background, but cannot be extended to multiple clusters.

Apart from the approaches listed above, a large amount of research has been conducted over the years to robustify FCM and its variants by modifying the objective functional given in Equation (1.1) and/or relaxing the equality conditions in Equation (1.2). A few noteworthy and relevant approaches are listed here,

1. Noise Clustering (NC). Davé [34] introduced the concept of *Noise Cluster*. The noise cluster is of no physical significance, but is a theoretical concept which states that noisy data points have a high degree of membership in the noise cluster with an accompanying low membership in the true clusters. The algorithm now detects c+1 clusters, comprising of c good clusters with the $(c+1)^{th}$ cluster being the noise cluster. Davé proposed a scheme based on the interpoint distances for the estimation of a constant noise distance - interpoint distances reflect the structural relationship between the feature points. This definition is shown to relax the equality constraint of Equation (1.2). The noise clustering methodology is described briefly in the next section.

2. Generalized Noise Clustering (G-NC). In NC the noise cluster is defined such that all the points are equidistant from the noise prototype. It was later shown [35] that all points need not have the same distance from the noise prototype, i.e. the noise distance need not necessarily be a constant. Hence, the NC model was generalized by introducing a varying noise distance.

3. Weighted Feature Point Approach. The FCM objective functional was modified by Keller [36] with a weighting factor added to patterns in order to detect outliers. The aim of this approach is to assign small weighting factors to patterns fitting well to at least one of the clusters, while the outliers were assigned a large weight. Unlike NC or G-NC, this approach does not define a noise class; instead outliers are identified by weights assigned to them.

4. Robust Estimator-based Clustering Techniques. The quadratic loss function used as a measure of dissimilarity in FCM is the reason why FCM-based clustering procedures are highly sensitive to noise and outliers. The reason for using a quadratic loss function

is its mathematical simplicity and low computational costs. Several robust loss functions have been proposed in the literature [37], [38]. The optimal cluster center was argued to be the weighted median (instead of the weighted mean as in FCM) by Kersten [39]. The Fuzzy c-Medians (FCMED) has the L_1-norm-based objective function, and was shown to be robust [39]. It is also claimed that the alternative optimization (AO) procedure in this case is more intuitive than in FCM. Vapnik's ε-insensitive estimator was used to formulate a robust FCM-based scheme called the ε-insensitive FCM (εFCM) [40].

Several robust techniques have been proposed in the field of regression statistics, such as L_1 regression, regression based on M-estimators, generalized M-estimators, R-estimators, and L-estimators [38]. These are all low breakdown regression estimators which severely restricts their practicability. Perhaps the first high breakdown (~50% noise) regression estimator was the Repeated-Median estimator [41], following which Rousseeuw [42] introduced his Least Median of Squares (LMS) estimator. Rousseeuw [42], [43] also proposed the Least Trimmed Squares (LTS) estimator which has the standard $O(n^{-1/2})$ asymptotics and is more efficient than the LMS estimator. Both LMS and LTS belong to a family of so-called S-estimators.

Perhaps the only attempt to incorporate high breakdown robust estimation techniques with prototype-based partitioning algorithms was made by Kim *et al.* [44]. The FCM algorithm is reformulated with an LTS functional and an FCM-AO type partitioning algorithm, called the Fuzzy Trimmed c-Prototype (FTCP), is proposed. Shown to be a robust partitional scheme, FTCP however, does not utilize membership information to do the LTS trimming; it is dependent on the data being well behaved, and

like all FCM based schemes, the *goodness* of the results are heavily contingent on a *good* initialization. It does not guarantee an exact fuzzy least trimmed squares solution either.

5. Other Robust Clustering Techniques. The Least Biased Fuzzy Clustering (LBFC) algorithm [45] partitions the data-set by maximizing the total *fuzzy entropy* of each cluster, which in turn is a function of clustering memberships. The scaled LBFC clustering memberships are shown to be related to Possibilistic c-Means (PCM) [46], [47] typicalities and the resulting LBFC algorithm is robust against outliers. The Fuzzy Possibilistic *c*-Means (FPCM) algorithm [48] has an optimization functional which is a combination of probabilistic (FCM) and possibilistic (PCM) components. The algorithm uses two types of memberships, a probabilistic FCM type membership that measures the degree of sharing of a datum among the different clusters, and another possibilistic membership that provides information on intra-cluster datum relationships. For an outlier, FPCM generates low-valued typicalities, and like PCM, is a noise resistant procedure. The Credibilistic Fuzzy *c*-Means (CFCM) algorithm [49] uses datum *credibility* as the measure to delineate outliers from good datum in the data-set. As opposed to typicality in PCM, credibility of a datum represents its typicality to the entire data-set and not to any particular cluster. An outlier is shown to have a low credibility and, hence, is atypical to the data-set.

The aforementioned techniques and algorithms have been shown to be effective in clustering noisy data but they are plagued with problems of their own. The clustering results of NC and G-NC can be sensitive to variations in noise distance. Strictly speaking PCM is not a clustering algorithm but rather a mode-seeking algorithm [50], which in disguise makes it tolerant of noise. One needs to have a reasonably good estimate of

cluster variances to start with, which might not be possible in all cases. The weighted feature point approach also depends on use- specified quantities like total datum weights and a weighting exponent in addition to the fuzzifier. The least biased fuzzy clustering algorithm suffers from the same anomaly as PCM; it often generates coincident clusters since the objective functional is linearly separable. The centroids generated by FPCM are often seriously affected by outliers as would be with FCM. The concept of data credibility, although appealing, is fundamentally plagued with the logic of total credibility – according to the current formulation, while outliers have zero credibility, no datum can have full credibility (unity). The techniques based on robust statistics are not intuitive, the relaxation functions need to be chosen appropriately and carefully, and the technique itself is often hard to interpret. Moreover, all FCM-based methods, robust or not, suffer from the dependency on proper initialization.

2.2 Fuzzy Mega-Clustering

2.2.1 Noise Clustering

The proposed mega-clustering procedure shares a lot of similarity with noise clustering [34], and hence, the noise clustering (NC) algorithm is described in this subsection. The methodology relaxes the unity constraint of Equation (1.2) by defining a conceptual class called the noise cluster which is identified by a noise prototype. The noise prototype is a universal entity such that it is always at the same distance from every point in the data-set. In order to detect c true clusters, the noise cluster is defined as the $(c+1)^{th}$ cluster. The noise prototype $v_{(c+1)}$ is such that the distance $d_{(c+1)k}$, of point x_k from $v_{(c+1)}$, is

$$d_{(c+1)k} = \delta, \qquad 1 \le k \le n. \tag{2.1}$$

In other words, the FCM functional of Equation (1.1) is modified to include the definition of a special class called the noise cluster. The NC functional is

$$J_{NC} = \sum_{i=1}^{c}\sum_{k=1}^{n} u_{ik}^{m} d_{ik}^{2} + \sum_{k=1}^{n}(1 - \sum_{i=1}^{c} u_{ik})^{m} \delta^{2}. \tag{2.2}$$

The procedure is similar to FCM but instead of searching for c clusters, the algorithm searches for c+1 clusters. A scheme based on average interpoint distances was also proposed in [34] for the prediction of the noise distance δ. Interpoint distances reflect the structural relationship among the patterns in the data-set. A simplified statistical average shown in Equation (2.3) is used to calculate δ.

$$\delta^{2} = \lambda\left[\frac{1}{nc}\sum_{i=1}^{c}\sum_{k=1}^{n} d_{ik}^{2}\right], \tag{2.3}$$

where λ is called the multiplier, and is employed as a suitable scaling factor. The algorithm is sensitive to the value of the noise distance (hence, the multiplier λ). If δ is chosen to be very small, then most of the patterns in the data-set will get classified as noise points (outliers), while if δ is large, a majority of the patterns will be classified into clusters other than the noise cluster [34]. A proper selection of δ will ensure that only the outliers are classified into the conceptual noise cluster. It was also shown that the algorithm produces remarkably good results for data-sets with spherical clusters when λ is chosen between 0.05 and 0.5. For elongated clusters, a range between 0.005 and 0.5 was shown to suffice. In the next subsection, the concept of a fuzzy mega-cluster is presented in detail and parallels are drawn between the noise cluster and the fuzzy mega-cluster. In a subsequent section, results of clustering of noisy data-sets using NC are presented for the sake of comparison with the proposed mega-clustering algorithm.

2.2.2 The Concept of a Fuzzy Mega-Cluster

FCM partitions the data-set into overlapping clusters but in general works well with compact, well-separated and spherical clusters. Outliers in the data influence the location of the prototypes; as a result, the centroids are pulled towards the outlier(s). At this point, a distinction is made between true outliers and non-conforming non-outliers (which in the course of the dissertation will be referred to as non-outliers). While the former data vectors are noise and do not belong to any cluster in the data, non-outliers are data vectors that neither belong to any cluster in the data nor can be considered noise. In other words, non-outliers can be considered to be equally likely to belong to any cluster in the data-set and because of this reason, such data vectors can not be assigned to any one particular cluster. The difference is clearly indicated in Figure 2.1, and any clustering algorithm should have the power to treat such entities differently. Unfortunately FCM and almost all robust clustering algorithms (except NC) fail to differentiate between true outliers and non-outliers; FCM will assign memberships of (0.5, 0.5) to both x' and x''.

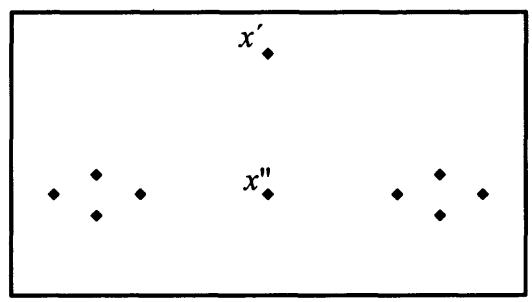


Figure 2.1 True outliers and non-conforming non-outliers, x' is a true outlier, equally *unlikely* to belong to either cluster; x'' is a non-outlier, equally *likely* to lie in either cluster.

A cluster called the mega-cluster is defined that views data vectors differently depending on how they belong to any good cluster in the data. Suppose, in a two cluster data-set, the datum x is a good representative of cluster *I*. In such a case, the membership of x in cluster *I* is the largest, followed by its membership in the mega cluster and it has the smallest membership in cluster *II*. On the other hand, if x' is a true outlier, its membership in the mega cluster is the largest, followed by relatively small memberships in the two good clusters *I* and *II*. This treatment is fundamentally different from the concepts of noise cluster and credibility of a data point vis-à-vis the entire data-set. With the noise cluster, the membership of x is the largest in cluster *I*, followed by its membership in cluster *II*, and it has a comparatively small membership in the noise cluster. However, like the mega-cluster, x' has a high degree of membership in the noise cluster, followed by low memberships in the two clusters. The concept of credibility as opposed to membership is defined as the degree of *representativeness* of a data point to the entire data-set and, as per definition, noise points have low data credibility and good data points have high credibility. If x'' is a non-outlier, its membership in the mega-cluster is the highest followed by almost equal memberships in the two clusters; moreover, if it is a symmetrically located non-outlier (as is x'' in Figure 2.1), the sum of its memberships in the two clusters would equal its membership in the mega-cluster. This treatment allows for the subjective fact that such a non-outlier is equally likely to be considered part of either of the clusters but most likely considered *noise*.

The mega-cluster can be thought of as a *super-group* encompassing the entire data-set and views the data points differently depending on their belongingness in true clusters of the data. A further proposition is that a mega-cluster membership is

representative of credibility and noise memberships (as well as true FCM memberships). A high mega-cluster membership corresponds to a high noise membership and low credibility; likewise, a low mega-cluster membership corresponds to a low noise membership and a high credibility (and thus a high true membership) of the data point in one of the clusters. This cluster can not be detected by the standard FCM formulation. It is further assumed that while all data points have varying degrees of membership in the mega-cluster, they are all equally representative of the mega-cluster in the sense that distance of the data points from the mega-cluster is zero (a concept similar to a constant noise distance of NC). Conceptually for the purposes of prototype calculations, the mega-cluster can be considered to be composed of n-point prototypes, each located exactly at the n data points. Furthermore, the membership of a datum summed over the true clusters and the mega-cluster is unity, hence, the FCM update equations can be used without any change of form.

2.2.3 The Proposed Mega-Clustering Algorithm

The aim here is to reduce the sensitivity of the FCM formulation towards noise by scaling down the memberships produced by FCM, in an inverse proportion to the cluster-datum distance. To speed up the convergence of FCM, two membership scaling procedures were proposed viz., Rival Checked-FCM [51], and the Suppressed-FCM [52]. In every iteration of the FCM-AO scheme and for each datum, the two algorithms reward the largest membership by scaling it up by a constant factor. Rival Checked-FCM (RCFCM) then suppresses the second highest membership, while Suppressed-FCM (SCFM) suppresses all other memberships by a corresponding factor. Because of the scaling up, the two algorithms were found to be highly sensitive to noise. In experiments with

RDFCM it is seen that (in almost all cases) it does not converge to a stable solution because the scaling disturbs the sequence of memberships (as a result there is much oscillation between successive iterations). In SFCM, if x_j has the maximum membership in cluster p, denoted by u_{pj}, then for a scaling factor α, the following operations are performed

$$u_{pj} = 1 - \sum_{i \neq p} u_{ij} = 1 - \alpha + \alpha u_{pj} , \tag{2.4a}$$

$$u_{ij} = \alpha u_{ij}, \qquad i \neq p . \tag{2.4b}$$

The magnification in Equation (2.4a) is performed on the highest membership of x_j, and the suppression in Equation (2.4b) is done on the rest of the memberships (other than the highest one). For $0.1 \leq \alpha \leq 0.3$, where SCFM behaves more like Hard c-Means (HCM), it is found that the algorithm generates singleton noise clusters and hence, with appropriate modifications can be used as an outlier diagnostic tool.

The proposed algorithm is based on the following logic – what essentially distinguishes a good data point from an outlier is their distance (dissimilarity) from a representative prototype. This difference becomes muddled in the presence of noise in the data because of the centroid-pulling effect of the outliers. Hence, for noisy data, if one can provide a mechanism which would accentuate this difference, one could conceptually reproduce results similar to FCM on a noise-free data. The proposed algorithm tries to underline this difference between good points and outliers by scaling down membership values of a data point across all clusters, in an inverse proportion to its distance from the cluster prototypes. Hence, the effective membership of all points in true classes is less than one. This scaling down is more prominent for outliers which

successively undergo a drastic reduction in memberships, which relates to a corresponding increase of its membership in the conceptual mega-cluster.

In the proposed algorithm (henceforth referred to as MC), the memberships as calculated by FCM are then modified depending on the datum-cluster center distance. For unusually large distances, the scaling is more intense and is achieved by scaling with respect to the maximum distance in the data-set, and is shown in Equation (2.5). This scaling repeatedly on the outliers reduces their memberships rapidly as compared to scaling done on good datum. For reasonable datum-cluster center distances, the scaling is moderate, and is done with respect to the sum of distances of the datum from all the c cluster centers as in Equation (2.6).

$$intense\ scaling: u_{ij} = \left[1 - \frac{\beta d_{ij}}{\max\limits_{\substack{i=1,..,c \\ j=1,..,n}}(d_{ij})}\right] u_{ij}, \tag{2.5}$$

$$moderate\ scaling: u_{ij} = \left[1 - \frac{\beta d_{ij}}{\sum_{k=1}^{c} d_{kj}}\right] u_{ij}. \tag{2.6}$$

This modification is introduced in the FCM-AO after the completion of FCM membership update of Equation (1.3). An if-else condition (based on a fraction ρ of the maximum distance, d_{max}) is used to decide whether to use a moderate or an intense scaling, and the condition checks how unusually large a particular datum-cluster center distance d_{ij} is, as compared to d_{max}. The scaling is comparable in content to the credibility of a datum x_j as proposed by [49], and given by

$$\Psi_j = 1 - \frac{(1-\theta)\alpha_j}{\max_{k=1,..,n}(\alpha_k)}, \text{ where } \alpha_j = \min_{i=1,..,c}(d_{ij}). \tag{2.7}$$

As with credibility, $\beta = 0$ reduces the formulation to FCM and at $\beta = 1$, the formulation serves as a *complete noise reduction* algorithm. At levels between $\beta = 1$ and $\beta = 0$, the algorithm tries to balance between assigning a low membership to true outliers and assigning comparatively higher memberships to non-outliers, such as x'' in Figure 2.1. If it is known that the data is noise free, a choice of $\beta = 0$ would reproduce FCM results known to be fairly accurate in the absence of noise. In the presence of noise and general outliers, a judicious choice of β needs to be made; as inferred from the experiments presented in the next section, it is seen that a value of $\beta = 1$ generates good partitions in noisy data-sets. This scaling down of memberships relaxes the unity constraint in Equation (1.2); the resultant constraint is an inequality condition, and the membership of a datum x_j in the mega-cluster is given by

$$u_{MCj} = 1 - \sum_{i=1}^{c} u_{ij}. \tag{2.8}$$

The MC algorithm is presented below,

Fix m, c, β and ρ. Initialize membership matrix U = u_{ij}
Fix termination condition ε (very small number, usually 10^{-5})
do {
    Store current memberships as U^{old}
    Find cluster centers v = v_i: Equation (1.4)
    Calculate distances $d_{ij} = \| v_i - x_j \|^2$
    Calculate maximum distance d_{max} among all d_{ij}'s
    Recalculate memberships U = u_{ij}: Equation (1.3)
    if ($d_{ij} > \rho d_{max}$)
        then scaling down u_{ij} intensely: Equation (2.5)
    else
        scale down u_{ij} moderately: Equation (2.6)
} *while* ($|U - U^{old}| > \varepsilon$)

In the next section, another novel robust clustering algorithm based on the least trimmed squares estimator is presented. Results of the MC algorithm on test data-sets are presented later.

2.3 Fuzzy Least Trimmed Squares Clustering

2.3.1 The Least Trimmed Squares Estimator

The Least Squares (LS) regression model fits the best regression line by minimizing the squared residuals of all the random observations from an arbitrarily-fitted line. The best fit is the arbitrary fit that results in the minimum sum of squared residuals. In case one or more explanatory variables are recorded erroneously, the corresponding observations might turn into *leverage points* and such points tend to pull the LS fit line towards them. The resultant fit in the presence of noisy observations is a bad fit and is one of the biggest criticisms of the LS regression model. The Least Trimmed Squares (LTS) [42] model rectifies this anomaly by minimizing the sum of the squares of the smallest h of the total n residuals. The estimator achieves robustness by trimming the $(n - h)$ observations, the ones with the large residuals. The functional is of the form,

$$\text{Minimize} \sum_{i=1}^{h} (r^2)_{i:n}, \tag{2.9}$$

where $(r^2)_{1:n} \leq \ldots \leq (r^2)_{n:n}$ are ordered squared residuals. The minimization of the sum of trimmed squared residuals results in an estimation of the parameter set Θ, which in the case of linear fit estimation fitting is the slope and the set of y-intercepts. The LTS estimator breaks down for $h < n/2 + 1$; in other words the LTS estimator can obtain a

good estimate of Θ from a noisy data-set with as much as 50% contamination. This high breakdown property of the LTS estimator makes it potentially very attractive.

The motivation for using the LTS criterion, however, does not provide any clue to its implementation, which involves determination of which cases to trim. One of the few currently-known exact algorithms for obtaining the LTS fit is a combinatorial one which involves fitting regression to every subset of size h of the data-set and subsequently finding the minimal residual sum of squares. This approach is practical only if both the data-set and the amount of trimming are small. The basic resampling scheme [42], [53] is a popular method of performing LTS fit. Here a combinatorial scheme is used on elemental subsets of size p ($p << h$) where, p is the total number of explanatory variables in the model. This approach is popular because of its manageable combinatorics - the number of possible elemental subsets is many orders of magnitude less than the number of possible trimmed sets. However, the basic resampling scheme can be applied to multiple regression models ($p > 1$) only; the fit obtained is crude with a high degree of approximation (depending on the number of elemental subsets considered), and does not in general yield the exact LTS solution.

2.3.2 The Feasible Solution Algorithm

The exact LTS solution is the ordinary LS fit to some subset of size h of the data which cannot be improved by any single pairwise exchange of one observation in the subset for one outside. This known form of the exact solution and the resulting necessary condition forms the basis of the simulated annealing approach [54] and the feasible solution algorithm [55]. Both are probabilistic schemes with guaranteed convergence to a global optimum and, hence, are an improvement over the methods discussed in the previous

sections. In simulated annealing, $n - h$ cases are trimmed at random and a trial solution obtained. Then observations in the retained set (of size h) are swapped, one at a time, with observations in the trimmed set, and a swap that leads to a reduction in the residual sum of squares is considered favorable. The retained set is then modified to include the favorable observation. The scheme requires tuning constants like most robust estimation techniques and hence, its performance depends on the choice of these parameters; a poor choice might even force the algorithm to terminate before an optimum is found [55].

The feasible solution algorithm defines a feasible solution as the local optimum solution obtained by a refinement (pairwise swapping) process from a randomly-picked starting subset of cardinality h. From a starting subset, observations are swapped pairwise with the trimmed subset and the starting subset is modified to include the observation (in the trimmed subset) that produces the largest reduction in the residual sum of squares in the retained set. This modified retained subset is then subjected to further refinement until no pairwise swap results in a reduction of the residual sum of squares. The retained subset is called a feasible solution and a single application of the refinement process from a starting subset will always lead to some feasible solution. However, this might not be the true LTS fit. To obtain the exact solution, the global optima fit in this case, the refinement process needs to be repeated using distinct starting subsets and following each to its feasible solution. The exact solution is the feasible solution with the lowest residual sum of squares. Exhaustive enumeration involves the application of the refinement process to all the possible starting subsets in the data-set. For a data-set of size n, with $n - h$ trimming, there are ${}^{n}C_{h}$ possible starting subsets. For

large n and h, this is a considerable number to account for and, hence, exhaustive enumeration is only possible for small data-sets with a small amount of noise.

Most feasible solutions can be reached from more than one starting subset. In fact the most frequent feasible solution is almost always also the exact LTS solution, but there are cases reported in [55] where this is not true. The *domain of attraction* of a particular feasible solution is the set of all starting subsets of size h which terminate in that solution. Let $\boldsymbol{I}$ be a feasible solution and π be the proportion of all starting subsets that terminate in $\boldsymbol{I}$. The probability of finding the global optimum solution from T starting subsets is given by

$$Q = 1 - (1 - \pi)^T . \tag{2.10}$$

In other words, if $\boldsymbol{I}$ is an exact LTS solution but is infrequent (low π), one would require a large number of starting subsets (high T) to locate $\boldsymbol{I}$ with high confidence. For all practical purposes however, values of T in the low dozens are enough to guarantee finding a global optima. Another value of interest is the average number of swaps made from a random starting point within the domain of attraction of $\boldsymbol{I}$; denoting this by $E(P)$, the expected number of intermediate LS fits made en route to the feasible solution is given by

$$E(\text{number of LS fits}) = T\,[1 + E(P)]. \tag{2.11}$$

In the next subsection, a fuzzy clustering scheme based on the feasible solution algorithm is presented. LTS based regression studies do not address the issue of how to choose an appropriate trimming ratio; in fact $h = n/2 + 1$ may be a good choice for line fitting, but fixed amount of trimming is not desirable in clustering [44]. This is the motivation for the development of a methodology along the lines of the unsupervised

FTCP (UFTCP) [44], using cluster validity techniques to justify picking an appropriate value of h.

2.3.3 FS-FLTS – When Amount of Contamination is Known

Clustering differs from regression analysis in a few fundamental aspects – (a) regression groups the data together in one cohesive group by finding the best fit line while clustering finds multiple clusters in data, (b) the estimation parameters in clustering are most often cluster mean and cluster shape characteristics (such as spread and deviation from the mean), (c) there are no explanatory and response variables in clustering, the data-set for clustering in d-dimensions is a set of n observations with d independent features, and (d) clustering lends itself to well-established iterative algorithmic schemes while there are no known iterative schemes to find the best fit LS fit. In spite of these differences, K-means-based clustering is essentially an implementation of the LS regression scheme on a *mixed* data-set, locating multiple best fit shapes in the data. One can similarly capitalize on the known form of the exact solution of the LTS approach and state the following – the exact LTS clustering solution is the K-means partition performed on some subset of size h of the data which cannot be improved further by any single pairwise exchange of one data vector in the set for one outside. The issue now is not one of reformulating the objective functional to incorporate a trimming function, but to find the most suitable subset of size h on which to perform a K-means (or related) clustering.

An FCM-based clustering scheme with the feasible solution algorithm is proposed here. The determination of π requires an exhaustive enumeration of the entire subset space of the data-set and since an exhaustive enumeration is rarely possible, we present a novel method that automatically blocks any further swapping and clustering based on the

average number of case swaps required to reach a feasible solution. The proposed scheme is a two-phase one. The first phase starts with randomly dividing the data-set into two subsets, X_0 of size h and Y_0 of size $n - h$. Let X be the retained subset and Y, the trimmed subset. The subset X_0 is clustered using FCM and the objective functional value, as defined in Equation (1.1), is stored as J_0. Now, one data vector x_i in the subset X_0 is randomly exchanged for another vector x_j in Y_0 and the modified subset labeled is as X_{ij}. This modified subset is clustered using FCM and the objective functional value is stored as J_{ij}. The case number of the swap is also noted. The change in the objective functional is calculated as $\Delta J_{ij} = J_0 - J_{ij}$. The subset X_{ij} is restored back to the starting set X_0 and the process repeated for all possible random but distinct combinations of (i, j). If all ΔJ_{ij}'s are negative, then the starting subset X_0 is the feasible solution. Otherwise, the intermediate modified set X_{ij} that results in the largest positive value of ΔJ_{ij} is the feasible solution obtained from the starting subset X_0 and the corresponding swap case number is stored in a result matrix.

This process of refinement of a starting subset is repeated for a pre-specified number of times. Like the randomized Hough transform [56] matrix, a trend can be observed here, leading to the theory that a particular feasible solution requires an almost equal number of case swaps starting from a randomly-selected set X_0. In other words, if two starting subsets result in the same feasible solution $\boldsymbol{I}$, then the refinement process would be characterized by an almost equal number of case swaps, k, in both cases. In the next phase of the refinement process, the swapping is stopped after the number of case swaps has exceeded the largest value of case swaps as calculated from the result matrix. This is done to keep the resource-intensive FCM calculations to a minimum. This phase

is also terminated after a fixed number of random starts from subsets X and Y. The number of starting subsets required for the two phases is heavily dependent on the typicality of the data-set in question; however, some general guidelines are proposed in the next section. The feasible solution algorithm for the implementation of fuzzy least trimmed squares clustering (this is referred to as FS-FLTS) is given below,

Fix h and c, $h < n$, $2 \leq c \leq h$, fix fuzzifier $m > 1$;
Fix $T = (T_1, T_2)$ as the number of starting subsets to be considered;

Phase I:

for ($t = 1$ to T_1)
    Randomly select subset X_t of size h and a complementary Y_t of size $n - h$;
    Cluster X_t using FCM, store objective functional value as J_t;
    for ($i = 1$ to h)
        for ($j = 1$ to $n - h$)
            Swap x_i in X_t for x_j in Y_t, $X_t \rightarrow X_{ij}$;
            Store case swap as k_{ij};
            Cluster X_{ij} using FCM, store objective functional value as J_{ij};
            Store $\Delta J_{ij} = J_t - J_{ij}$;
            Swap back $X_{ij} \rightarrow X_t$;
            $j \rightarrow j + 1$;
        end for (j)
        $i \rightarrow i + 1$;
    end for (i)
    Find largest positive ΔJ_{ij} and retrieve the corresponding k_{ij};
    X_{ij} is the feasible solution for X_t;
    Populate the result matrix $R_1 = [X, k]_t$;
    $t \rightarrow t + 1$;
end for (t)

Locate exact solution X'_1 (frequency t'_1) from R_1 and find corresponding average case swaps k';

Phase II:

for ($t = 1$ to T_2)
    Randomly select subset X_t of size h and a complementary Y_t of size $n - h$;
    Cluster X_t using FCM, store objective functional value as J_t;
    for ($l = 1$ to k')
        Swap ($X_t \leftrightarrow Y_t$), $X_t \rightarrow X_l$;
        Cluster X_l using FCM, store objective functional value as J_l;

Store $\Delta J_l = J_t - J_l$;
Swap back $X_l \rightarrow X_t$;
$l \rightarrow l + 1$;
end for (l)
Find largest positive ΔJ_l;
X_l is the feasible solution for X_t;
Populate the result matrix $R_2 = [X]_t$;
$t \rightarrow t + 1$;
end for (t)

Locate exact solution X'_2 (frequency t'_2) from R_2, $\pi = t'_2 / T_2$.

The exact solution in phases I and II should be the same feasible solution; this provides the motivation to skip phase I for large data-sets by assuming an arbitrary value for k'. A case study is presented later in this chapter where only phase II is implemented and satisfactory results are obtained.

2.3.4 FS-FLTS – When Amount of Contamination is Unknown

Until now, it is assumed that h is constant and known *a priori*. However, it is practically impossible to ascertain the amount of contamination in a data-set before subjecting it to clustering. An approach similar to the one in [44] is presented here, wherein the value of h is varied and the optimum value is picked based on a suitable validity criterion. In this case the Xie-Beni compactness index [57] is used as the validity criterion. The retention ratio is defined as

$$H = h/n, \tag{2.12}$$

and in this implementation, H is varied from 1 until $(n/2 + 1)/n$, in steps of ΔH. For every H, the proposed FS-FLTS scheme is implemented. The compactness index is then calculated for the exact solution for all the values of H as

$$S(H) = \frac{J_h}{h\left[\min_{1 \le i,j \le c, i \ne j} ||v_i - v_j||^2\right]}, \qquad 1 \le i, j \le c, \tag{2.13}$$

where J_h is the value of the objective functional associated with the exact solution for the particular value of h. Since all refinement calculations are based on comparing the objective functional value of the modified subset X_{ij} with that of the starting subset X_0, calculation of *S(H)* does not involve any additional computation. The value of the objective functional for the exact solution for a particular value of h is given by

$$J_h = \sum_{i=1}^{c} \sum_{k=1}^{h} u_{ik}^2 d_{ik}^2 , \tag{2.14}$$

where u_{ik}, and d_{ik} follow the usual FCM definitions. It is presumed that the number of good clusters in the data-set, c, is known beforehand and is a constant throughout the unsupervised implementation. With increasing H, it is likely that the resultant clusters would get more and more compact as more cases are trimmed off and hence, *S(H)* will have a monotonically decreasing tendency with H. For this reason, the incremental change in *S(H)* as given by Equation(2.5) is calculated as,

$$s(H) = S(H + \Delta H) - S(H), \qquad H < 1. \tag{2.15}$$

The feasible solution fuzzy clustering algorithm for unknown h (for brevity, it is referred to as hFS-FLTS) is summarized below.

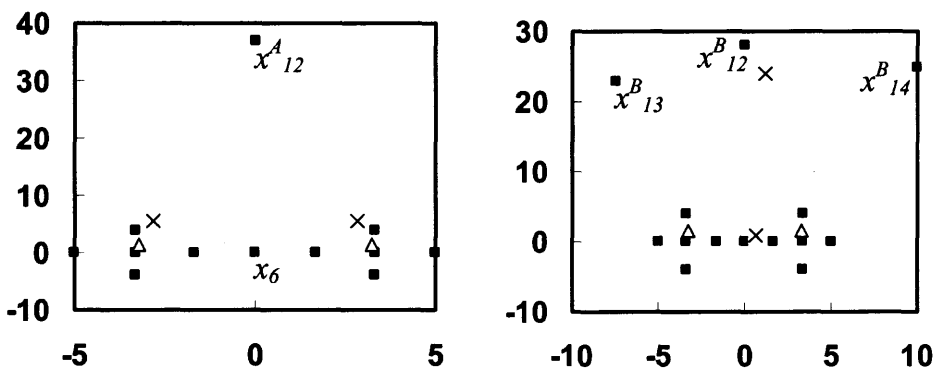
Fix c, $2 \le c \le n/2$, fix fuzzifier $m > 1$ and ΔH;
Fix $T = (T_1, T_2)$ as the number of starting subsets to be considered;
Set $H = 1$;

Repeat
Run the FS-FLTS algorithm with T number of starting subsets;
From the feasible solutions, locate the exact solution;
For the exact solution calculate S_H as in (2.31);
If $(H \neq 1)$
Calculate $s(H)$ from (2.33);

$H = H - \Delta H$;
Until ($H \leq 0.5$)
Pick the value of h corresponding to the largest ΔS_H.

2.4 Simulation and Results

The three data-sets presented in [48] called X11, X^A12 and X^B14, were analyzed with the MC algorithm. X11 is a noise-free data-set consisting of 11 two-dimensional vectors while X^A12 and X^B14 are noisy versions of X11. A comparison of FCM, FPCM and CFCM on X11, X^A12 and X^B14 is presented in [49]. The performance of FCM and NC was compared with the MC algorithm on the three data-sets and is presented here. The vector x_6 is a non-outlier with equal probability of belonging to the two underlying clusters in X11. The two well-defined clusters lie on either side of x_6. The vectors x^A_{12} (in X^A12) and x^B_{12}, x^B_{13}, and x^B_{14} (in X^B14) are true outliers. In the implementation, $c = 2$, $m = 2$, and $\varepsilon = 0.001$ are used for both FCM and the MC algorithm (for $\beta = 1$).



(a) (b)

Figure 2.2 Centroids generated by the MC algorithm (Δ-MC), and FCM (x-FCM) on the (a) X^A12 and, (b) X^B14 data-set.

For a prototype initialization of $v_1 = x_2 - 10^{-3}$ and $v_2 = x_{11} - 10^{-3}$ in the case of X^A12, it was found that the MC algorithm performs better than FCM; the cluster centers generated are shown in Figure 2.2(a). The data vectors are shown by solid *squares*, FCM centroids are depicted by *crosses*, and the MC centroids by small *triangles*. In fact, the results are comparable to the ones generated by CFCM and certainly better than FPCM (see Figure 2. of [49]). FCM also fails to distinguish between x_6 and x^A_{12}. The proposed algorithm produced a higher membership for x_6 compared to x^A_{12} in the two clusters. For the data-set X^B14 shown in Figure 2.2(b), the results provide a striking contrast – while FCM groups the three outliers in one cluster and the rest of the data-set into another, the MC algorithm finds the two real clusters. This is comparable to what FCM would generate on the noise-free data-set, X11. The MC algorithm generates the same partitions over a wide range of distance factors for intense scaling (ρ = 0.8, 0.5, and 0.3) in case of X^A12 while there was a little difference in memberships for X^B14 when intense scaling was done for ρ = 0.8, as compared to the memberships obtained for ρ = 0.5. The results presented in Table 2.1 pertain to intense scaling done for ρ = 0.5 (the difference was insignificant, affecting only third decimal places in the memberships). In both cases, the symmetrically-located x_6 has a membership of about 0.5 in the mega-cluster and the true outliers have relatively large memberships in the mega-cluster compared to their memberships in the *good* clusters.

The proposed algorithm is also compared with NC on the X^A12 data-set and the results presented in Tables 2.2 and 2.3. For λ = 0.05, the results of NC are similar to the one generated by MC; NC also identifies the outliers correctly for λ = 0.5 and λ = 5.0, however, it starts behaving like FCM for higher values for λ. On the other hand for a

fixed $\beta = 1$, results for MC clustering do not depend on the value of the parameter ρ. For $\rho = 0$, the vector x_6 is equally shared between the two clusters (and subsequently, has a low membership in the mega-cluster). For $\rho = 0.1$ till $\rho = 0.9$, the results are consistent with the philosophy of the mega-cluster. For $\rho = 1$, the symmetric outlier x^A_{12} shows an abnormally higher tendency of belonging to the left cluster (cluster 1), than cluster 2. This leads to the hypothesis that MC requires a combination of intense and moderate scaling to produce satisfactory results. If all the scaling is either intense ($\rho = 0$) or moderate ($\rho = 1$), the results are skewed partitions incongruent with the underlying philosophy.

Table 2.1 Memberships for X^B14 Generated by the MC Algorithm

Vector	Feature 1 X	Feature 2 Y	Memberships for X^B14		
			u_{1j}	u_{2j}	u_{MCj}
x_1	-5.00	0.00	0.930136	0.001265	0.068599
x_2	-3.34	1.67	0.916823	0.001806	0.081371
x_3	-3.34	0.00	0.997538	0.000002	0.002460
x_4	-3.34	-1.67	0.865677	0.004842	0.129481
x_5	-1.67	0.00	0.794419	0.011815	0.193766
x_6	**0.00**	**0.00**	**0.241098**	**0.259063**	**0.499839**
x_7	1.67	0.00	0.009883	0.811054	0.179063
x_8	3.34	1.67	0.002283	0.906717	0.091000
x_9	3.34	0.00	0.000000	0.999322	0.000678
x_{10}	3.34	-1.67	0.003771	0.880852	0.115277
x_{11}	5.00	0.00	0.001369	0.927366	0.071265
x^B_{12}	**0.00**	**27.00**	**0.041163**	**0.037726**	**0.921111**
x^B_{13}	**-7.00**	**23.00**	**0.179158**	**0.094016**	**0.726826**
x^B_{14}	**10.00**	**25.00**	**0.000000**	**0.089727**	**0.910273**

(Note: Compare memberships with Table 1, [49], p. 1468)

Table 2.2 Memberships for X^A12 Generated by the NC Algorithm

Vector	$\lambda = 0.05$		$\lambda = 0.5$		$\lambda = 5$	
	iterations = 10		*iterations* = 7		*iterations* = 7	
	u_{1j}	u_{1j}	u_{1j}	u_{1j}	u_{2j}	u_{2j}
x_1	0.662285	0.662285	0.910962	0.045792	0.940426	0.054364
x_2	0.685851	0.685851	0.909057	0.055513	0.967706	0.030401
x_3	0.996423	0.996423	0.998872	0.000705	0.991737	0.007790
x_4	0.685413	0.685413	0.905415	0.057674	0.899251	0.094387
x_5	0.701785	0.701785	0.886903	0.084939	0.904416	0.092476
x_6	**0.291109**	**0.291109**	**0.467284**	**0.466376**	**0.496694**	**0.496117**
x_7	0.069130	0.069130	0.085349	0.886393	0.092742	0.904143
x_8	0.042360	0.042360	0.055540	0.909046	0.030367	0.967743
x_9	0.000472	0.000472	0.000681	0.998911	0.007791	0.991735
x_{10}	0.042385	0.042385	0.057704	0.905398	0.094468	0.899168
x_{11}	0.032114	0.032114	0.045685	0.911203	0.054307	0.940490
x^A_{12}	**0.004825**	**0.004825**	**0.044354**	**0.044353**	**0.247084**	**0.247090**

Table 2.3 Memberships for X^A12 Generated by the MC Algorithm ($\beta = 1$)

Vector	$\rho = 0$		$\rho = 0.1 - 0.9$		$\rho = 1.0$	
	iterations = 7		*iterations* = 7		*iterations* = 7	
	u_{1j}	u_{1j}	u_{1j}	u_{1j}	u_{2j}	u_{2j}
x_1	0.949995	0.045614	0.928494	0.001326	0.909804	0.002131
x_2	0.939464	0.056887	0.892885	0.003033	0.965825	0.000297
x_3	0.999291	0.000668	0.999954	0.000000	0.975194	0.000156
x_4	0.939464	0.056887	0.892878	0.003034	0.792975	0.011992
x_5	0.911086	0.086094	0.805989	0.010451	0.783770	0.013154
x_6	**0.496836**	**0.496253**	**0.250279**	**0.249722**	**0.244982**	**0.255068**
x_7	0.086349	0.910825	0.010507	0.805502	0.010395	0.806482
x_8	0.056905	0.939448	0.003037	0.892824	0.003450	0.885981
x_9	0.000654	0.999306	0.000000	0.999949	0.000000	0.999955
x_{10}	0.056905	0.939448	0.003037	0.892827	0.002712	0.898558
x_{11}	0.045547	0.950071	0.001321	0.928620	0.001344	0.928034
x^A_{12}	**0.000004**	**0.000000**	**0.000005**	**0.000000**	**0.260222**	**0.000000**

For purposes of illustration, the FS-FLTS scheme was first applied on a simple synthetic data-set. The data-set shown in Figure 2.3 consists of $n = 13$ two-dimensional vectors, with four outliers, $h = 9$. The number of distinct starting subsets (X, Y) is $^{13}C_9 =$

715. For exhaustive enumeration, the total number of case swaps to be made for each starting subset is 36, and hence, FCM was performed 36 x 715 = 25,740 times for the entire exhaustive enumeration process. The parameters chosen for FCM are: $c = 2$, $m = 2$ and $\varepsilon = 10^{-5}$. The feasible solution for all the 715 starting subsets trims the four outlier vectors and, disregarding extreme values (the result of *extremely* bad initialization of FCM), the average number of case swaps required to find the optimal solution was found to be 4.3. Implementing the FS-FLTS algorithm with $T_1 = 10$ in the phase I (360 FCM runs), the average number of case swaps was found to be $k' = 7.5$ (corrected to nearest higher integer). Terminating swaps after $k' = 8$ case swaps in phase II with $T_2 = 100$, three feasible solutions were obtained with the global solution attracting 75% of the total starting subsets in this phase. In phase II, FCM calculations were performed for a total of 800 times and the CPU usage time was a little less than 2 min.

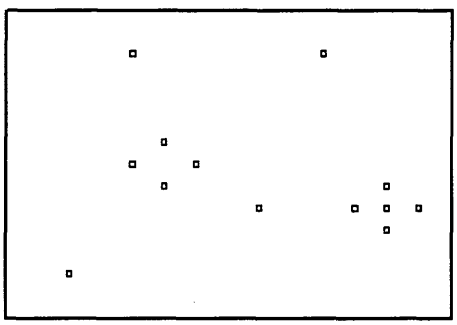


Figure 2.3 The $n = 13$, $h = 9$ synthetic data-set.

The FS-FLTS algorithm was applied to the birth-death data-set [58]. The birth-death data-set is a collection of birth and death percentages of 70 countries and is shown as a scatter plot in Figure 2.4. Three countries are obvious outliers – Denmark with its death rate exceeding the birth rate, and Ghana and Ivory Coast with their abnormally high birth and death rates. Like before, it is assumed that $h = 67$ is known. With 54,740

possible starting subsets, an exhaustive enumeration to locate the exact LTS solution is not feasible. FS-FLTS was applied to the data-set three times to demonstrate the effect varying T_1 and T_2 has on the feasible solution set. Clustering with for $c = 2$ and $c = 4$ is also done separately; however, the trimming results do not appear to be influenced by the choice of c. The results of the three runs are presented in Table 2.4.

Increasing T_1 resulted in a reduction of the average number of case swaps required to reach a feasible solution from a starting subset. However, it would also result in increased FCM computations in phase I. Lesser case swaps would mean lesser FCM computations in phase II. Hence, one needs to strike a balance between T_1 and T_2. Three models were developed – (1) Moderate $T_1 = 10$, and Moderate $T_2 = 60$, (2) Low $T_1 = 5$ and High $T_2 = 100$, and (3) High $T_1 = 15$ and Low $T_2 = 15$. It took approximately 25 min for model 3, 47 min for model 1 and a little over 1⅓ hrs for model 2, which however, produced the best results (only one feasible solution all along). The only unexpected result was obtained from model 3, where 22.8 % of the starting subsets resulted in trimming Ghana, Ivory Coast and Cambodia (not shown), instead of the true outlier Denmark, while only 13.7 % of the starting subsets resulted in the exact LTS solution. It gives credence to the hypothesis that $T_2 \geq 5T_1$ (approximately) for the results to be meaningful.

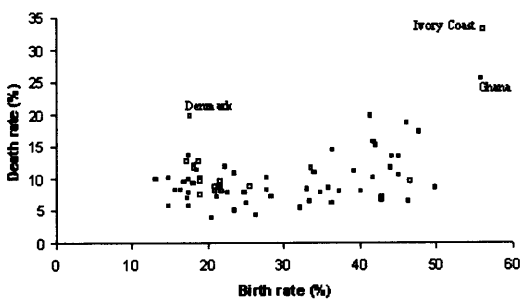


Figure 2.4 The $n = 70$, $h = 67$ birth-death rates data-set.

Table 2.4 FS-FLTS Results for Birth-Death Data

Nos.	Phase I			Phase II			# FCM runs	CPU usage time (min)
	T_1	# feasible solutions	Average case swaps	T_2	# feasible solutions	% of starting subsets for exact LTS		
1	10	1	68.9	60	2	65.9	6210	47.4
2	5	1	121.0	100	1	100.0	13105	102.0
3	15	1	25.8	15	8	13.7	3405	25.9

The hFS-FLTS algorithm is applied to the data-set shown in Figure 2.5. The data-set consists of two normally-distributed clusters, each consisting of 150 two-dimensional vectors with means and variance,

$$\mu_1 = (10,15), \Sigma_1 = \begin{bmatrix} 1 & 0 \\ 0 & 0.25 \end{bmatrix}; \quad \mu_2 = (4,18), \Sigma_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix};$$

which is corrupted by 100 uniformly-distributed noise vectors. In this case it is assumed that there is no prior knowledge of the amount of contamination in the data and hence, is a suitable candidate for the hFS-FLTS. The hFS-FLTS algorithm was implemented with $\Delta H = 0.05$, $c = 2$, $m = 2$ and $\varepsilon = 10^{-5}$. Phase I of the FS-FLTS process was skipped ($T_1 = 0$) and swapping within phase II was terminated after 20 case swaps. In phase II, $T_2 = 20$ was fixed; FCM was performed 400 times for every value of H. In decrements of 0.05, the algorithm is terminated at $H = 0.55$, a total of 10 distinct H values starting from $H = 1$. As a consequence, FCM is performed 4000 times over data-sets of varying size, ranging from $h = 400$ to $h = 220$. Interestingly, at $H = 0.75$ (corresponding to the correctly-retained, $h = 300$), the least number of feasible solutions (only three) were found and the exact solution attracted 14 of the 20 starting subsets (70 %). At all other

values of H, the number of feasible solutions found were more than three, in some cases as many as 12 were found. At values of $H < 0.75$, all the exact solutions trimmed the 100 noise vectors (and some normally distributed vectors too), but in many cases the exact solutions attracted as few as 10 % of the starting subsets.

The *good* behavior of the algorithm at $H = 0.75$ could be taken as a direct indication of the validity of $h = 300$ as the true retention amount. The plot of S_H versus H is shown in Figure 2.6. The plot of $s(H)$ versus H in Figure 2.7 gives a direct proof of the choice of $H = 0.75$ as the true retention ratio. This is a very promising result - the two normally-distributed clusters have been correctly identified and the right amount of trimming (with random starts as low as 20) was correctly estimated. However, skipping phase I of the FS-FLTS and using an arbitrary case swap cut-off value for phase II resulted in a large amount of feasible solutions in that phase.

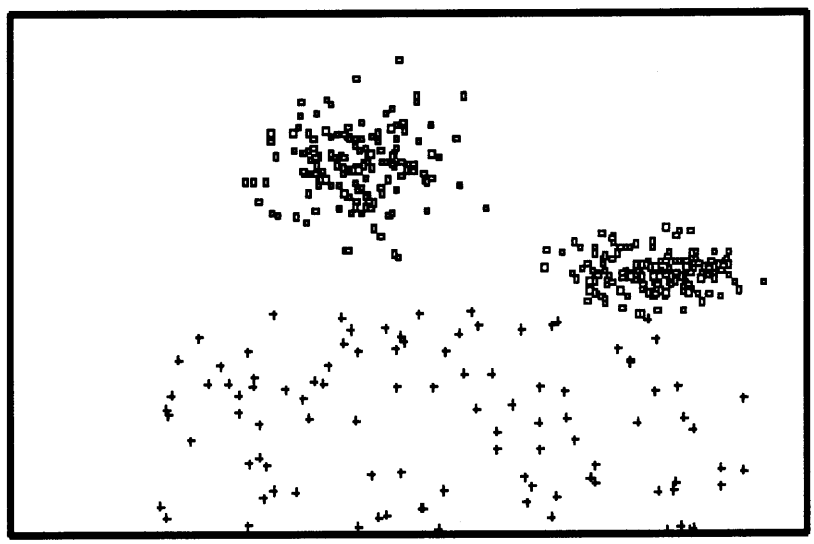


Figure 2.5 The two-cluster normal data-set with uniformly distributed noise, ($n = 400$, $h = 300$).

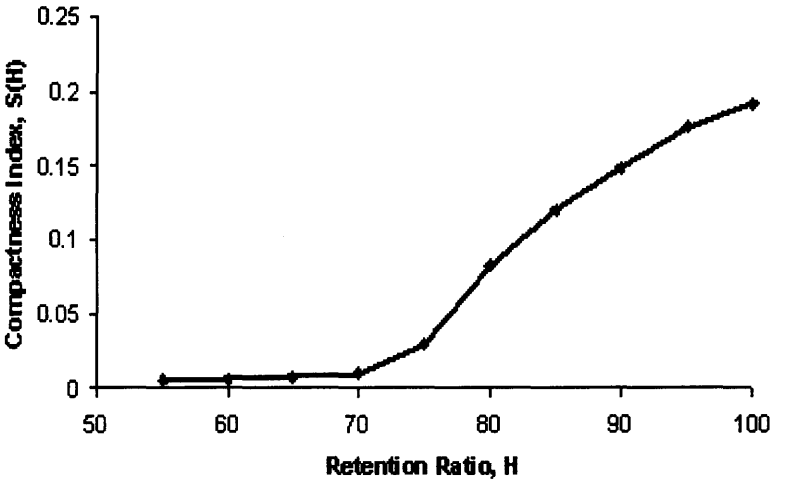


Figure 2.6 Xie-Beni compactness index, $S(H)$ vs. Retention ratio, H.

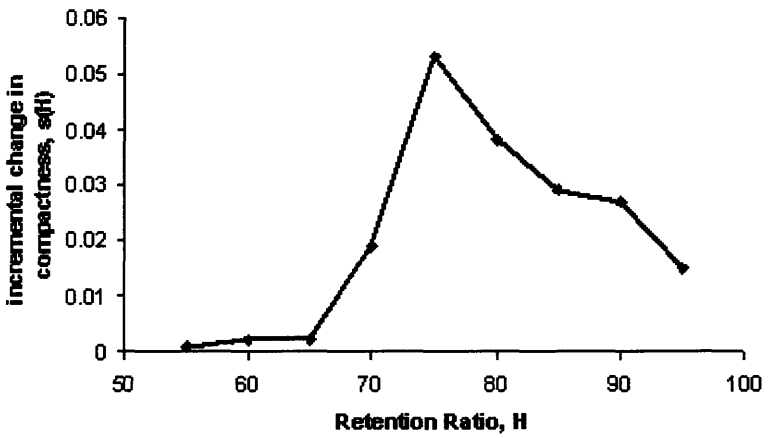


Figure 2.7 Incremental change in the compactness index, $s(H)$ vs. Retention ratio, H. (Note that $s(H)$ is not defined for $H = 1$.)

The MC algorithm was also tested on this data-set. For $c = 2$, $\beta = 1$, and $0.5 \leq \rho \leq 0.0$ (in increments of 0.1), the MC algorithm correctly identified as many as 96 of the 100 noise points as true outliers, while FCM clustered the outliers together with one of the good cluster (the true cluster on the right). The results are shown in Table 2.5. More and more outliers were identified as the criterion for intense scaling became less stringent. In

other words, more good points were identified as outliers. Scaling at $\rho = 0.7$ produces the best results with 96 true outliers identified, with only two false positives (good points identified as outliers). The primary concern here is the identification of good data points as outliers. The percent error is defined as false positives identified per 100 data points (total number of outliers).

Table 2.5 MC Results for the Normal Data-Set

ρ	# of iterations	# of Outliers identified	# of false positives among outliers	% Error
0.9	14	84	5	1.25
0.8	10	92	4	1.00
0.7	9	98	2	0.50
0.6	8	112	12	3.00
0.5	8	119	20	5.00

2.5 Conclusions

Two intuitive and easily-realizable robust clustering schemes have been discussed in this chapter. The concept of a fuzzy mega-cluster which is central to the proposed mega-clustering scheme was also introduced and discussed in detail and shown to be conceptually similar to Davé's noise cluster. While the robust properties of the proposed mega-clustering algorithm were investigated using test cases from the literature, another interesting property of the algorithm was enunciated – the power to distinguish true outliers from non-conformers. The sensitivity of FCM towards noise was reduced by scaling down the memberships and the excess membership was attributed to the mega-cluster. This scaling of memberships in the good clusters was more intense for vectors which are perceived to have an unnaturally large distance from the prototypes and such a definition makes more intuitive sense when the data-set is noisy.

Although the MC algorithm is robust, it still suffers from typical FCM drawbacks such as dependence on fairly good initialization and the tendency to get trapped in local minima. The MC algorithm, like most robust clustering procedures, expects that the approximate amount of contamination in the data-set is known beforehand. For X^A12, where the contamination was less severe (~ 15%), it was found that intense scaling can be done for almost any distance factor ρ, and the results produced are identical. But in the case of X^B14 and the two-normal cluster set (Figure 2.5), where the contamination is almost close to 30%, the results differed when intense scaling was done for different values of ρ. This dependency needs to be further investigated with larger and more natural data-sets.

An FCM-based fuzzy clustering methodology based on the minimization of the least trimmed squares (LTS) functional was also developed and the resultant clustering scheme was shown to be robust, inheriting the high breakdown qualities of the LTS estimator. The feasible solution implementation is also perhaps the only method that ensures sure convergence to the global LTS solution. The algorithm was also modified to a two-phase technique based on a case-swap criterion to minimize computational costs. Tests using data-sets, both synthetic and from the literature, were shown to produce encouraging results. In addition to providing the global solution for the fuzzy LTS partition, the algorithm also generates a variety of distinct *second-best* solutions, each satisfying the condition for an optimum (local in this case).

CHAPTER 3

RANDOM POSITION HYPOTHESIS TESTS FOR CLUSTER VALIDATION

3.1 Introduction

Cluster validation procedures evaluate the results of clustering procedures, such as the ones presented in Chapter 2, in a quantitative and objective fashion [31]. In its entirety however, cluster validity analysis could be used in a wider range of activities, such as,

- Comparison of two or more clustering methodologies on the same data-set with a known number of clusters, to determine the suitability of one clustering technique over the others for the type of application in hand.
- Finding the best partitioning solution among many local-minima solutions (as would be the case with algorithms like FCM).
- Validation of individual cluster structures by measuring and quantifying cluster *goodness* and completeness.
- Validation of the entire partition over a range of cluster values, c, to unravel natural groups in the data, if any.

There are two criteria which seem to be pivotal for clustering evaluation and the selection of an optimal clustering scheme [59] – Compactness and Separation. By compactness, one means that entities of each cluster should be as close to each other as possible, which also happens to be the underlying premise of cluster analysis itself. A common measure of compactness is the variance, which needs to be collectively minimized. Separation means that the clusters themselves should be as widely spaced as possible. The comparison of cluster center distances is one of the ways to measure cluster separation, besides single linkage and complete linkage distances. Validity indices are commonly used to express validity or adequacy in quantitative terms [31].

All cluster validity indices can be broadly classified into,

1. External indices. These indices measure performance of a clustering algorithm by matching a generated partition to *a priori* information e.g. an external validity index measures the degree of correspondence between cluster numbers obtained from a clustering algorithm and category labels assigned a priori. External indices are often used to compare the performance of different clustering schemes on labeled data, especially during supervised clustering.

2. Internal indices. These measure the fit between the structure and the data, using information from the data e.g. an internal index measures the degree to which a partition, obtained from a clustering algorithm, is justified by the given proximity matrix.

3. Relative indices. These comparatively measure the appropriateness of two or more clustering structures produced by the same clustering algorithm. A relative index is used to arrive at the best value of the number of clusters, c to be detected. Information used by a relative index comes from the partition and not from the data.

Statistical measures used to validate clustering results are based on the premise that problems of cluster validity are inherently statistical [31]. A result is usually tested by building an alternative hypothesis and comparing it against a null hypothesis, H_0. The null hypothesis is a statement of randomness and could be based on a random graph, a random label, or a random position hypothesis. An alterative hypothesis, H_a is then a statement of *orderliness* and captures the intent of the phrase – "the data are clustered". The test is then one of comparing H_0 with H_a based on the value of some test statistic T and deciding whether to accept or reject H_0 with a certain degree of certainty. Hubert's Γ statistic [60] and the Goodman-Kruskal γ statistic [61] are well known examples of test

statistics used for cluster validity studies. The Γ statistic compares a clustering structure which is the result of a clustering scheme, to an *a priori* structure, such as one generated by using pre-defined category labels. The γ statistic measures the rank correlation between two ordinal sequences of numbers, one of which might be derived from the clustering structure and the other from an *a priori* labeling scheme. Almost all statistical techniques measure some sort of an external index for clustering. For a detailed discussion on statistical cluster validity statistics, tests and indices, the reader is referred to [31].

Several non-statistical cluster validity indices, mainly relative indices and procedures were independently developed within the fuzzy clustering community to validate partitions obtained using fuzzy clustering algorithms. Prominent among these are the partition coefficient [62], classification entropy [63], proportion exponent [64], the uniform data functional [65], non-fuzziness index [66], information ratio [67], separation ratio [68], and the Xie-Beni index [57]. The simplest of these is the partition coefficient which describes the fuzziness of the partition. It is inversely proportional to the average fuzzy overlap between the clusters, and is given by

$$F = \frac{1}{n} \sum_{i=1}^{c} \sum_{k=1}^{n} u_{ik}^2 . \tag{3.1}$$

$F = 1$ indicates no overlap between clusters and is the case when FCM degenerates to hard c-means (K-means). On the other hand, $F \rightarrow 1/c$ is the extreme fuzzy case when all the entities are shared equally between all the clusters. Hence, the partition coefficient can take values between $1/c \leq F \leq 1$. Normalizing F as shown in Equation (3.2) compensates for this dependence on c.

$$F' = \frac{cF - 1}{c - 1}. \tag{3.2}$$

A high value of F (and F') indicate a better partition, where clusters are compact and well separated, as compared to a low value, which indicates almost equal sharing of all entities among all the clusters. The application of Shannon's entropy [69] to fuzzy clustering resulted in another cluster validity measure known as the partition entropy, given by

$$H = -\frac{1}{n}\sum_{i=1}^{c}\sum_{k=1}^{n} u_{ik} \ln u_{ik} . \tag{3.3}$$

A good partition is characterized by a low value of H; it can take values between $0 \leq H \leq \ln c$. Since H varies with $\ln c$, the monotonically decreasing tendency of H with c is not as severe as in the case of F, and hence normalizing H has little beneficial effect.

Both the partition coefficient and entropy measure the amount of fuzziness from cluster membership information and do not consider geometric properties such as size, shape, and compactness of the clusters. Dunn's separation index, also known as the CS index [70], [71], identifies unique cluster structure with well-defined properties that depend on the data and a measure of distance. It provides information about the separation and compactness of the clusters, but is computationally unfeasible to apply to large data-sets since a distance matrix between all the data membership values has to be calculated. It also works on a hard c-partition derived from the fuzzy partition. The Fukuyama-Sugeno index of cluster validity [72] consists of the difference of two terms - the first term combines the fuzziness in the membership matrix with the geometrical compactness of the representation of the data-set via the prototypes, and the second term combines the fuzziness in a row of the partition matrix with the distance from the i^{th} prototype to the grand mean of the data. The minimum of this index over a range of c

values indicates the best partition. Gath and Geva [73] proposed using fuzzy volume and fuzzy density of the clusters as a cluster validity criteria; a good cluster is characterized by a high value of fuzzy partition density and an accompanying low value of fuzzy hypervolume. The compactness criterion of Xie and Beni [57] considers cluster compactness and separation as a measure of cluster validity. This criterion is also sometimes referred to as the Xie-Beni index and is given by

$$S = \frac{\sum_{i=1}^{c}\sum_{k=1}^{n} u_{ik}^2 d_{ik}^2}{n\{\min_{i,j=1,..,c} \|v_i - v_j\|^2\}}. \tag{3.4}$$

While the numerator describes the compactness of clusters in the partition, the factor in the denominator describes the separation of the clusters. A low value of S indicates a good partition.

Most indices work directly on the fuzzy clustering outputs but a few of them first convert the results to a hard c-partition before evaluating it. Specialized partitioning schemes such as shell partitioning require the use of specialized validity indices such as the partition density and the shell thickness measure [74]. More recently advances have been made in visual assessment of clustering using intensity displays [75], [76]. Some indices are known to function poorly across a wide range of data-sets while others are specifically suited to a particular type of data-set. In other words their dependence on the type of data-sets, and on the type of clustering scheme employed, seriously hinder the practical usage of fuzzy validity indices.

In the next few sections, a novel cluster validity technique is presented. A statistical concept from the field of clustering tendency is borrowed and its applicability to validate clustering results generated in a partitioned data is shown.

3.2 Sparse Sampling Tests and the Hopkins Statistic

The problem of testing for clustering tendency can also be described as the problem of testing for spatial randomness. Unlike statistic-based cluster validity measures, a test for clustering tendency is stated in terms of an internal criterion and no *a priori* information is brought into the analysis [31]. The null hypothesis in most cases is a random position hypothesis, such as,

> H_0: The patterns are generated by a Poisson process with an intensity of L patterns per unit volume.

Under H_0, the number of patterns falling in a region of volume V has a Poisson distribution with mean LV and since L is constant and the numbers of patterns falling in disjoint regions of V are independent random variables, the Poisson process is a reasonable model for randomness (absence of structure). Sparse sampling tests have been shown to have high power against clustered alternative hypotheses. On the other hand, tests based on small interpattern distances (such as nearest neighbor distance tests) have low power against clustered alternatives primarily because such tests depend heavily on the intensity L of the Poisson process assumed under H_0. Other tests for spatial randomness include Scan tests [77], Quadrat analysis [78], and Second moment structure tests [79].

Sparse sampling tests are based on sampling origins randomly identified in a sampling window. Several tests involving sampling origins have been proposed in the literature, based on a multitude of test statistics such as the Hopkins [80], Holgate [81], [82], T-square [83], Eberhardt [84], and the Cox-Lewis statistic [85]. These statistics have been compared but it has not been categorically shown that any one of them

outperforms the others. The Hopkins statistic is easy to comprehend and has been shown to be as good as the Holgate statistic [86].

Let $X = \{x_i \mid 1 \leq i \leq n\}$ be a collection of n patterns in a d-dimensional space such that $x_i = \{x_{i1}, x_{i2}, .., x_{id}\}$. Also, let $Y = \{y_j \mid 1 \leq j \leq r\}$ be r sampling origins placed at random in the d-dimensioned sampling window, $r << n$. A sampling window can be thought of as a subspace of the entire d-dimensioned sample space. Two types of distances are defined: u_j as the minimum distance from y_j to its nearest pattern in X and w_j as the minimum distance from a randomly-selected pattern in X to its nearest neighbor (r out of the available n patterns are marked at random for this purpose). The Hopkins statistic in d-dimensions is defined as,

$$HS = \frac{\sum_{j=1}^{r} u_j^d}{\sum_{j=1}^{r} u_j^d + \sum_{j=1}^{r} w_j^d}. \tag{3.5}$$

This statistic compares the nearest-neighbor distribution of randomly-selected locations to that for the randomly-selected patterns. Under the null hypothesis, H_0, the distances from the sampling origins to their nearest patterns should, on the average, be the same as the interpattern nearest neighbor distances, implying randomness and hence *HS* should be about 0.5. However, when the patterns are aggregated or clustered into clouds, the sampling origin to pattern nearest neighbor distances should, on the average, be larger than the randomly-selected interpattern nearest neighbor distances. In other words, *HS* should be larger than 0.5; almost equal to 1.0 for very well defined clustered data. By the same reasoning, *HS* is supposed to be much less than 0.5 for regularly-spaced data, data that are neither clustered nor random. To ensure that no pattern is the neighbor of more than one sampling origin, r is chosen to be substantially less than n; it

has been suggested that $r < 0.1n$ [79]. With such a condition, it can be ensured that all $2r$ nearest neighbor distances are statistically independent and *HS* has a beta distribution with parameters (r, r), independent of both the intensity L, and the dimensionality of the data-set d. The distribution and the density function of each of the terms in Equation (3.5) are also known; the individual sums each have a gamma distribution (assuming that the nearest neighbor distances are all independent random variables). Studies done on random data-sets, clustered data-sets, and regularly-spaced data-sets show that *HS* consistently has a value of around 0.5, 0.7-0.99, and 0.01-0.3 respectively, and is hence a powerful estimator of randomness.

3.3 Random Position Hypothesis Tests

A natural cluster is *unusually* compact and *unusually* isolated [31]. A clustered data-set is ordered because of the presence of natural clusters; in the absence of natural groups, it is a random collection of data points, approximating a Poisson process distribution. In this section the applicability of the random position hypothesis is shown and the Hopkins statistic of Equation (3.5) is used as a measure for cluster validity. Suppose a data-set with 3 compact and isolated clusters, as shown in Figure 3.1, is subject to partitioning. At $c = 2$, most partitioning schemes would club clusters II and III together as one cluster, as cluster *A,* and identify cluster I as an independent cluster, *B*. A random position hypothesis test would lead to the rejection of H_0 for cluster *A* because it still is a collection of clustered data points. However, it would be difficult to reject H_0 for cluster *B* because it is the natural cluster, cluster I. *A natural cluster hence apart from being isolated and compact is also random within itself.* Intracluster data might also exhibit

some kind of mutual repulsion as in Figure 3.1, and in such a case the null hypothesis H_0 should include a statement conforming to random position as well as a non-random non-clustered (regularly-spaced) distribution. In such a case, the null hypothesis cannot be rejected if the data is either random or regularly spaced. However, in real world clustering problems, this is rarely the case and so a random position hypothesis alone should suffice. At $c = 3$ however, all the three natural clusters are most likely to be identified during partition and hence it would be difficult to reject H_0 for all the three clusters identified. The rejection or acceptance of H_0 depends on the value of the Hopkins statistic. At any value of $c > 3$, any partitioning algorithm would either subdivide or recombine the clusters that were produced by partitioning at $c = 3$, and hence one might not have reason to reject H_0 for any of the clusters (in case clusters are subdivided further) or in some cases just enough evidence to reject H_0 (in case clusters are recombined) for some of the generated clusters.

The test for cluster validity based on the random position hypothesis can hence be stated as follows – Accept the lowest value of c at which it is impossible to reject the null hypothesis H_0 for all the clusters, the test applied one cluster at a time. Let HS_i be the value of the Hopkins statistic for the i^{th} cluster at a particular level of clustering c, the average value of the statistic is given by HS_μ and the variance by HS_v,

$$HS_\mu = \frac{1}{c}\sum_{i=1}^{c} HS_i \,, \tag{3.6}$$

$$HS_v = \frac{1}{c}\sum_{i=1}^{c} (HS_i - HS_\mu)^2 \,. \tag{3.7}$$

A non-rejection of H_0, as given in the last section, would mean that on an average, the value of HS_μ is very close to 0.5, and the value of HS_v is close to zero. Proceeding from $c = 2$ to $c = n - 1$ (or any suitably chosen cutoff value), the lowest value of c pertaining to $HS_\mu \approx 0.5$ and $HS_v \approx 0$, most likely generates a partition that identifies the natural clusters in the data.

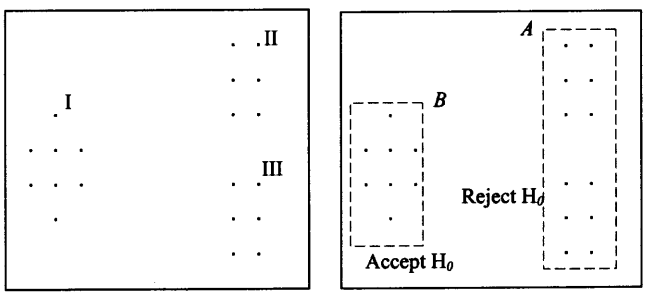


Figure 3.1 A three cluster data-set, (a) the three natural clusters, (b) clusters A and B identified at $c = 2$.

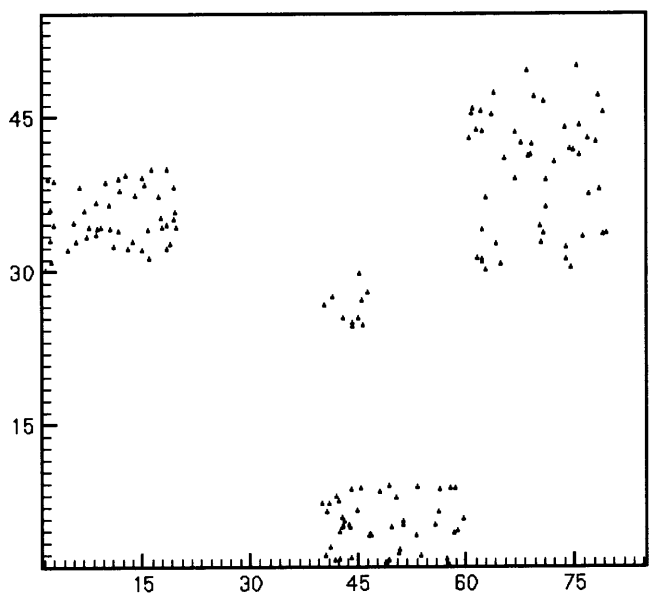


Figure 3.2 The 160 pattern four-cloud data-set (three clusters of 50 patterns each and one cluster of 10 patterns).

3.4 Simulation and Results

In this section cluster validity studies done on two artificially produced 2-D data-sets are presented – (1) four-cloud data (160 patterns), and (2) seven-cloud data (1400 patterns). The patterns are generated within a pre-specified window using the C++ *rand*( ) function, which produces pseudo-random numbers using a seed initialized by the CPU clock time. The data-sets are partitioned using FCM, ranging from $c = 2$ to $c = 8$ for the four-cloud data-set, and from $c = 2$ to $c = 11$ for the seven-cloud set.

The fuzzy partitions are first converted into hard partitions and then all the generated clusters are subject to the random position hypothesis test. The sampling window in all cases encompasses the entire cluster set and the number of sampling origins; r is chosen to be $n/10$ (or the closest integer value), where n is the number of patterns in the cluster being investigated. In case there were less than 10 patterns assigned to a cluster, $r = 1$. The average partition Hopkins statistic and the statistic variance are then calculated using Equations (3.6), and (3.7), respectively, and the results plotted against c.

The four-cloud data is shown in Figure 3.2. The resultant average Hopkins statistic HS_μ, and the resultant variance of the Hopkins statistic HS_v, are plotted against the number of clusters, c (shown in Figure 3.3). As can be seen, the null hypothesis cannot be accepted for $c = 2$ ($HS_\mu = 0.76$), and $c = 3$ ($HS_\mu = 0.64$). However, it can be accepted with a fair degree of confidence for $c = 4$ ($HS_\mu = 0.47$, $HS_v = 4 \times 10^{-4}$) and thenceforth. Hence, according to the cluster validity criterion enunciated in the previous section, one can accept $c = 4$ as the partition identifying the natural groupings in the data, which is indeed the case.

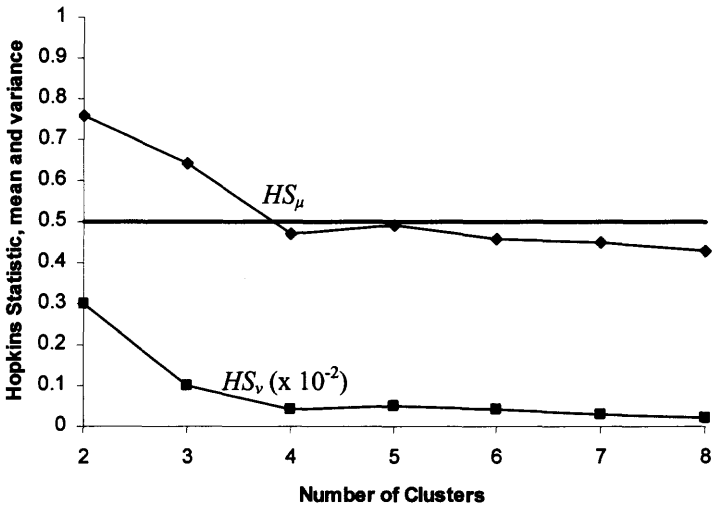


Figure 3.3 HS_μ and HS_v plotted against c ($2 \leq c \leq 8$), for the four-cloud data-set.

The seven-cloud data-set, as shown in Figure 3.4, is different from the four-cloud data in that the former is not as well separated as the latter. The applicability of the Hopkins statistic as a cluster validity index and the random position hypothesis test as an appropriate test for cluster validity, are illustrated in a much broader sense in this case. The two clusters in the lower left hand corner of Figure 3.4 overlap each other and can be argued to be one big tilted *8*-shaped cluster. This can be seen from the plot of HS_μ and HS_v versus the number of clusters in Figure 3.5; it is difficult to chose between $c = 6$ ($HS_\mu = 0.53$, $HS_v = 10^{-4}$), and $c = 7$ ($HS_\mu = 0.48$, $HS_v = 10^{-4}$). Other values of c can be rejected outright. Hence, the Hopkins statistic does reflect the nuances and subtleties in the data-set. In the absence of the overlap, HS_μ and HS_v would have indicated a clear preference for $c = 7$, suggesting natural grouping at that level of partitioning.

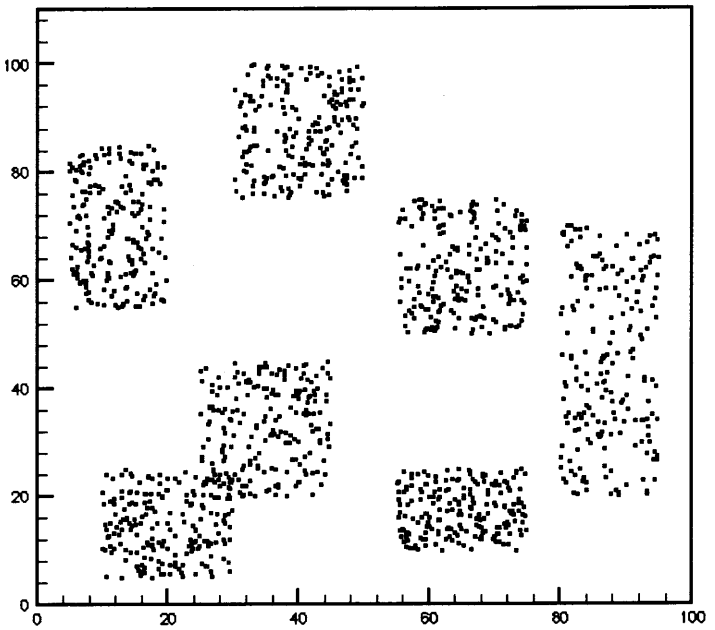


Figure 3.4 The 1400 pattern seven-cloud data-set (200 patterns per cluster).

In case the clusters are not as well-separated as they are in these two test cases, one might need to accept or reject H_0 for individual clusters one at a time, instead of making a decision based on HS_μ and HS_v. As in the four-cloud data-set, one can reject H_0 for at least one generated cluster for both $c = 2$, and $c = 3$ partitions. However, at $c = 4$, H_0 cannot be rejected for any of the four generated clusters. For the seven-cloud data-set, one can reject H_0 for at least one cluster in the range $2 \leq c \leq 5$. However, it becomes difficult to reject H_0 for any of the clusters generated at $c = 6$, and $c = 7$. If there were no overlap of clusters in the lower left-hand corner, one could have outrightly rejected H_0 for at least one cluster in the range $2 \leq c \leq 6$, and $c = 7$ would have been the smallest value of c where one cannot reject H_0 for any of the seven clusters.

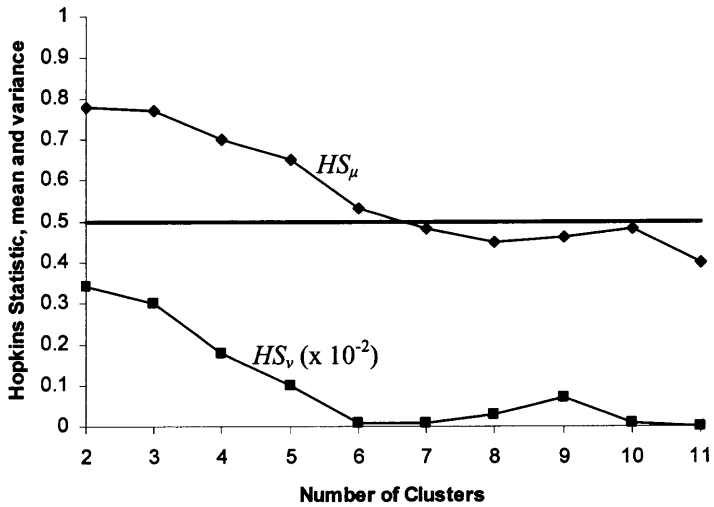


Figure 3.5 HS_μ and HS_v plotted against c ($2 \leq c \leq 11$) for the seven-cloud data-set.

Apart from these synthetically-developed data-sets used to demonstrate the applicability of the concept, the random position hypothesis test was also used to validate clusters produced in three molecular conformational data-sets. These data-sets are presented in detail in Chapter 4. The three data-sets are designated DS-1, DS-2, and DS-3 and consist of a set of 728 patterns in 3-D space (n = 728, p = 3). DS-1 and DS-2 are shown in Figures 4.16(a) and (b), respectively and DS-3 is shown in Figure 4.13(a). The cluster validity studies of these data-sets, using indices from literature, is presented in Chapter 4, but are reproduced here for the sake of clarity and comparison. In the figures presented in Chapter 4, the clusters are color-coded according to the c-partition results produced by a fuzzy relational clustering algorithm. The same clustering results are used here. The four validation indices used here from literature are the ones shown in Equations (3.1) – (3.4). The Xie-Beni index of (3.4) had been slightly modified to be used with relational data, and is subsequently shown in Equation (4.8). The data-sets are

partitioned from c = 2 till c = 14, and the results for DS-1 and DS-2 are presented in Figures 3.6 – 3.9. As can be seen, the results of the proposed validation scheme agree with the results of the four indices from literature.

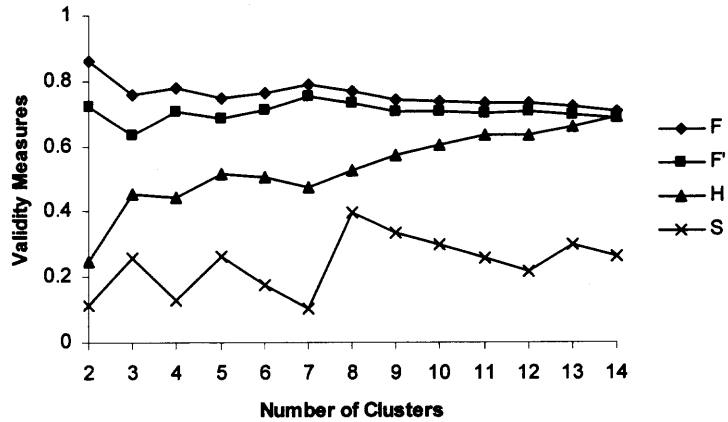


Figure 3.6 The four validity indices plotted for the DS-1, see Figure 4.16(a). The results indicate $c = 7$ as the best partition, followed by $c = 4$.

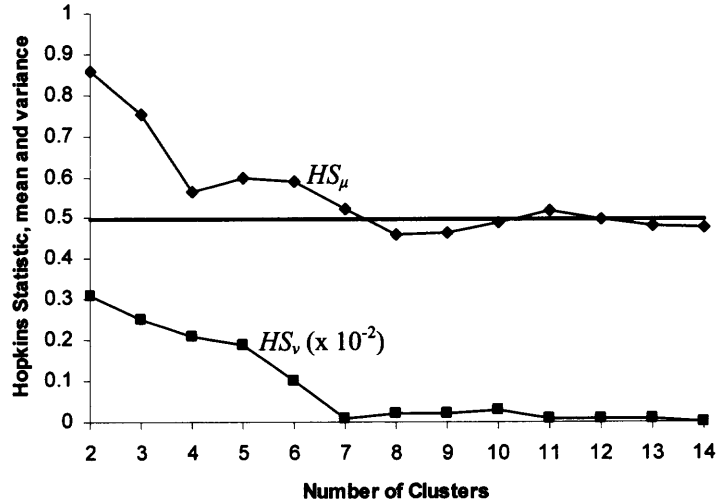


Figure 3.7 The mean and variance for the Hopkins statistic for DS-1. The results indicate good partition at $c = 7$, in agreement with Figure 3.6.

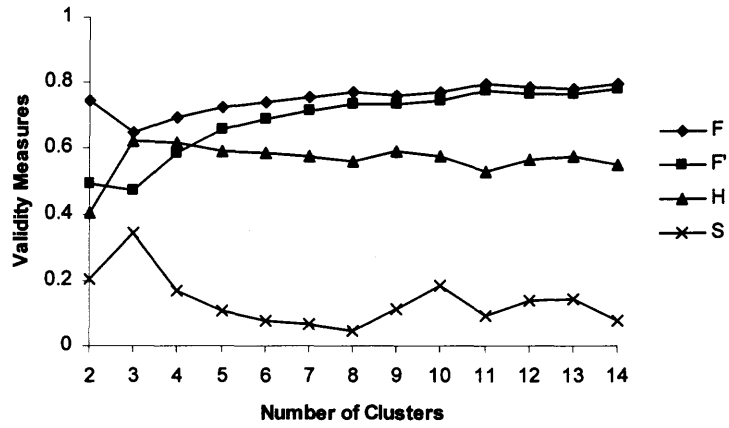


Figure 3.8 The four validity indices plotted for the DS-2, see Figure 4.16(b). The results indicate $c = 8$ as the best partition.

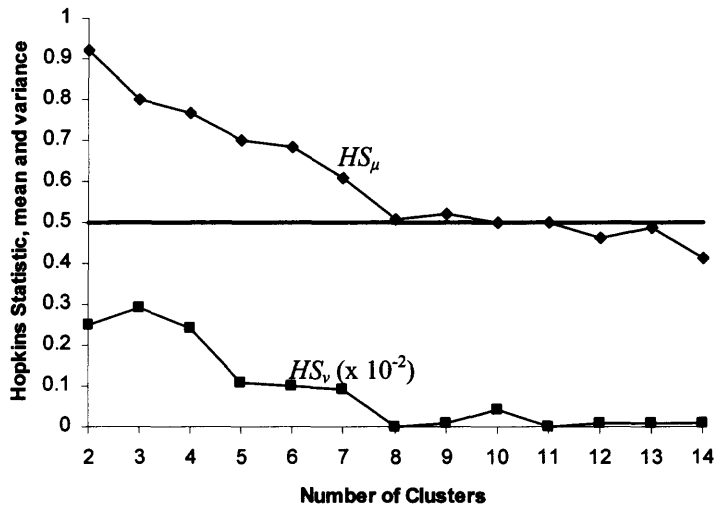


Figure 3.9 The mean and variance for the Hopkins statistic for DS-2. The results indicate good partition at $c = 8$, in agreement with Figure 3.8.

However, a visual inspection of the DS-3 data-set suggests that there are no natural clusters to be found. The data is a dispersed random blob of 3-D points. Relational clustering is carried out for $c = 2$ till $c = 14$, but the validation indices are plotted from $c = 1$ till $c = 14$. At $c = 1$, $F = 1$, $H = 0$, while F', and S are not defined. The results are shown in Figures 3.10 and 3.11. The absence of natural groups in DS-3 is

indicated in Figure 3.11 – the average Hopkins statistic is always around 0.5 for every partition. This however, is not readily apparent from the behavior of the other indices (Figure 3.10).

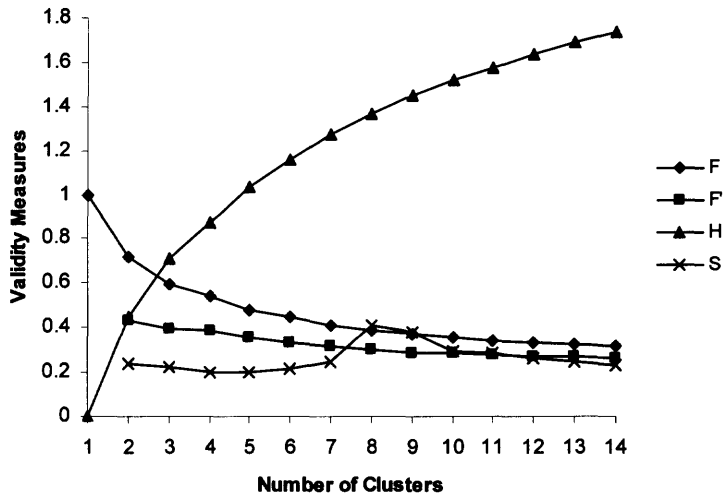


Figure 3.10 The four validity indices plotted for the DS-3, see Figure 4.13(a). The results indicate $c = 5$ as the best partition among the clustered options. However, visual inspection reveals no existence of substructure.

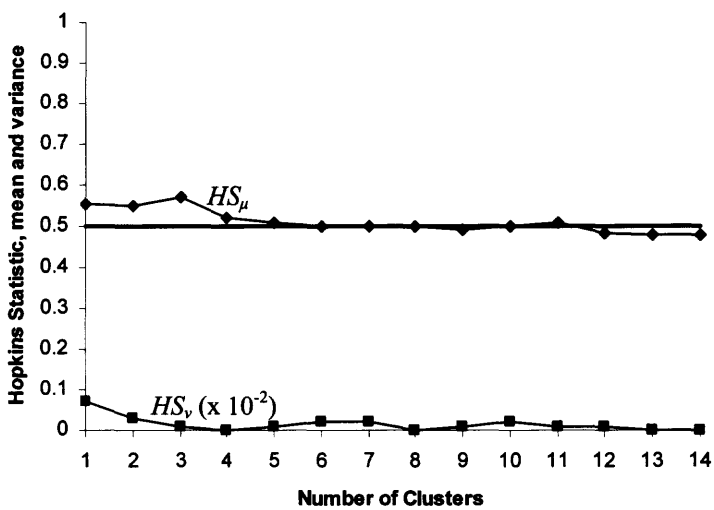


Figure 3.11 The mean and variance for the Hopkins statistic for DS-3. The results indicate randomness (no apparent substructure, and hence, the absence of natural groups) in the range $1 \leq c \leq 14$ ($HS_\mu \approx 0.5$, and $HS_v \approx 0$).

3.5 Conclusions

The applicability of the random position hypothesis test as a criterion for cluster validity is demonstrated, using one of the well known test statistics, the Hopkins statistic as an index for the test. The random position hypothesis and the Hopkins statistic have been used previously in the context of clustering tendency. Showing that with virtually no change in form, the same could be applied to testing for cluster validity, is the novel idea. The scheme has been validated on test cases, both small (160 patterns) and big (1400 patterns). Generation of sampling windows and sampling origins, location of random patterns to generate nearest-neighbor distances and the associated calculations are computationally inexpensive on powerful desktop PCs of today. The program to generate fuzzy partitions using FCM, defuzzify the results, and then test each partition for randomness was written in C++, compiled using a visual C++ compiler running on a windows PC environment.

The applicability of the Hopkins statistic and the random position hypothesis test for cluster validity to non-cloud data (such as detection of lines in a data-set) needs to be investigated. The key in applying the theory to linearly-clustered data might rest on an appropriate selection of the sampling window. A skewed line cluster within a rectangular sampling window will appear to be a clustered (non-random) collection of data because a line cluster is neither an isolated nor a compact cluster. It is also impossible to apply the Hopkins statistic to very small data-sets where each cluster might contain just 4-5 patterns each. At such intensities of pattern distribution, the theory of randomness does not hold and hence the random position hypothesis is meaningless. However, this is a blessing in disguise because very small data-sets are rarely encountered in real world

situations; in fact it could be claimed that the random position test for cluster validity produces better results as the data-set gets larger in size. The data should be *isolated* in groups and *compact* within the groups, is perhaps the only restriction to the applicability of the Hopkins Statistic. The theory can easily be extended to data in more than two-dimensions because the Hopkins statistic is essentially defined in d-dimensions. One could even use other statistics, such as the Cox-Lewis statistic which extends better in d-dimensions ($d > 2$) than the Hopkins statistic. The random position hypothesis test for cluster validity using an index such as the Hopkins statistic not only provides an answer to the ever-elusive question – "*how many clusters to find?*" but also provides a validation measure for individual clusters. If H_0 cannot be rejected for any of the clusters at the lowest $c = c^*$ partitioning, then one could argue that all the clusters found are in fact the true natural clusters in the data. Not intended to replace the existing cluster validity techniques and indices, the test for random hypothesis is an interesting and promising addition to the repertoire of cluster validity methodologies.

CHAPTER 4

DIMENSIONALITY REDUCTION AND CLUSTERING APPLIED TO COMPUTATIONAL CHEMISTRY

4.1 Introduction

Feature selection and feature extraction are techniques that aim to reduce the feature space for computational reasons, cost considerations, or other technical reasons [87]. The reduced feature space is expected to have only a set of highly predicate features. Feature selection is of utmost importance to fields such as pattern recognition, data mining, image processing, and computational chemistry. Feature selection is also directly related to the *curse of dimensionality* [88]. This rather emotive term was initially used to describe the difficulties associated with statistical density estimation in higher dimensions. However, it is well-known fact that computational costs grow exponentially as dimensions of a system increase. Appropriate feature selection also provides a reduction in feature space dimensions, and reduces associated system complexity.

All feature selection algorithms have two key elements. One is the measure of the quality of the features, and the other is a search strategy to find the best feature subset as defined by the measure. However, what most automated feature selection methods fail to recognize is that feature selection or feature extraction is heavily application-dependent and it serves no practical purpose by having a *fit-all* technique by generalizing the process. In this work, unique feature extraction, and dimensionality reduction techniques are developed for clustering a large data base of molecular conformations, and the conformations are clustered on features in the reduced dimensional space using a recently developed relational clustering scheme.

Previous attempts at feature selection have focused mainly on statistical approaches such as the typical Principal Component Analysis (PCA) method [89], and the Linear Discriminant Analysis (LDA) method [90]. These methods attempt to reduce the dimensionality of the feature space by creating new features that are linear combinations of the original ones. The new features in many cases have no real physical interpretation, and hence no true meaning. Other statistical techniques include metric and multidimensional scaling techniques [91]. Blum and Langley [92] have published an excellent survey for relevant feature selection for machine learning tasks. Almost all of the approaches use some kind of a quantitative evaluation criterion, such as gain-entropy [93], relevance [94], or contingency table analysis [95], to be used on feature sets that are real, symbolic, mixed, nominal, or categorical. It has also been shown that appropriate features can also be selected using genetic algorithms [96], [97] where each feature subset is evaluated by a fitness function during an optimization cycle and have been shown to produce a number of optimal feature sets.

High dimensional data-sets are often encountered in conformational analysis of molecules for computational chemistry applications. Conformations are families of molecules that have the same molecular structure but differ in their spatial orientation. In ligand-based drug design, the bioactive conformation of a promising drug molecule is defined as the conformation in which the drug binds to the protein receptor. In the absence of structural information about the receptor, the prediction of the bioactive conformation is a challenge. Conformation searching techniques are used to explore the conformational space of a ligand to generate stable conformations with low potential energies. However, if the molecule is flexible (spatially), the number of conformations

generated is very large. This prohibits the consideration of every low energy conformation as a putative bioactive conformation. This provides the motivation for reducing the conformational space by selecting a suitable set of representative conformers using techniques such as clustering. These representative structures are then further analyzed to relate 3-D structure with properties and activity (the relationship is known as 3-D Quantitative Structure-Activity Relationship or 3-D QSAR).

Attempts to cluster conformations have been based on some sort of proximity measure between pairs of conformers. The popular clustering techniques for generation of representative conformers are hierarchical techniques such as the single-link clustering, and the average-link clustering schemes. The single-link clustering package, XCluster [98] clusters conformations based on a root mean square (RMS) distance matrix derived either from a set of atom coordinates (with or without rigid body superposition of conformations), or from a set of torsional angles. An average-link clustering technique has been used to demonstrate clustering of 63 conformations of a tripeptide fragment based on a Euclidean distance measure of proximity in a 36-dimensional space [99]. However, as data-sets become larger, techniques based on dendograms are impractical with more than a few hundred patterns [31]. Another problem is that such techniques may tend to find singleton clusters unless carefully selected termination criteria are utilized. Non-hierarchical techniques used in conformational clustering include a variant of the Nearest Neighbor (NN) based scheme. The technique, called the nearest single neighbor method [100], was used to cluster different sets of peptide conformations and is based on a proximity measure derived from RMS distances between pairs of conformations by considering only the peptide backbone structure. While NN techniques

have been shown to be useful, they tend to be computationally expensive for large data-sets.

More recently, attempts have been made to cluster families of conformations using statistical scaling techniques as cluster analysis tools. In the first of a related set of papers, families of relatively small or rigid molecules such as dopamine, roseotoxin-B, and, cycloheptadecane were clustered by first scaling the higher dimensional data in real space to a reduced 3-D conformational space using both multidimensional and metric scaling techniques [101], [102]. Then either visual inspection, or a hierarchical technique applied to a proximity matrix derived from the reduced 3-D data-set, was used to complete the clustering [101]. Subsequently, the same 3-D data-set was clustered using fuzzy clustering [102]. This is the only instance where a partitional scheme has been successfully applied to cluster families of conformations.

4.2 Feature Extraction of DM 324 Conformers

This work presents feature extraction and clustering studies on conformations of a GBR 12909 analogue. The molecular structure of GBR 12909 is shown in Figure 4.1. The analogues of GBR 12909 belong to a class of dopamine reuptake inhibitors that might be potentially useful in the treatment of cocaine abuse [103]. One analogue, DM 324, is shown in Figure 4.2. The purpose of cluster analysis of the conformations of DM 324 is to identify a small number of structurally dissimilar conformations that could aid in understanding the interaction between the molecule and the dopamine transporter.

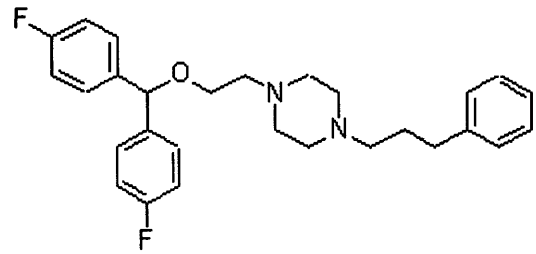


Figure 4.1 Molecular structure of the GBR 12909 molecule.

This section describes a novel feature extraction technique designed for selecting the best features for clustering from the family of 728 DM 324 conformers, generated by random search of the conformational space using the SYBYL molecular modeling package. The 728 conformations are described by a unique set of heavy atom locations in space. All atoms excluding hydrogen (not shown in Figure 4.1) constitute the set of heavy atoms for the molecule; for DM 324, there are 35 heavy atoms in 3-D space which constitutes a raw feature set. In addition to heavy atom locations, certain other features are also potentially useful – location of certain planes on which the rings and chains lie. There are four ring structures in the molecule and these are described by the four planes shown in Figure 4.2. Any three atoms on a chain can also describe a potentially useful feature plane. Two distinct feature extraction methodologies with respect to the GBR 12909 are described here.

4.2.1 The Minimal Feature Set

Two motivations guided the feature extraction process in this case – reduction of the feature space, and handling of the symmetry of each phenyl ring. In general, reduced dimensionality of a large input data matrix is desirable for more easily-interpretable results. Moreover, some features are redundant and retaining redundant data not only

makes the feature space high dimensional and cluttered yet sparse, but also usually has periodicity that makes data classification and interpretation very difficult. These redundancies are eliminated by considering only the features essential to completely describe a molecular conformation in its spatial configurations. The set of features defined by such essential features is henceforth called the *minimal feature set.*

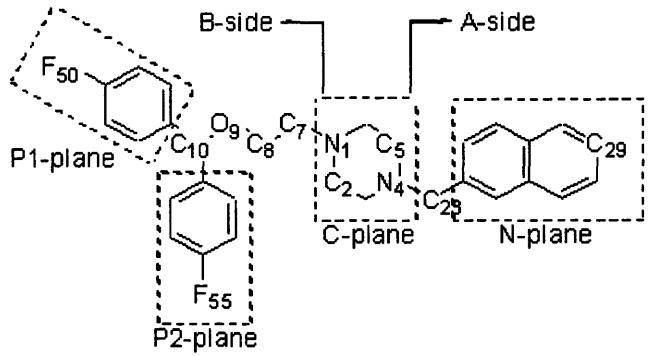


Figure 4.2 Detailed structure of DM 324 showing the four planes considered for the minimal features set.

The problem of phenyl ring symmetry was handled by using molecular planes as part of the feature set. Each of the two phenyl rings (the rings on planes P1 and P2) contains symmetry-equivalent atoms that have different atom labels. For example, the 2-position carbon in the P1-plane in Figure 4.2 is atom number C_{12}, whereas the 2′-position carbon is atom number C_{16}. Rotation of the phenyl ring of the P1-plane by 180° gives a molecular structure which is indistinguishable from the previous one, yet the labeled atoms are in different positions. Superposition of these two structures would show a perfect fit, yet calculations that are based on atom labels, such as the RMS distance between atoms, would show a large difference. Considering only the planes (or a relation

between them) on which the phenyl rings lie provides an atom-label-independent solution to the symmetry problem. This atom-label-independent description of the phenyl rings was achieved by using plane equations, which specify the planar orientations of the phenyl rings, and selected atomic coordinates (of the two fluorine atoms), which specify their exact location.

In order to aid in the feature extraction process, the molecule was conceptually divided as described below into regions containing the two pharmacophore elements – the N_4 in close proximity to an aromatic ring (the naphthalene ring on the N-plane in Figure 4.2), and the bisphenyl group (which has been shown to be necessary for good binding affinity). Two different types of superpositions were applied to the data-set of molecular conformations and different *minimal features sets* were identified for each superposition. In this way the effect of clustering the conformations using a feature set defined for the molecule as a whole versus using feature sets defined for various fragments could be compared. The superpositions and the related feature vectors are summarized in Table 4.1, and are described below.

1) Superposition 1: The data-set of molecular conformations was superimposed by a rigid body superposition using atoms N_1, C_2, N_4, and C_5 in the piperazine ring (on the C-plane). The C-plane was fixed in the $y = 0$ plane for all structures. The molecules were translated and/or rotated in space so that N_1 was at the origin of the coordinate system, and the locations of C_2, N_4, and C_5 coincided for all the conformers. The molecule was divided into A- and B-sides around the C-plane as shown in Figure 4.2. The A-side and N_4 contain the DAT inhibitor pharmacophore elements. The B-side contains the bisphenyl group. If the features were defined by the Cartesian coordinates of each heavy

atom, the dimensionality of the resulting feature space would be 35 x 3 = 105. However, since the six heavy atoms in the ring have the same coordinates in every conformer in Superposition 1, they can be excluded from the coordinate data matrix. This results in a feature space of size 29 x 3 = 87, which is still quite large. Three different feature vectors were constructed using the novel feature extraction method described below in order to further reduce the size of the feature space and to compare the effects of clustering on the full molecule versus the A-side or the B-side.

Examination of the molecule indicates that the A-side of the molecule can be completely reconstructed using two sets of atom coordinates and one plane equation. The reconstruction sequence for the A-side, using coordinates of atoms C_{23} and C_{29}, and the plane equation for the N-plane, is illustrated in Figure 4.3. It should be noted that all atom coordinates and plane angles are calculated after the rigid body superposition. Starting with the known position in space of a single atom, C_{29}, it is possible to use bond length and bond angle information to construct the rest of the naphthalene fragment in the plane specified by the known plane equation of the N-plane. Once an arbitrary orientation of the naphthalene fragment is obtained, it is rotated about C_{29} within the N-plane such that C_{23}', the arbitrary location of C_{23}, coincides with the true known coordinates of C_{23}. The resulting fragment fully specifies the A-side. The coordinates of atoms C_{23} and C_{29} and the plane equation for the N-plane form the minimal feature set for the A-side because these features contain the minimum information needed to completely specify the A-side of each conformation. The A-side feature vector used as the input to construct the proximity matrix was derived from the minimal feature set and consists of coordinates of C_{23} and C_{29}, and the angle between the N-plane and C-plane, as

summarized in Table 4.1. Since the C-plane is fixed in the y = 0 plane for all conformations, it is excluded from the definition of the minimal feature set and only the equation for the N-plane need be included. The two atoms and the two planes that define the angle between planes are labeled in Figure 4.1. The dimensionality of the feature space for A-side-only clustering is thus reduced to [2 x 3 coordinates + 1 angle] = 7.

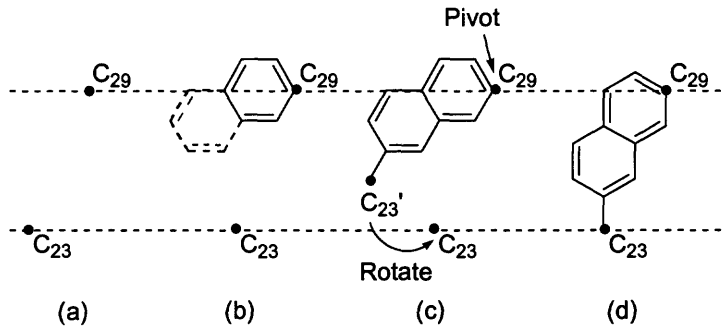


Figure 4.3 Reconstruction sequence for the A-side. (a) Exact locations of C_{23} and C_{29} are known and fixed on the N-plane. (b) Construction of the naphthalene fragment is possible using information about bond lengths and bond angles to obtain an arbitrary orientation of the fragment. (c) C_{23}' is the arbitrary position of C_{23} obtained after construction of the naphthalene ring. (d) Rotation about C_{29} so that C_{23}' coincides with C_{23} gives the true orientation of the ring on the N-plane.

Table 4.1 Feature Vector Summary

Superposition[a]	Clustering Side	Feature Vector		Best Result[c]
		atoms	angle between planes[b]	
1	A	C_{23}, C_{29}	N/C	c = 3,c = 6
1	B	C_7, C_8, O_9, C_{10}, F_{50}, F_{55}	P1/C, P2/C	--
1	Full Molecule	C_7, C_8, O_9, C_{10}, C_{23}, C_{29}, F_{50}, F_{55}	P1/C, P2/C, N/C	--
2	B'	C_{10}, F_{50}, F_{55}	P1/O, P2/O	c = 9

[a] 1: Atoms defining the C-plane (N_1, C_2, N_4, C_5). 2: Atoms defining the O-plane (C_7, C_8, O_9).

[b] X/Y denotes angle between X-plane and Y-plane.

[c] "--" indicates no natural groups detected.

The B-side of the molecule can be reconstructed using six sets of atom coordinates and two plane equations. The reconstruction sequence for the B-side begins with known coordinates for atoms F_{50} and F_{55}, and known equations of the P1- and P2-planes. The two phenyl rings are constructed within the P1- and P2-planes using bond angle and bond length information for atoms in a phenyl ring. Once arbitrary positions for each phenyl ring are obtained, they are rotated about F_{50} and F_{55} within the P1- and P2-planes, respectively, such that C_{10}', the arbitrary location of C_{10}, coincides with the true known coordinates of C_{10}. Further inclusion of the known coordinates of atoms O_9, C_8, and C_7 then completely specifies the B-side of each conformation. Thus, the coordinates of atoms C_7, C_8, O_9, C_{10}, F_{50}, and F_{55}, and the equations of P1- and P2-planes form the minimal feature set for the B-side. The feature vector for B-side clustering derived from this minimal feature set is summarized in Table 4.1, and the required atoms and planes are labeled in Figure 4.2. The dimensionality of the feature space on the B-side is thus reduced to [6 x 3 coordinates + 2 angles] = 20.

The combination of the minimal feature sets for the A- and B-sides leads to the minimal feature set for the entire molecule. Since, for all conformations, N_1 was fixed at the origin and the C-plane was fixed in the $y = 0$ plane, the entire molecule can be fully described using the minimal feature sets of the A- and B-sides. Thus, the molecule can be reconstructed using known coordinates of eight atoms and three known plane equations. These eight atoms and three planes are labeled in Figure 4.2, and the feature vector derived from this minimal feature set is summarized in Table 4.1. Compared to a dimensionality of 87 based only on atom coordinates, the dimensionality of the feature space obtained here is significantly reduced to [8 x 3 coordinates + 3 angles] = 27.

2) Superposition 2: In order to focus on the part of the molecule containing the bisphenyl group, the molecule is conceptually divided into an A′- and a B′-side as shown in Figure 4.4. The molecular conformations were superimposed on the O-plane formed by atoms C_7, C_8, and O_9. For all conformers, the O-plane was fixed in the z = 0 plane with O_9 at the origin.

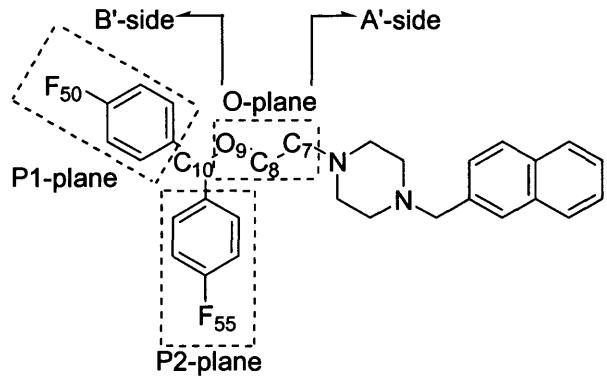


Figure 4.4 Elements of the modified feature vector for the B′-side only.

Examination of the molecule indicates that the B′-side of the molecule can be reconstructed using three sets of atom coordinates and two plane equations. The minimal feature set for the B′-side consists of coordinates of atoms C_{10}, F_{50}, and F_{55}, and the equations of the P1- and P2-planes. The reconstruction sequence for the B′-side begins with known coordinates for atoms F_{50} and F_{55}, and known equations of the P1- and P2-planes. The two phenyl rings are constructed within the P1- and P2-planes as above. After arbitrary positions for each phenyl ring are obtained, the rings are rotated about F_{50} and F_{55} within the P1- and P2-planes, respectively, such that C_{10}', the arbitrary location of C_{10}, coincides with the true known coordinates of C_{10}. Since, for all conformations, the O_9 atom is fixed at the origin and the O-plane is fixed in the z = 0 plane, atom O_9 and the O-plane are excluded from the definition of the minimal feature set for the B′-side. The

feature vector for B′-side clustering is summarized in Table 4.1, and the required atoms and planes are labeled in Figure 4.4. The dimensionality of the feature space for B′-side is [3 x 3 coordinates + 2 angles] = 11.

4.2.2 The Molecular Planes Parameter Based Feature Set

Another feature set considered for this study comprised of a less-extensive molecular planes set. The molecular planes set consisted of a relationship between a pair of planes; in this case, the four planes shown in Figure 4.2 are considered. The four planes can be related in six possible ways e.g. parameters can be defined relating the C-plane to each of the other three planes. The relationship between two planes in space can be captured in a 6-dimensional vector, which specifies the three translational parameters and three rotational parameters of one plane relative to the other. Collectively these are the six orientational parameters, given which two planes can be completely specified relative to each other in space; these six parameters are,

1) Translation along X-axis, called Shift,
2) Translation along Y-axis, called Slide,
3) Translation along Z-axis, called Rise,
4) Rotation along X-axis, called Tilt,
5) Rotation along Y-axis, called Roll, and
6) Rotation along Z-axis, called Twist.

For a detailed explanation on the significance and calculation of these parameters, the reader is referred to [104]. This methodology also ensures that the feature vectors are independent of superposition (unlike the feature vector resulting from the minimal feature set). This is due to the fact that feature vectors do not consist of absolute values; instead they consist of relative values of parameters within a particular conformer that do not depend on the absolute spatial location of the conformer itself. However, the parameters

are calculated after rigid body superposition on the four atoms of the piperazine ring (same as superposition 1 described before). To make way for a later stage comparison with clustering results produced on the proximity matrix derived from the minimal feature set, the molecule is conceptually divided into A-side and B-sides. The feature sets are described below.

- A-side feature set: The A-side can be described by the orientational parameters of the N-plane relative to the C-plane. This relationship is depicted through the resultant feature vector, $[NxC]_{T+R}$, where T stands for translation and R for rotation. Considering only the translational or the rotational parameters results in two other feature vectors, denoted by $[NxC]_T$ and $[NxC]_R$ respectively. While $[NxC]_{T+R}$ is a 6-D vector with mixed features, $[NxC]_{T+R}$, and $[NxC]_R$ are both 3-D vectors of homogenous features.
- B-side feature set: The B-side consists of three planes, C-, P1-, and P2-planes. As a result 18 orientational parameters completely define the B-side. However, there is a structural dependency between the P1- and the P2-planes. There is a certain redundancy involved if both P1- and P2-planes are explicitly defined. This means that instead of using all 18 parameters, an equivalent B-side relationship can be built by considering any six parameters that define the orientation of the C-plane with respect to either P1- or P2-planes. The three feature vectors considered in this study are, $[CxP2]_{T+R}$, $[CxP2]_T$, and $[CxP2]_R$.
- Full molecule feature set: The molecule is also considered in its entirety. This can be seen either as a combination of A- and B-sides or even more concisely, as a relationship between two of its extreme pair of planes. The latter approach is

considered here and the full molecule is described by the parameters relating the N- and P2-planes. The three feature vectors obtained are denoted by, $[NxP2]_{T+R}$, $[NxP2]_T$, and $[NxP2]_R$.

4.2.3 Distance Measures and Proximity Matrices

The feature vectors generated using the minimal feature set and the molecular planes feature set are converted to proximity matrices and these matrices are used as input to the clustering routine. The clustering procedure based on these matrices is described in the next section. In all cases, except the $[\ ]_T$ and the $[\ ]_R$ feature vectors from the molecular planes feature set, the feature vectors consist of mixed features. Strictly speaking however, the features are absolute values in different units of measurements, as against strict mixed data types (such as binary combined with absolute or interval data). It is not advisable to construct a distance measure on a mixed feature set; however, since the features are mixed features only in terms of their units, a simple distance measure is built based on the sum of Euclidean distances with or without a scaling factor.

Any metric-based distance measure between two entities j and k, be it Euclidean, non-Euclidean, or semi-metric, has to conform to the following

$$D_{jk} \geq 0; D_{jk} = D_{kj}; D_{jj} = 0. \tag{4.1}$$

For the feature vectors based on the minimal feature set, the distance measure utilized a sum of RMS distances without scaling. The atom coordinates are measured in a Cartesian Angstrom (Å) space, while the angle between planes is measured in angles, and later converted to radians (rad). The range of atom coordinate RMS differences is seen to be of the same order of magnitude as the range of plane angle RMS differences. This prompted a metric-based on sum of Euclidean distances to be used, one that at worst

would be a semi-metric, satisfying the constraints in Equation (4.1). However, there is no way of ascertaining if the metric is indeed non-Euclidean or even Euclidean. The feature vector based on the minimal feature set describes a conformer over a set of a atom locations (in x-y-z space) and b plane angles, [ang]; the distance between two conformers, j and k is defined as,

$$D_{jk} = \left[\sum_{a} (x_j - x_k)^2 + (y_j - y_k)^2 + (z_j - z_k)^2 \right]^{1/2} + \left[\sum_{b} (ang_j - ang_k)^2 \right]^{1/2}. \tag{4.2}$$

This formulation satisfies the three constraints in (4.1). For proximity matrices based on the orientational parameters from the molecular planes feature set, the distance between any two conformers j and k is defined as,

$$D_{jk} = \left[\sum_{p=1}^{3} (t_{pj} - t_{pk})^2 + s \sum_{p=1}^{3} (r_{pj} - r_{pk})^2 \right]^{1/2}, \tag{4.3}$$

where t_{pj} and t_{pk} are the translational parameters for j and k respectively, and r_{pj} and r_{pk} are the rotational parameters for j and k respectively, $1 \leq p \leq 3$. The scaling factor, s is a constant chosen accordingly to the scales of the translational parameters relative to the rotational parameters. A judicious choice here is a ratio of the absolute squared differences between the maximum and minimum of the translational parameters and the rotational parameters over the entire data-set, and is given by

$$s = \frac{(t_{\max} - t_{\min})^2}{(o_{\max} - o_{\min})^2}. \tag{4.4}$$

Such a scaling scheme is known as *range-based scaling*. Another attractive scaling methodology involves transformation of each column of features (six columns for the six orientational parameters) to standard z-scores, such that the resultant standardized columns each have a mean of zero and unit standard deviation [105]. This is done prior

to computing the proximities using the Euclidean distance norm. Multidimensional and metric scaling to reduce the 6-D mixed feature set to a consistent lower dimensional set can also be done. However, the range scaling employed in this study is found to be sufficient for analysis. For feature sets consisting of only the translational or the rotational parameters, no scaling is required.

4.3 Fuzzy Relational Clustering of DM 324 Conformers

Though an object-space-based fuzzy clustering scheme such as FCM can be used to cluster the data on the reduced feature space, it was decided that converting the data in the reduced feature space into a proximity distance matrix would provide a better understanding of the inter-conformational similarities. Also, once such a matrix is obtained, it is easier to work in a relational domain rather than in the object space. As a step towards the development of a general methodology, such a proximity matrix could also handle any subjective or non-Euclidean similarity information which would be nearly impossible to achieve in an object space. The proximity matrix obtained from this could be a non-Euclidean measure of dissimilarity (or in the worst case, it could be a semi-metric). This provided the motivation to use a relational clustering technique capable of handling non-Euclidean data to generate partitions. The Fuzzy Relational Clustering [106] algorithm is used to generate fuzzy partitions.

Fuzzy Relational Clustering (FRC) is a recently-developed relational clustering technique, and is conceptually attractive because it works directly on the non-Euclidean data without first converting it to a Euclidean measure. The scheme is therefore less constrained than most of the other relational clustering techniques. Given a dissimilarity

data matrix, $\mathbf{D} = [D_{jk}]$, $1 \leq j, k \leq n$, FRC only assumes that its elements are subject to the minimal constraints in Equation (4.1). The algorithm then alternates between optimizing the memberships, $\mathbf{U} = [u_{ik}]$, and a related distance matrix, $\mathbf{A} = [a_{ik}]$, $1 \leq i \leq c$, $1 \leq k \leq n$, using a successive-substitution method as described in [106]. Here n = 728, is the number of conformers and c is the number of clusters fixed *a priori*. The update equations used for **U** and **A** are shown in Equations (4.5) and (4.6).

$$u_{ik} = \frac{\left[\frac{1}{a_{ik}}\right]^{1/(m-1)}}{\sum_{w=1}^{c}\left[\frac{1}{a_{wk}}\right]^{1/(m-1)}}, \tag{4.5}$$

$$a_{ik} = \frac{m\sum_{j=1}^{n} u_{ij}^{m} D_{jk}}{\sum_{j=1}^{n} u_{ij}^{m}} - \frac{m\sum_{h=1}^{n}\sum_{j=1}^{n} u_{ij}^{m} u_{ih}^{m} D_{jk}}{2\left[\sum_{j=1}^{n} u_{ij}^{m}\right]^{2}}. \tag{4.6}$$

The c-mean vectors, $\mathbf{V} = [v_i]$, $1 \leq i \leq c$, are scaled n-tuples of memberships

$$v_i = \frac{(u_{i1}^{m}, u_{i2}^{m}, \ldots, u_{in}^{m})^{T}}{\sum_{k=1}^{n} u_{ik}^{m}}. \tag{4.7}$$

The membership matrix, **U** is initialized randomly. The number of clusters c (> 1), and the fuzzifier m (= 2), are fixed. The algorithm then iterates between Equations (4.5) and (4.6), until the change in memberships in two successive iterations falls below a certain prefixed threshold, ε (= 10^{-6}). Termination of the algorithm indicates that a local minima partition is achieved. In every iteration, the c-mean vectors are updated using Equation (4.7) after all the membership values have been updated. After the algorithm converges, the membership information is defuzzified by assigning the conformation j to the cluster i if $u_{ij} > u_{kj}$ ($k \neq i$) for all $1 \leq i \leq c$, $1 \leq j \leq n$. The representative conformation

is identified as the one with the highest membership value in that particular cluster i.e. for cluster i, the representative conformation is defined as the conformation l if $u_{il} > u_{ij}$, $(l \neq j)$ for all $1 \leq j \leq n$. This process is carried out for a range of values for c. The clustering results are then evaluated by using the following cluster validity indices,

1) Partition coefficient F, as given by Equation (3.1),

2) Normalized partition coefficient F', as given by Equation (3.2),

3) Partition entropy H, as given by Equation (3.3),

4) A modified relational version of the Xie-Beni compactness index Equation (3.4) based on the object space compactness index S, and reformulated as,

$$S = \frac{\sum_{i=1}^{c}\sum_{k=1}^{n} u_{ik}^2 a_{ik}}{n\left[\min_{1 \leq i,j \leq c, i \neq j} \| v_i - v_j \|^2\right]} \tag{4.8}$$

4.4 Results

The results of cluster analysis for the various feature sets described in Section 4.2 are presented here. This includes clustering and validation studies. With the parameters described in the last section, FRC is run multiple times for a range of clusters and the results stored. For each level of clustering (fixed c), the partition that repeats itself the most number of times is chosen as the solution. This is necessary because bad initializations in many cases lead to bad partitions and unless the process is tested for repeatability, the results can not be trusted. The partition-frequency approach was later discarded in favor of an approach based on the minimum value of the objective functional. The partition that repeats itself more than others is also the one that has the

lowest value of the FRC objective functional among all partitions. Keeping a track of the FRC functional J_{FRC} is sufficient, as opposed to a record of partition frequencies.

$$J_{FRC} = \sum_{i=1}^{c} \frac{\sum_{j=1}^{n}\sum_{k=1}^{n} u_{ik}^{m} u_{ij}^{m} D_{jk}}{2\sum_{t=1}^{n} u_{it}^{m}} \tag{4.9}$$

Once the solution partition for a particular c is identified, the memberships and prototype information are used to calculate the four validity indices. The process is then repeated with single step increments for c. For every feature vector, the validity indices are plotted against the number of clusters and trends identified. A good partition is characterized by high values of F and F', and corresponding low values of H and S. However, in many cases, the validity indices are found to be inconclusive, and the only conclusion in such cases is that there are no natural groups to be found in the data when clustered using a particular feature vector. This may or may not mean that there is no substructure in the data-set; however, if there is a substructure, inconclusive evidence from validity measure only suggests that the feature vector under consideration is insufficient to capture information about the presence of divisive substructure.

4.4.1 Clustering Results for the Minimal Feature Set

The optimal number of clusters found for each feature vector for the minimal feature set, sorted by superposition, is given in the last column of Table 4.1. The flexibility of the molecule ensured that a large conformational space was covered by the random search protocol, as can be seen by superposition of all 728 conformations in Superposition 1, (shown in Figure 4.5). Clustering of the conformations using the full-molecule feature vector outlined in Table 4.1 indicated the absence of natural groups according to the

behavior of the cluster validity indices (not shown). This is perhaps not surprising given the wide range of positions occupied by the atoms of the B-side in Superposition 1. Figure 4.5 shows more clearly-defined groups on the A-side of the superimposed conformations due to more limited positions available to the naphthalene ring. Since the piperazine and naphthalene rings contain the pharmacophore features that are found in most inhibitors, the next clustering study used a feature vector defined only in terms of the A-side in order to focus on these pharmacophore features. The cluster validity results for the A-side partitions for Superposition 1 are shown in Figure 4.6. All four validity indices attain their first inflexion point and their respective optima at $c = 3$ suggesting good three-cluster partition. The compactness index S indicates good partitioning for $c = 6$ through $c = 9$, with the other three indices either monotonically increasing or decreasing over that range. This suggests a good second-level partitioning at the lower bound, $c = 6$.

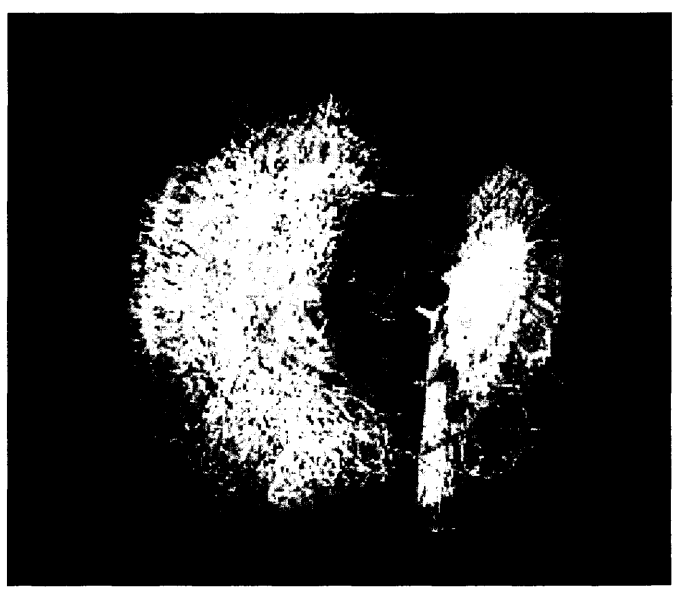


Figure 4.5 Side view of the 728 conformations superimposed on the four atoms of the piperazine ring (Superposition 1).

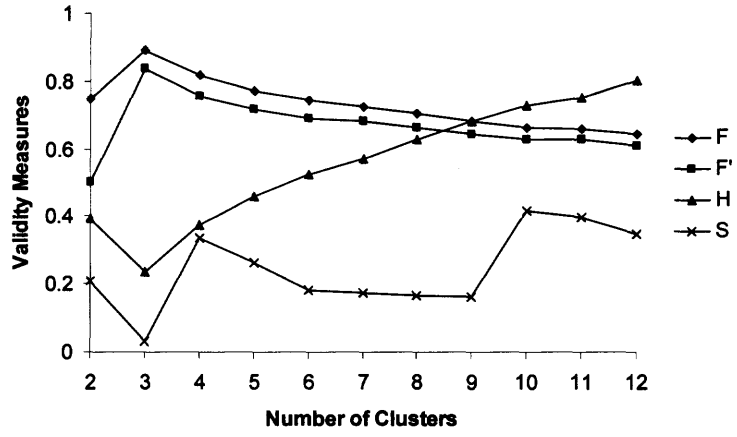


Figure 4.6 Cluster validity plots for partitions on the A-side.

Figure 4.7 shows the molecular conformations that correspond to $c = 3$ and $c = 6$ clustering results. The view depicted is a 90° clockwise rotation about the central plane of Figure 4.5 such that the A-side naphthalene rings are presented frontally (the piperazine ring and B-side are not shown). For a detailed discussion on bond torsional angles, the reader is referred to [107].

Clustering using the B-side feature vector indicated the absence of natural groups. This is consistent with the fact that the B-side is much more flexible than the A-side due to the presence of six rotatable bonds on the B-side versus two on the A-side. The B-side can access a much wider range of conformational space than the A-side, as shown in Figure 4.5. None of the validity indices provide a reason to believe that there is an underlying structure on the B-side (Figure 4.8). The compactness index S is not plotted because the results were not considered to be sufficiently consistent, indicating a lack of substructure. The normalized coefficient, F', takes values very close to zero ($cF \rightarrow 1$), and hence the results at all levels of clustering are too fuzzy to be of any significance.

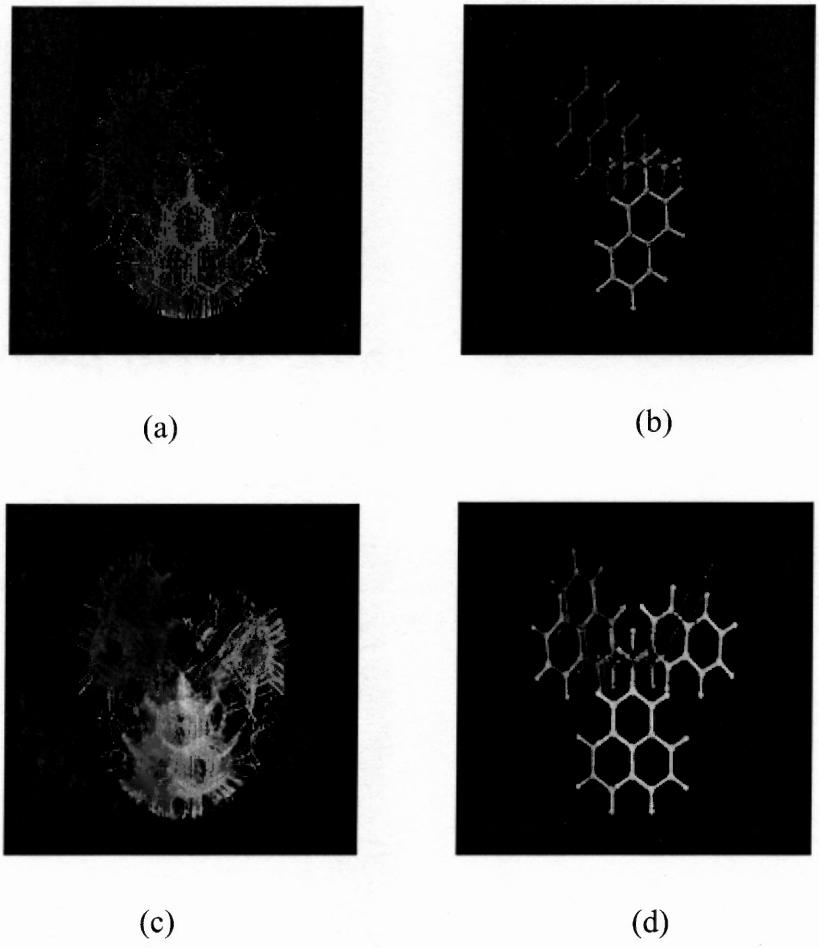


Figure 4.7 Results for the A-side clustering at $c = 3$ and $c = 6$, Superposition 1. For clarity, the B-side and the superimposed piperazine ring are not displayed.
(a) For $c = 3$, three distinct clusters. Number of conformations: red - 229, blue - 270, and green - 229.
(b) For $c = 3$, representative structures. Conformation number (membership value): red - #62 (0.998), blue - #251 (0.997), green - #96 (0.999).
(c) For $c = 6$, six distinct clusters. Number of conformations: magenta - 77, red - 153, blue - 128, cyan - 142, yellow - 82, and green - 146.
(d) For $c = 6$, representative structures. Conformation number (membership value): magenta - #154 (0.990), red - #531 (0.994), blue - #428 (0.991), cyan - #248 (0.995), yellow - #232 (0.996), green - #177 (0.998).

The cluster validity indices plotted in Figure 4.9 suggest nine optimal clusters for the B′-side (superposition 2). The compactness index, S, has its lowest value for $c = 9$. The other indices support this partition, indicating well-separated and compact clusters.

Comparison of Figure 4.4 to Figure 4.2 shows that the B'-side contains only three rotatable bonds instead of the six bonds on the B-side. Since Superposition 2 is based on the O-plane, it allows for observable partitions on the B'-side of the superimposed conformations.

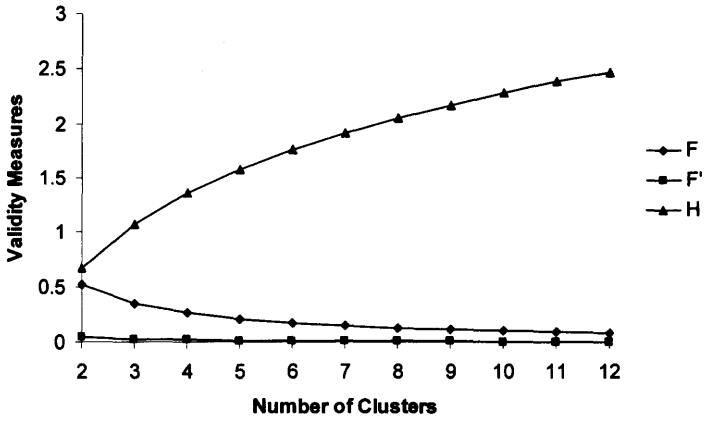


Figure 4.8 Cluster validity plots for partitions on the B-side.

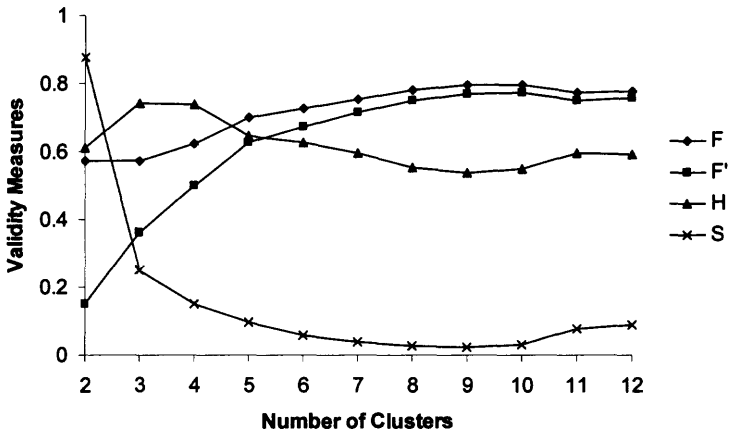
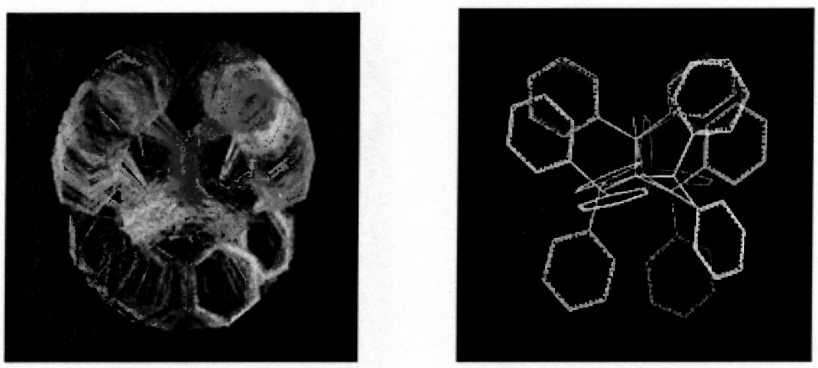


Figure 4.9 Cluster validity plots for partitions on the B′-side.

Figure 4.10 shows the nine B′-side clusters as well as the representative conformations from each cluster. Each cluster is formed by the bisphenyl group on the B′-side (the A′-side is not shown). Each phenyl ring of the bisphenyl group in a cluster

occupies two different regions in space. For example, three clusters (blue, green, and white) have both of their phenyl rings located out on the edge and six clusters (red, magenta, purple, cyan, orange, and yellow) have one phenyl ring located out on the edge and the other located in the center, coming out of the plane of the figure. Since no two colors appear in the same region, the clusters are distinct e.g. while one phenyl ring of both the orange and the yellow clusters seems to be overlapping in the center, the other phenyl ring of the orange cluster lies on the bottom left and that of the yellow cluster lies on the top left. Thus, the orange and yellow clusters are distinct and do not overlap.



(a) (b)

Figure 4.10 Clustering results for the B′-side at $c = 9$.
(a) Nine distinct clusters. Number of conformations: red - 99, orange - 48, magenta - 146, blue - 31, white - 49, cyan - 87, purple - 52, green - 114, and yellow - 102.
(b) Nine representative structures. Conformation number (membership): red - #213 (0.999), orange - #72 (0.994), magenta - #307 (0.999), blue - #207 (0.995), white - #108 (0.997), cyan - #402 (0.998), purple - #150 (0.995), green - #692 (0.978), and yellow - #716 (0.992).

(a) (b)

Figure 4.11 Full-molecule representative structures that will be used as input for CoMFA. Conformers are aligned using (a) Superposition 1, and (b) Superposition 2.

Since the full-molecule clustering suggested the absence of natural groups, the superposition-based and region-specific clustering results obtained above are used to identify putative representative structures. The A-side torsion angles of the six cluster representatives from the A-side clustering are combined with the two B′-side torsion angles of the nine representatives from the B′-side clustering to construct 54 *ideal* combinations of the four torsion angles. A search through the data-set of 728 conformations using a tolerance of ±2.5° on each torsion angle produced six matches. The representative conformers are shown in Figure 4.11 in both superposition 1 and 2.

4.4.2 Clustering Results for the Molecular Planes Feature Set

Cluster analysis failed to locate natural groups when the full molecule was clustered using a proximity matrix derived from eight atom locations in 3-D and three sets of angles between planes, as described in the previous section. In contrast, partitions produced for the $[NxP2]_{T+R}$ proximity matrix indicate the presence of five clusters. This is confirmed by the compactness index *S*; however, the other validity measures are

inconclusive. The cluster validity results for the full molecule are shown in Figure 4.12. The compactness index, S takes it lowest value at $c = 5$ over the range $2 \leq c \leq 14$. Entropy, H, is seen to be monotonically increasing and F and F' are monotonically decreasing over the entire range and are hence, inconclusive.

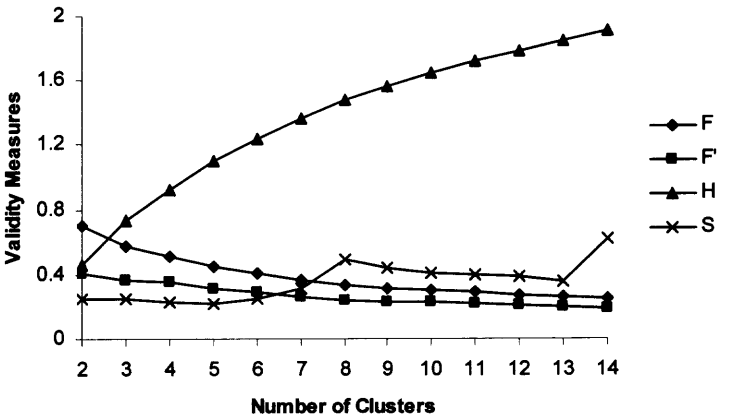
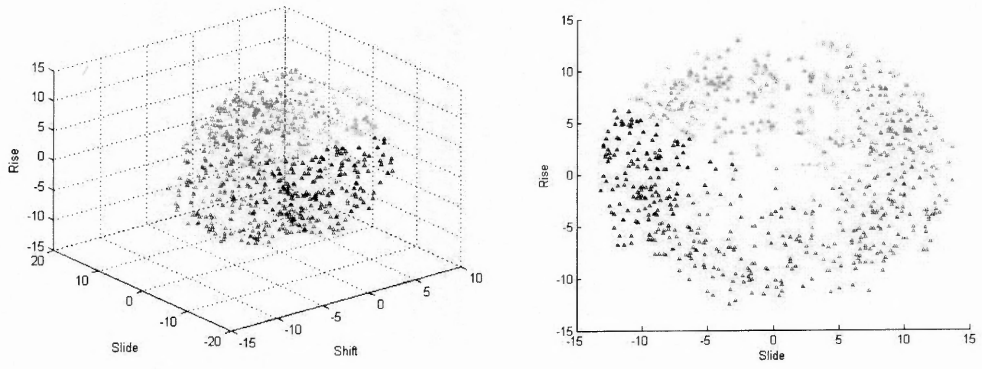


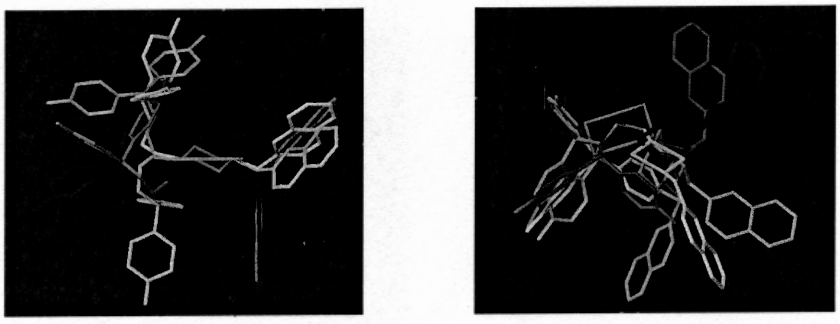
Figure 4.12 Cluster validity plots for partitions on the $[NxP2]_{T+R}$ proximity matrix.

The separation into five clusters is best visualized in the 3-D translational space (Shift vs. Slide vs. Rise), and on the 2-D (Slide vs. Rise) plane as shown in Figures 4.13(a) and (b). This indicates the possible importance of the translational parameters over the rotational parameters in full-molecule clustering and provides motivation for examination of the results from the full-molecule clustering based on either translational or rotational feature vectors. In Figure 4.13 and all other 2-D and 3-D plots, the conformers are color-coded by cluster; the translational parameters are given in Angstroms (Å), and the rotational parameters in degrees.



(a) (b)

Figure 4.13 Conformers plotted for $c = 5$ on $[NxP2]_{T+R}$, (a) in the 3-D Shift vs. Slide vs. Rise space, and (b) on the 2-D Slide vs. Rise plane.



(a) (b)

Figure 4.14 Full-molecule representative conformers for $c = 5$ on $[NxP2]_{T+R}$; aligned using (a) Superposition 1, and (b) Superposition 2.

The validity plots for the translational component proximity matrix, $[NxP2]_T$, are very similar to those in Figure 4.12, and also identify five clusters (not shown here). This seems to indicate that the translational parameters may be the chief determinant of separation in full-molecule clustering, at least for molecules with planes separated by a distance of the order magnitude (or greater) than that between the N- and P2-planes. The cluster validity plot for the rotational proximity matrix, $[NxP2]_T$, identifies 13 clusters

(not shown here). The representative conformers for c = 5 are shown in the two superpositions in Figure 4.14. It should be noted that while clustering using the molecular planes feature set is independent of superposition, results are better visualized by plotting the superimposed representatives.

The A-side is described by the proximity matrix on the $[NxC]_{T+R}$ feature space. As in the full-molecule case, separate analyses are carried out for proximities on the $[NxC]_T$ and $[NxC]_R$ feature spaces. Figure 4.15 shows the cluster validity plots for the A-side over the range $2 \leq c \leq 14$. At $c = 9$, F and F' take their maximum values, and H and S take their minimum values. Unlike the full-molecule partitions, all four validity measures seem to be in agreement in this case.

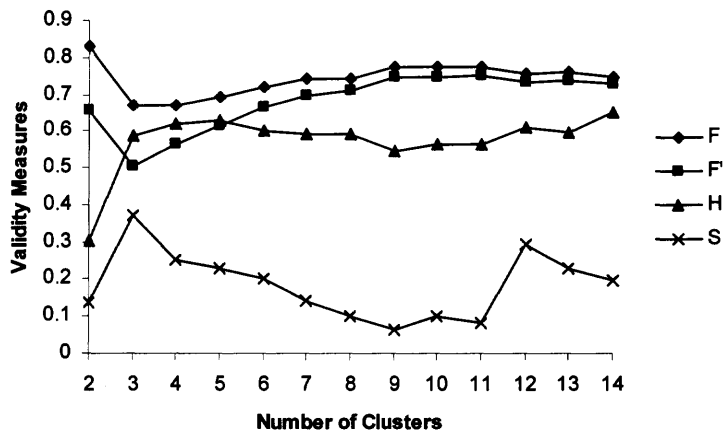
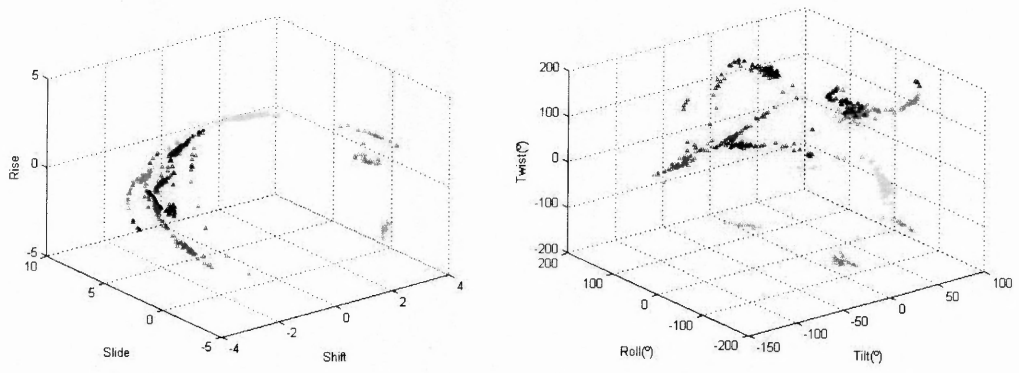


Figure 4.15 Cluster validity plots for partitions on the $[NxC]_{T+R}$ proximity matrix.

Conformers in the 3-D translational (Shift vs. Slide vs. Rise), and 3-D rotational (Tilt vs. Roll vs. Twist) space are shown in Figures 4.16(a) and (b). The separation of conformations into nine clusters is clearly visible in both translational and rotational space.



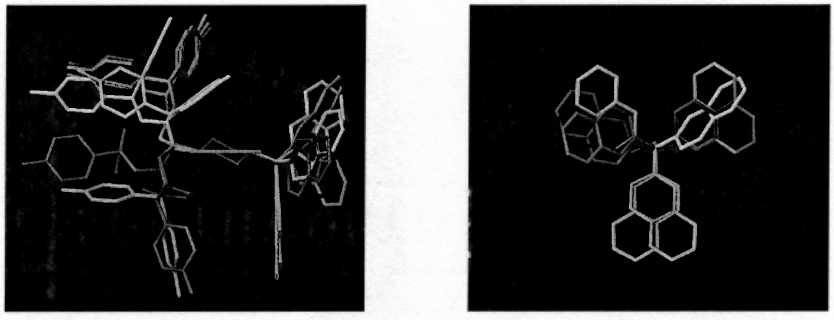
(a) (b)

Figure 4.16 Conformers plotted for $c = 9$ on $[NxC]_{T+R}$, (a) in the 3-D Shift vs. Slide vs. Rise space, and (b) in the 3-D Tilt vs. Roll vs. Twist space.

This corroborates the fact that the nine clusters identified by the FRC, and validated by the cluster validity measures are indeed natural clusters. In contrast to the full-molecule results, both the translational and rotational parameters appear to play a role in separating the conformations into clusters. This may be because the N- and C-planes are, for most of the conformations in this study [108], much closer in space than the N- and P1- or N- and P2-planes. The proximity of the N- and C-planes means that their relative rotation as well as their relative separation is important to the clustering. For planes that are far apart (N- and P1-, or N- and P2-planes), their relative rotational orientation is of lesser significance to clustering than their distance of separation. The nine representative conformers are superimposed based on superposition 1, and are shown in Figures 4.17(a) and (b).

As shown in Figure 4.18 for the $[CxP2]_{T+R}$ proximity matrix, the cluster validity measures for the B-side clustering are not as indicative as those for the A-side or the full-molecule. The compactness index, S behaves well over $2 \leq c \leq 6$, after which it assumes unnaturally big values, which is indicative of an infinitesimally small distance between

closest prototype centers found for all $c > 6$. In other words, good clusters are arbitrarily subdivided into artificial overlapping clusters for all $c > 6$. This means that the searchable region was confined to $2 \leq c \leq 6$. At $c = 3$, S attains its lowest value and F' attains its maximum value. The other two indices, F and H, are non-indicative for $2 \leq c \leq 6$. Figure 4.19 shows the conformers plotted in 3-D translational space for $c = 3$ for cluster analysis on the $[CxP2]_{T+R}$ proximity matrix.



(a) (b)

Figure 4.17 Representative conformers for $c = 5$ on $[NxP2]_{T+R}$; aligned on the piperazine ring (superposition 1); (a) side view, and (b) end view.

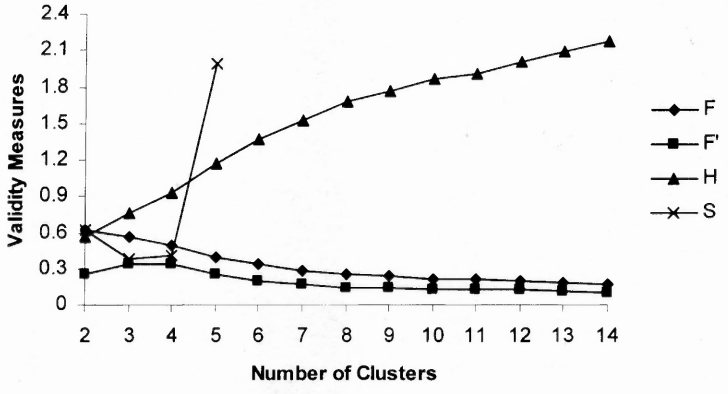


Figure 4.18 Cluster validity plots for partitions on the $[CxP2]_{T+R}$ proximity matrix. The plot for S is truncated at $c = 5$.

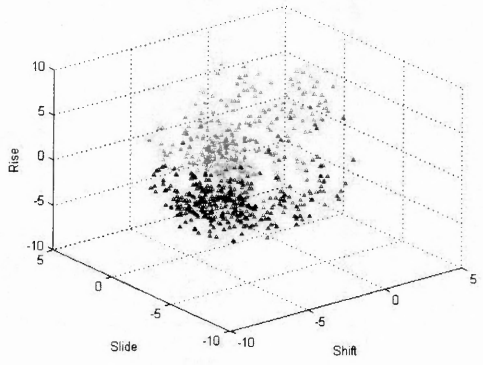
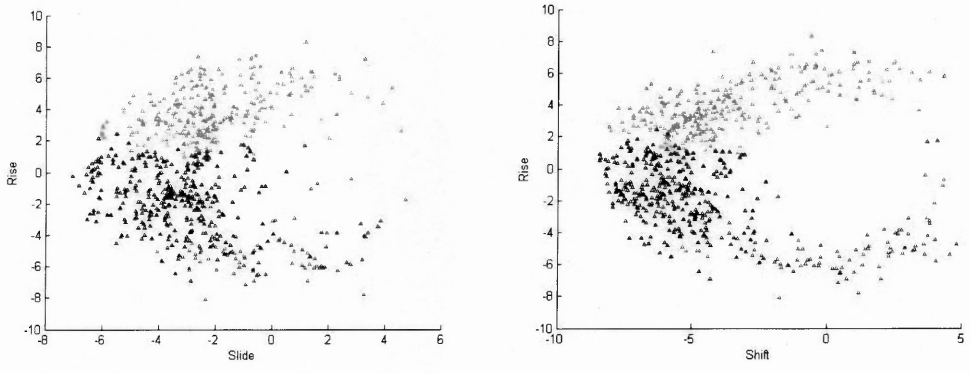


Figure 4.19 Conformers plotted for $c = 3$ on $[CxP2]_{T+R}$, in the 3-D Shift vs. Slide vs. Rise space.

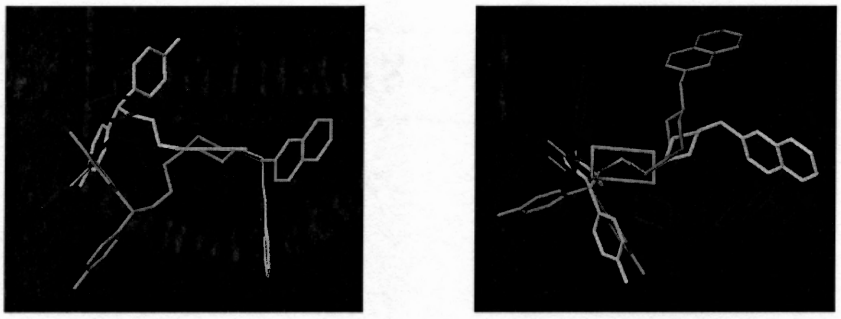


(a) (b)

Figure 4.20 Conformers plotted for c = 3 on $[CxP2]_{T+R}$, (a) on the 2-D Slide vs. Rise plane, and (b) on the 2-D Shift vs. Rise plane.

In contrast to the 3-D and 2-D rotational parameter plots, the conformers separate well in the translational space. This can be seen particularly in the 2-D (Slide vs. Rise), and (Shift vs. Rise) plots of Figures 4.20(a) and (b), respectively. This indicates that the translational rather than rotational parameters determine the B-side clustering. This is similar to the full molecule case and is again due to the fact that the C- and P1- (or P2-) planes are relatively further apart than the C- and N-planes of the A-side study, for which both translational and rotational parameters contribute to clustering. This is supported by

the cluster validity results for the $[CxP2]_T$ study, which indicate a similar number of clusters (c = 4). In contrast, the $[CxP2]_R$ study identifies eight clusters. The three representative conformers as identified by cluster analysis on $[CxP2]_{T+R}$ are shown in Figures 4.21(a) and (b).



(a) (b)

Figure 4.21 Representative conformers for c = 3 on $[CxP2]_{T+R}$; representatives are aligned using (a) Superposition 1, and (b) Superposition 2.

4.5 Discussion and Conclusions

The approaches presented here differ from conventional classification approaches in computation chemistry. Two feature extraction procedures have been proposed here with emphasis on conformational clustering. Cluster analysis is performed on large feature sets with very encouraging results. The minimal feature set methodology has also been generalized and is shown to be applicable to any large flexible molecule [107]. The proposed approaches are novel for several reasons. First, it seems to be the only fuzzy clustering study of a very flexible molecule. Second, region-specific clustering that focused on individual pharmacophore elements of the molecule was made possible by defining feature vectors in terms of the A- and B-side, or A′- and B′-side moieties which

contain important chemical features of the pharmacophore. Third, the practical applicability of FRC to a large data-set is shown. The FRC procedure used a proximity matrix derived from a feature vector that contained real spatial elements (atom coordinates, angles between planes, orientational parameters relating pairs of planes) that were related to the pharmacophore elements of the molecule.

4.5.1 Minimal Feature Set vs. Molecular Planes Set

Both the approaches deal with the problems associated with labeling the symmetric heavy atoms and the large dimensionality of the data-set, in different ways. While the minimal feature set approach constructs the feature set by focusing on bare-minimal information needed to reconstruct the molecule, the molecular planes approach constructs the feature set based on the structural relationship between a pair of molecular planes. There are numerous ways in which objective information can be extracted from the two different feature sets, reflected later in the numerous features vectors used for cluster analysis. The minimal feature set for the full-molecule consisted of eight atom locations and three molecular planes, from which the feature vector comprising of eight atom coordinates and three angles was extracted. However, this feature vector failed to uncover any meaningful substructure.

On the other hand, the molecular planes feature set for the full-molecule consisted of a pair of extreme planes, and one of the corresponding feature vectors was a 6-D vector of three translational and three rotational parameters relating the two extreme planes. Cluster analysis on this feature vector resulted in a five-cluster partition. The reason for this could be that although the former feature set is minimal, the resulting feature vector is not minimal. In other words, the feature vector consisting of eight atom

coordinates and three angles between pair of planes does not capture sufficient structural information. This is because of the fact that only three scalar parameters are extracted from a set of three planes, which might be less than sufficient. The molecular planes approach extracts six parameters from a pair of planes (which might be more than minimal and is certainly more than sufficient). A combination of the two feature vectors seems to be a better choice for clustering and could be a topic of future research.

4.5.2 Cluster Validity Measures

Interesting trends are observed in the cluster validity measures used in this study. Any clustering procedure has the capacity to uncover groups in data; however, in most cases, one is only interested in uncovering *natural groups* (as defined in Chapter 3), as against non-existent artificial groups. It has been observed in this study that whenever there are natural groups to be found, all four cluster validity measures showed a definite trend (were all conclusive and in accordance with each other). Such was the case with the A-side for both features sets and the B'-side for the minimal feature set. In the absence of natural groups, F and F' show a monotonically decreasing tendency, while H exhibits a monotonically increasing tendency. The compactness index S is the only index among the four considered in this study that shows definite trends (inflexions, minima, and maxima) irrespective of the presence or absence natural substructure. This leads to an interesting hypothesis – a two-level cluster validation methodology; in the first step, the absence or presence of natural partitions is ascertained using F, F', H, or a combination of these. Later, S is used along with the other three indices to accurately identify the optimum value of c. However, the testing of this hypothesis is beyond the scope of this work.

CHAPTER 5

IMAGE SEGMENTATION AND CLUSTERING APPLIED TO CONDITION STATE ASSESSMENT OF PIPELINES

5.1 Introduction

There are several hundred thousand miles of pipelines stretching across the United States of America carrying essential fluids such as natural gas, chemical and petroleum products, and drinking water. These pipelines are not typically monitored for failures, resulting in loss of product, contamination of fluids, release of hazardous materials, stoppage of essential fluid delivery, and collateral life and property damage. Digital photography is a preferred medium for inspection of underground infrastructure such as water main pipes. With digital photography, large amounts of information collected render it unviable and impractical for manual processing. However, due to the unavailability of automated fault diagnosis techniques, the process of identifying defects is still performed manually.

Closed circuit television (CCTV) surveys are conducted using a remotely controlled robotic vehicle carrying a television camera through an underground pipe. The output is usually in the form of analog or digital videotapes of the interior of the pipe, which are then analyzed either on site or later by a technician. The record produced by the technician depends on his or her experience, expertise and capability, making the process subjective and prone to errors. References can be found in the literature where various researchers have come up with techniques to automate the process of locating defects in pipelines [109]-[115]. However, manual intervention does not end at the defect identification level; more subjectivity enters in when condition states (or ratings)

are assigned to the pipe based on the observed level and severity of the damage. Not only is an automated fault diagnosis system needed to identify and locate damage, but an automated condition state assessment system is also needed to augment it.

The objective of image segmentation is to divide an image into meaningful regions [116]. All image segmentation procedures divide image into regions that are homogenous with respect to some criteria and adjacent regions differ with respect to the same criteria; the criteria most commonly chosen are gray levels and/or texture [117]. For a detailed survey of image segmentation procedures the reader is refered to [118] and [119]. In many situations it is not clear whether a certain pixel should belong to a region or not. This is because the features used to determine homogeneity may not have sharp transitions at region boundaries. A fuzzy set-based segmentation process can take care of the uncertainty associated with assignment of a boundary pixel. The first reference of fuzzy image segmentation was made in [120]. Four broad classes of segmentation methods are

- Edge-based methods – These methods are based on detection of spatial discontinuity and edges in the edge.
- Region-based methods – These methods are based on detection of spatial similarity between pixels, such as region growing methods.
- Shape-based methods – These methods are based on the knowledge of the shape of the objects to be segmented. Template matching and mathematical morphology are two types of shape based segmentation procedures.
- Classification-based methods – These methods use pixel information and classify them into regions based on an optimization criterion. Thresholding and clustering are examples of classification based segmentation procedures.

Fuzzy clustering has not been extensively used as an image segmentation tool. An image of an outdoor scene is segmented using FCM, Gustafson-Kessel (GK) algorithm,

and Gaussian mixture decomposition (GMD) algorithm, and the results compared in [116]. A segmentation algorithm that models the uncertainty in pixel information based on FCM is presented in [121]. Other instances of the use for FCM or FCM-based clustering procedures for image segmentation can be found in [122], [123]. In the next section a novel two-step classification based image segmentation procedure is presented.

5.2 Image Segmentation by Fuzzy Clustering

Monitoring structural integrity of pipes is essential to timely implementation of maintenance and rehabilitation tasks. Closed circuit television (CCTV) based inspection is one of the most inexpensive inspection techniques developed over the recent years. Video inspection data provide general information about the state of the pipe compared to information obtained using specialized inspection techniques that use acoustic, magnetic and electrical property changes in the pipe to ascertain and pinpoint damage locations. Video inspection output in the form of videotapes, both analog and digital, provide a visual verification of the presence of damage. Internal damage in pipes is usually in the form of random-shaped cracks, holes and others. A number of pattern recognition and image processing methods have been proposed in the literature and almost all of these are based either on edge detection methods, or mathematical morphology analysis, or a hybrid method of these two approaches [111]. A Neural Network based approach for image processing, image segmentation, and feature extraction is presented in [114], and recently a neuro-fuzzy classification algorithm has been proposed in [109]-[111]. The defect diagnosis approach presented here is different from previous attempts at automated diagnosis – the images used here are extremely noisy and low in resolution, while the

defects in images in [114] are clearly identifiable. The pipe is viewed *straight-up* with the pipe-end at the center of the image frame, unlike [111] where the camera zooms into the defect portions of the pipe and captures a close-up view of the defect section. The scanned images in [111] are obtained by the Pipe Scanner and Evaluation Technology (PSET) camera [124], while images in this study are obtained using a regular CCTV system of defect reporting

Typically, the inspection camera is directly hooked onto an image processing center, and transmits real time video images to an operator in a mobile unit connected to cameras and the crawler mechanism. The videos are converted to digital mpeg files, and the corresponding identifying information is entered by the operator. The software prompts the operator to enter damage location, a descriptive account of the damage and assign a corresponding severity number. The software then provides detailed graphical and summary reports of the damage. The involvement of the operator makes the system vulnerable to lapses in operator concentration, inexperience and subjectivity. Additionally, in many cases, the lighting conditions are insufficient to provide the operator with a clear picture of the state of the pipe. The aim of this research is to implement a quick and real time automated system for simultaneous detection of defects and condition state assessment based on the collated defect data.

The mpeg file produced by the inspection is a visual record of the interior of the pipe. The present system of manual inspection involves moving the camera to all places of the pipe, not just looking at it *straight-up* but moving it sideways and zooming-into locations where a defect has been identified by the operator. In order to automate the system, it is proposed that the camera be moved in a straight line along the center of the

pipe and always photograph the pipe *straight-up*. This will eliminate ambiguities within frames, expedite the inspection, and result in a consistent series of images. It is also proposed that for proper installation of the completely automated system, the internal lighting conditions produced by the camera flash be consistent throughout. The present digital inspection systems record the video on a 3-plane RGB format. The mpeg format is tagged on a time scale and frames can be extracted using frame-capture software such as Pinnacle Studio® or any other appropriate software. The tagged mpeg can also be broken down into frames using the tag numbers as frame capture criterion. For a 30 frames per second (fps) video, the pinnacle system can extract 30 frames in a second on a real time basis and save each frame as a jpeg image. For the sake of automation, it is not advisable or practical to perform image analysis on all the extracted frames – two consecutive frames on a 30 fps video would be almost identical, and this implies that the second frame can be neglected. A consistent methodology to skip frames is described later. Figure 5.1 shows 10 frames extracted from an 18 second segment of a typical video.

5.2.1 Preprocessing

To perform a meaningful analysis of images extracted from the video, the images need to be pre-processed, since the captured image is usually poor in contrast and contains more information than needed. The first step of the preprocessing is the reduction of the image dimensionality. The assumption made here is that the camera movement is controlled to prevent variation in orientation between images. With a steady straight line motion, the center of the image is always occupied by the end of pipe, which appears as a dark circular region, due to the inability of the camera flash to illuminate regions beyond a

certain limiting distance away from the camera. It is also known that most of the damage typically occurs below the water line, which passes approximately through the center of the image. Hence, as a first-cut, the entire top portion of the image above water line can be ignored. Moreover, it is seen that the image is considerably darker near the top compared to the well-lit bottom portion. From experiments and *a priori* knowledge of the movement patterns of the camera, these regions can be accurately estimated and eliminated.

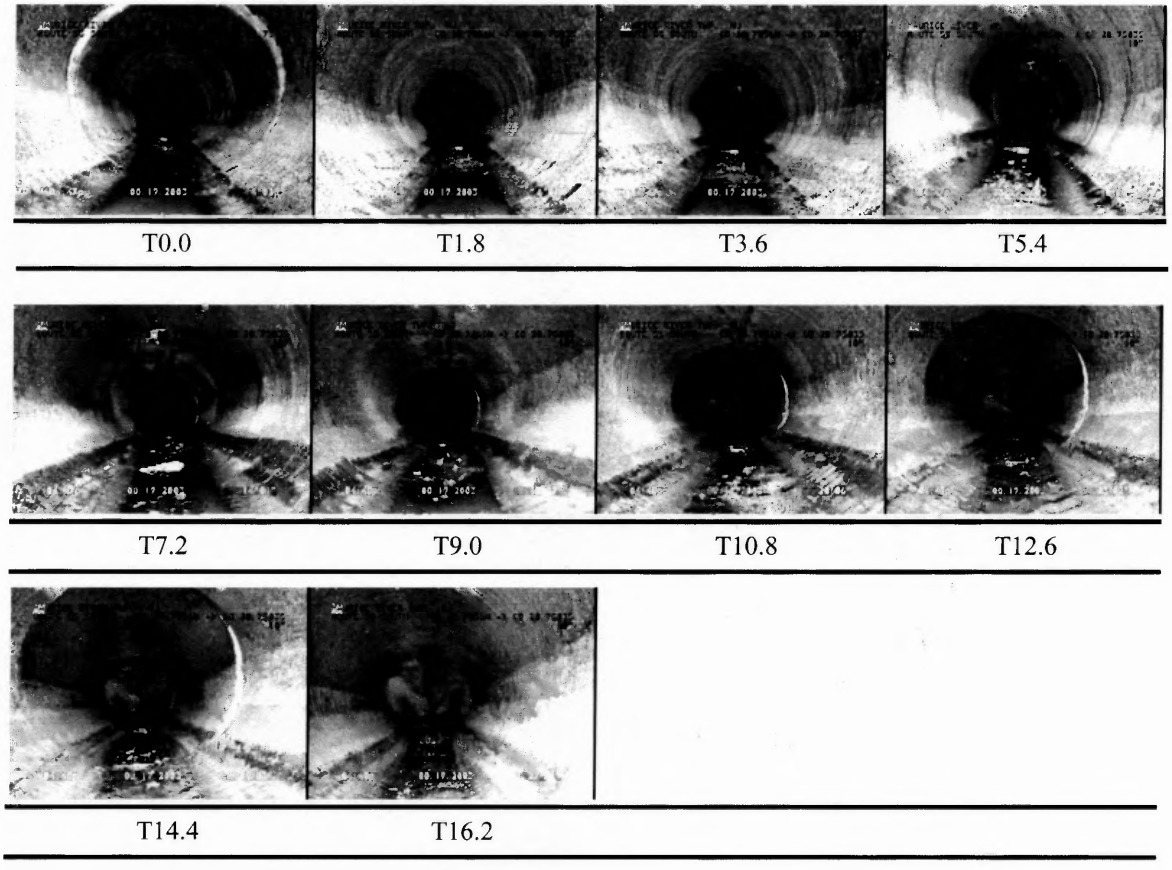


Figure 5.1 Ten Frames analyzed with a 1.8 sec time gap (T = 1.8 s).

The origin of the coordinate system used in standard image analysis is located at the upper left corner of the image. In the proposed methodology a T-shaped region as

shown in Figure 5.2 is blocked out (whitened portion). The retained regions are symmetric of size p x q, and are located on the left and right bottom corner of the image. For an image of size n x m, a rectangular portion of size n x (m - q) is blocked out in the upper part of the image, and a rectangular portion of size (n - p) x q is blocked out in the lower middle part of the image directly below the other blocked out portion. This results in two sub-images – for identification purposes these sub-images are referred to as Left (L) and Right (R). While the pixel coordinate system of the original image is located at the upper left corner, the two sub-images employ a slightly different definition of coordinate systems. The left sub-image has its coordinate system O_L at the bottom left corner, the right sub-image has its coordinate system O_R at the bottom right corner. This is done for the ease of mathematical operations and interpretation, and is shown below.

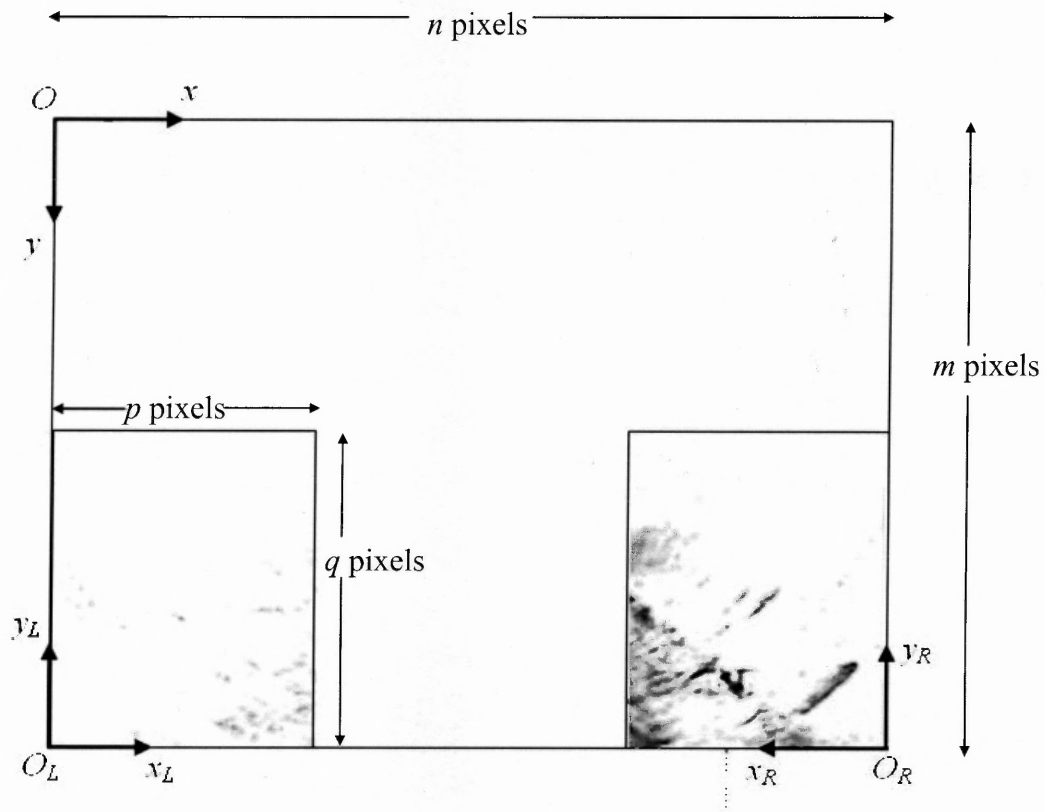


Figure 5.2 Left (L) and Right (R) sub-images and their coordinate systems.

For easier and quick implementation of the proposed methodology, the RGB sub-images are then converted into 8-bit grayscale images, where each pixel is represented on a scale of 0-255 gray shades. This is done using the standard Photoshop guideline of 30% red, 59% green and 11% blue for every pixel and every plane in the RGB image. In this experiment, the above simple format seems to produce quick and effective grayscale representations.

To enhance the contrast in the sub-images, the proposed methodology implements an adaptive Ridler-Calvard scheme [125] to choose a threshold gray-value, which provides a consistent framework based on the image properties (compared to an image independent threshold criterion). The scheme is described in brief below and has been shown to produce better results than the standard histogram equalization methods for thresholding. The histogram-based methods are inconsistent and require manual intervention to determine a suitable *cutting* level. The Ridler-Calvard scheme chooses the threshold value by the process of relaxation. It first calculates the mean gray-value t_o and, thresholds the image at this mean value

$$t_o = \sum_{x,y} f(x,y)/N, \tag{5.1}$$

where $f(x,y)$ is the gray-value at the (x,y) location and N is the total number of pixels in the image. For the p^{th} iteration, the threshold value, t_p, is calculated as

$$t_p = \frac{1}{2}\left[\frac{\sum_{x,y} f(x,y)T_{p-1}(x,y)}{N_{p-1}} + \frac{\sum_{x,y} f(x,y)\overline{T_{p-1}(x,y)}}{N - N_{p-1}}\right], \tag{5.2}$$

where $T_{p-1}(x,y)$ is a binary image resulting from thresholding the image at t_{p-1} and N_{p-1} is the number of *on*-pixels in the binary image $T_{p-1}(x,y)$. The algorithm stops when the threshold value converges; in most of the experiments here, the threshold value

converges after 4-5 iterations. This scheme is easily implementable and consistently produces deep-contrast images. All pixels below the threshold t_p are retained in their original form and those above are converted to white as

$$f_T(x,y)=\begin{cases} f(x,y) & \text{if } f(x,y)<t_n \\ 255 & \text{otherwise.} \end{cases} \tag{5.3}$$

The resultant image after preprocessing is a crisp deep contrasted grayscale image of lower dimensionality than the original RGB image. This converted image is then segmented using a two-stage fuzzy segmentation procedure, which is the mainstay of the proposed methodology.

5.2.2 Image Segmentation

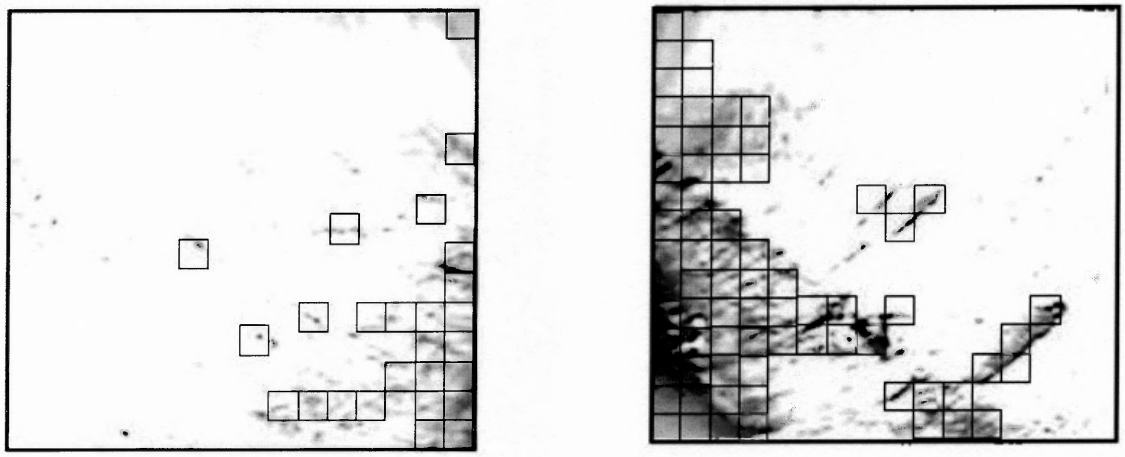
The segmentation is carried out as a two-step fuzzy clustering scheme. The aim is to identify shapes characterized by distinctive grayscale features – individual shapes are identified as clusters. Fuzzy clustering (and all unsupervised clustering) finds clusters in data even if there are no real clusters to be found; all clustering procedures assume a pre-defined value for the number of clusters to be identified. This provides the motivation to use a two-step clustering procedure – the first preliminary step provides information about the existence (or non existence of clusters), while the second step detects clusters if they are present. To ascertain the presence of clusters, one could even get a count of dark pixels and then identify zones of high density in the dark regions. This approach, however, is less preferred compared to a quick unsupervised clustering to identify interesting shapes. In the first step, the image is segmented into two clusters, viz. foreground and background. The foreground is defined as regions of interest in the image – these regions are *most likely* to have defect shapes. The background is defined

as the less intense gray region, which can be discarded from future analysis. This includes regions of small dark blobs and other non-interesting features that can also be discarded.

The image segmentation process is carried out separately for the two sub-images produced after preprocessing. In the first segmentation step, the preprocessed sub-image is divided into consecutive row-column square blocks of a x a pixels. For each block, a 3-D feature vector is defined for clustering – block gray-value mean, block gray-value standard deviation, and inter-block gray-value gradient. The mean and standard deviation are properties of the block of pixels under consideration, while the gradient is a "neighborhood" property, which relates a block to its neighboring blocks. The gradient of a block specified by mean location (x, y) is given by

$$g(x,y) = \frac{\left[\sum_{i=1}^{8} f(x,y) - f(x_i,y_i)\right]/8}{255}, \tag{5.4}$$

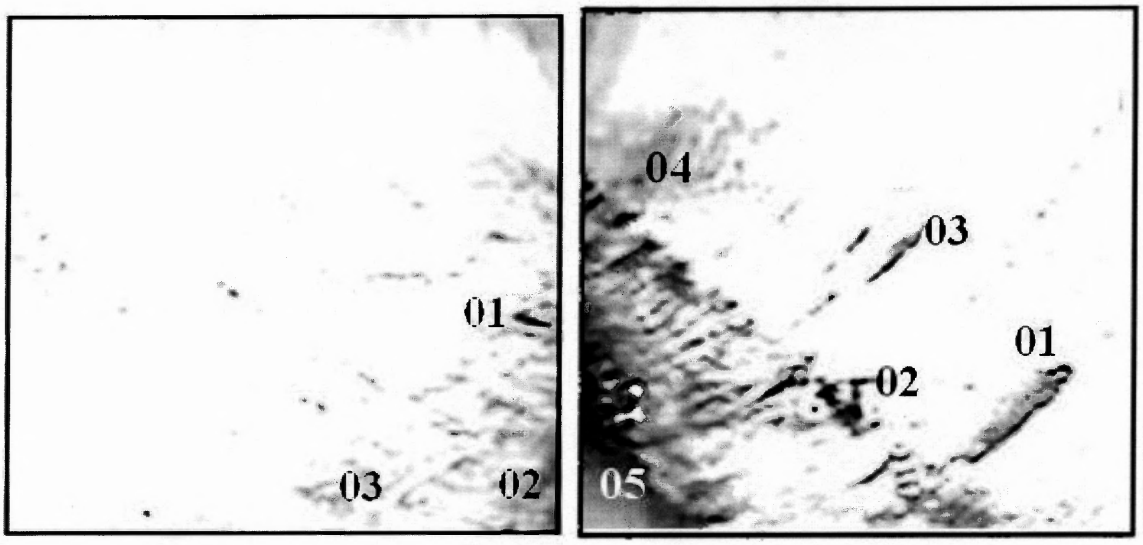
where $f(x_i,y_i)$ is the mean gray-value of the i^{th} neighboring block. A block (other than edge or corner blocks) has eight neighboring blocks. The gradient is scaled over the 256 grayscale values. For each sub-image, if there are P total blocks, then FCM is implemented on a feature set of size P x 3, with the usual user-defined parameters. Foreground blocks ($< P$) are identified as feature vectors characterized by low mean, low standard deviation and similar gradient. Some images might have no faults. Hence, if the number of foreground blocks identified is less than a certain threshold, then the image is dropped from further inquiry. The threshold chosen is small, such as five foreground blocks. Results of clustering to uncover foreground blocks in two sample sub-images are shown in Figure 5.3.



(a) (b)

Figure 5.3 First step segmentation results using fuzzy clustering (foreground is shown as blocks, exaggerated in size) - (a) 66 foreground blocks identified in T1.8-L, (b) 108 foreground blocks in T1.8-R.

In the second step, each of these foreground blocks is broken down to the elementary pixel level, and another feature vector for identifying specific shapes is constructed. This new feature vector is comprised of pixel x-location, pixel y-location, and pixel gray-value. If P_f foreground blocks (out of P total blocks in each sub-image) are identified in the first step, then the data-set for the second stage of segmentation is of the size $n = P_f a^2$ and each of these is represented by a 3-D vector. As opposed to the first step, the number of clusters to be found in the second step is not known beforehand. Strictly speaking, this step is not a segmentation process, but rather a shape detection procedure. The optimal number of extracted shapes is then calculated simultaneously by plotting two cluster validation indices F and H (defined in Chapter 3), over the range of *values*.



(a) (b)

Figure 5.4 Classification of defect shapes based on fuzzy clustering – (a) three defects identified in T1.8-L, and (b) five defects identified in T1.8-R.

Defect ID	Surface Area (pixel^2 x 10^3)	Mean Location x	y	Scaling factor	Corrected Area (pixel^2 x 10^3)	Smallest Grayvalue (mean)	Depth Severity 1 = Deep 0 = Shallow
T0.0-R-0X							
T1.8-L-01	10.5	271	145	1.846875	19.3921875	0	1
T1.8-L-02	6.2	282	36	1.88125	11.66375	12	0
T1.8-L-03	3.9	221	41	1.690625	6.5934375	15	0
T1.8-R-01	19	95	105	1.296875	24.640625	0	1
T1.8-R-02	12.9	215	112	1.671875	21.5671875	0	1
T1.8-R-03	8.1	172	175	1.5375	12.45375	0	1
T1.8-R-04	21.8	282	245	1.88125	41.01125	27	0
T1.8-R-05	39.2	305	92	1.953125	76.5625	17	0
T3.6-L-01							

Figure 5.5 Snapshot of the database with corrected (scaled) defect area in pixel^2 and depth severity information

At the optimal number of clusters, F takes a maximum value, and H takes a minimum. Defects shapes identified for the two sub-images using the two-step clustering procedure are shown in Figure 5.4. The optimal shapes identified for every sub-image are then stored in a data-base as a function of some of its basic characteristics – unique identification label as a function of the sub-image, size of the defect shape in pixels, and

location in sub-image identified by the mean x- and the mean y- coordinates. A snapshot of the database is shown in Figure 5.5, and will be discussed in the next sections.

5.3 Framework for Condition State Assessment

The defects identified by image analysis need to be collated to provide information about the condition state of the pipe. A simple and efficient framework is presented in this section. Techniques such as backpropagation neural networks and fuzzy learning can be especially useful here because this stage involves imprecise information and absence of well-identified guidelines or documentation. A suitable learning algorithm, which models itself on known information, can be a useful tool for automation. A simple rule-based system, which attempts to correlate the severity of the most severe defect to the condition state, is presented in this research. Severity of a defect is a function of the surface area of the defect (average size) and the percentage loss of wall thickness at the defect location (average depth). From the defect shapes identified in the last section, the physical size of the defect can be easily calculated by using a suitable scaling and/or correction factor for perspective, which translates size information in pixels to physical size information in cm^2 or $inch^2$. However, extracting depth information from 2-D images has always been a problem. In the proposed methodology, a simple pixel grayscale mapping approach for an effective and quick approximation of the average depth at these defect locations is proposed.

The underlying assumption in this approach is that depth is manifested as darker-than-usual pixels in the image when compared with those of the boundary. A dark pixel could be the result of several other factors – improper light conditions, a black patch left

by sediments or the flow, previous repair patch-work etc. However, if the shapes are identified close to the edges of the frame and are categorized as "defects", then dark pixels along the center of the defect can be attributed to the depth of the defect. A well-documented study characterizing actual depth of a defect and corresponding gray-values on the image needs to be done before the theory can be used practically. However, this is beyond the scope of the present study and can a topic of future research. For the proposed methodology, it is assumed that such a well-documented depth-pixel relationship exists.

The exact physical depth of a defect (in inches) is not critical at this stage of model development – the methodology relies on identifying linguistic labels for defects. A dichotomous labeling scheme would have been the simplest, i.e., one that labels defects as deep and not-deep. However, a more detailed 6-tier depth labeling system – gray-value between 0-5 is defined as *very deep*, 5-10 *deep*, 10-15 *not too deep*, 15-20 *not too shallow,* 20-25 *shallow*, and greater than 25 *very* shallow, is presented in this research. A quantitative scale may be developed in future research. The average gray-value in and around the center of the defect is a direct indicator of the depth. The average gray-value of the 10 darkest pixels in a 5 x 5 pixel block centered on the mean *(x, y)* location of a defect is used in this study.

For every identified defect, a record in the database is created, which stores the linguistic depth information and the surface area (in cm^2 or $inch^2$). The database can then be consolidated after all relevant images have been analyzed and segmented, and all defects identified. Frames extracted from the video need to be skipped in an organized manner so as to avoid finding the same defects over and over again. The segmentation

procedure lends itself well to identifying defects near the periphery of the image where they are readily apparent both to the naked eye (of the operator), and to the automation process (of the proposed methodology). The size of the retained sub-image is a parameter that can be used to determine the number of frames to be skipped. The motivation is that defects should not be identified more than once. Hence, the next frame to be analyzed should not contain any part of the preceding sub-images. This time gap can be easily approximated if the assumptions that the camera moves in a straight line with a constant speed, indeed hold true. Let the time gap between successive non-overlapping frames be T seconds, and if the camera captures the video at 30 fps, then the number of frames to be skipped between analyses is 30 times T. After skipping, the next non-overlapping frame can then be analyzed, and the process repeated until the end of the pipe is reached. A more comprehensive (and conservative) approach would be to analyze a few frames that produce *almost* similar sub-images, e.g., analyze 10 successive frames within the $30T$ time period. This provides a method to verify and consolidate the results of image segmentation, because these 10 successive frames, more or less, look at the same part of the pipe. In the next section, the results of the quick and less conservative approach are presented. Once all the defects are located (with no defect identified more than once, and no defect overlooked), the database is complete. The database can be analyzed in many different ways – in the proposed methodology, a simple rule-base analysis is used. Condition states based on a 4-point scale, i.e., 1 through 4, are defined. The definitions are given below, with emphasis on Repair, Rehabilitation or Replacement (R^3) actions,

Condition State 1: There is no evidence of section loss and loss of structural integrity and suggested corrective action would be to do nothing.

Condition State 2: Minor section loss less than or equal to 10% of total internal surface area. Structural integrity not compromised and suggested corrective action would be to schedule next inspection.

Condition State 3: Section loss is between 10 to 30% of the total internal surface area or appreciable deterioration of structural integrity and suggested corrective action would be to take quick R^3 decision.

Condition State 4: Section loss is greater than 30% of the total internal surface area or structural integrity compromised and suggested corrective action would be to implement R^3 immediately.

The surface area and depth information are treated separately for reasons of simplicity. The total internal surface area of the pipe is calculated, based on length and internal diameter. This is then compared to the total surface area of all defects combined together. If S is the total internal surface area of the pipe, and S_d is the internal area covered by defects, the surface area ratio is defined as

$$R = \frac{S_d}{S} \tag{5.5}$$

The R value less than 0.01 is considered Condition State 1, between 0.01 and 0.1 Condition State 2, between 0.1 and 0.3 Condition State 3, and greater than 0.3 Condition State 4 [126], [127]. This rating scheme can then be modified if depth labels are known. This is done by using additional information about the number of defects that fall into either *very deep*, *deep*, or *not too deep* types. If all the defects are either *very shallow*, *shallow*, or *not too shallow*, then above Condition States can be used in its original form, with the contingency that shallow defects (in their present form) are not a threat to the structural integrity of the pipe. In other words, the surface area of defects is more important than the depth if all the defects are shallow. By shallow, it is assumed that all the three types – *very shallow*, *shallow* and *not too shallow* are included. However,

condition ratings needs to be modified if some of the defects identified are deep. The flowchart in Figure 5.6 illustrates a conceptual aggregated methodology.

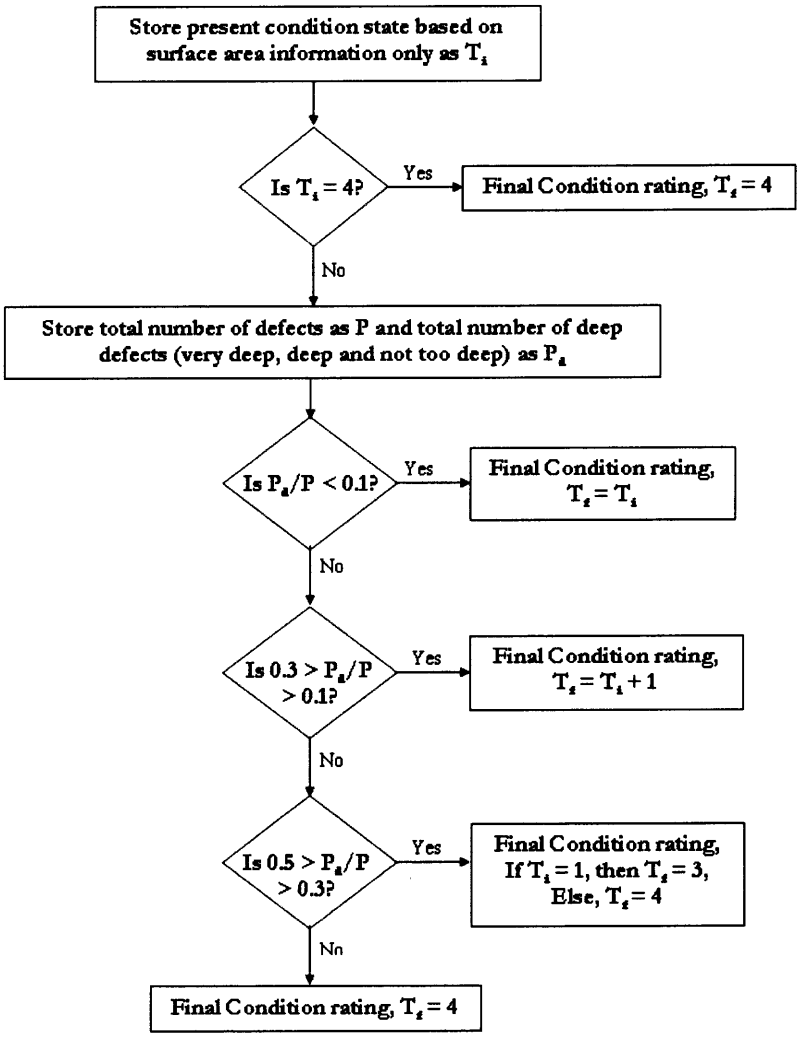


Figure 5.6 Combining depth and surface area information to obtain final condition state rating.

5.4 Results

In this section, results of a sample run using the proposed methodology of condition state assessment based on image segmentation are presented. An 18 sec mpeg video is analyzed using a time gap of T = 1.8 sec. The necessary assumptions about constant camera trajectory and speed are not fulfilled here, and hence, some manual adjustments had to be made. This however, does not affect the methodology in any way. Frames are analyzed at 1.8 sec intervals, i.e., one frame in every 54 frames is analyzed. The frames are shown on a time scale in Figure 5.1. Each frame is an RGB image of size 960 x 720 pixel^2. The images are first reduced in size to produce two sub-images with p = 320 and q = 360. Hence, the two sub-images are of size 320 x 360 pixel^2, and are located on the left and right sides of the original frame. The sub-images are tagged by a label for easy identification. The original image is identified by its position on the time size – the first image on the series will be T0.0, followed by T1.8, and so on. The sub-images are identified as Left (L) and Right (R); this helps in easy interpretation of results. The sub-images are then converted into 8-bit grayscale images, followed by a contrast enhancement using the Ridler Calvard thresholding scheme. Figure 5.2 shows the sub-image T1.8-L and T1.8-R, and their coordinate systems, after all the preprocessing steps have been carried out.

The sub-images are then divided into square blocks with a = 10. Other values of a, such as a = 20, and a = 5, were also examined, and a = 10 was found to produce better results compared to the other two. For the 320 x 360 pixel^2 sub-images, there are 1152 such square blocks. The block grayscale mean and the block grayscale standard deviation are computed within the 100 elements of each block, and the intra-block

gradient is calculated. For blocks on the edges and corners of the sub-image the formula is modified during run time to count only three neighborhood blocks (for corner blocks) and five neighborhood blocks (for blocks on the edges). The sub-images are clustered and only the blocks in the foreground clusters are retained. The foreground cluster is comprised of blocks with low mean, low standard deviation, and almost equal gradient. The background blocks are discarded from future analysis. For T1.8-L and T1.8-R shown in Figure 5.3(a) and (b) respectively, the first step of clustering identifies 66 and 108 foreground blocks among 1152 blocks each. These also include darker regions on the pipe surface left behind by sedimentation and water level marks. The water level mark usually produces a straight-line pattern, which can be instantly discarded from future analysis. Other marks manifest themselves as random blobs, and these are not recognized as large shape clusters in the second step of fuzzy clustering.

In the second step, three large shape clusters are identified in T1.8-L, and five large shape clusters are identified in T1.8-R. These are verified during run time by simultaneously plotting the cluster validation measures, and are shown in Figures 5.4(a) and (b). A snapshot of the database created after image segmentation and classification is shown in Figure 5.5. The defects are labeled as T1.8-L-01, T1.8-L-02, and so on. Characteristics such as size, location, and mean of the 10 darkest center pixels, are stored in the database. An appropriate scaling factor is used to correct for perspective, and another factor is used correlate surface area in pixel^2 to surface area in cm^2 or inch^2 (not shown in the database). The two factors can also be combined into one. While the perspective correction factor depends on the location, the correlation factor is a constant. The perspective correction factor (labeled "scaling factor") depends only on the mean x-

location of the defect. For a shape near the outside edges of the sub-image, the scaling should be less intense, but for a defect shape near the inside edge is more compressed (due to perspective) and hence, requires a larger scaling. Let $x(m)$ denote the mean x-location of the defect, then a typical scaling factor is defined as

$$SF = \frac{x(m)}{p} + 1 \tag{5.6}$$

The exact determination of these factors is out of the scope of this work, but has been conceptually explained and studied. Experiments and learning algorithms can be used to specify and implement suitable correction and correlation factors.

5.5 Discussion and Conclusions

The automated methodology proposed here attempts to bridge the gap between image analyses used to identify internal defects in pipelines and the subsequent condition state assessment analysis. As of now, no such fully automated system exists in theory or practice. The study presented here can be used as a starting point to formulate better methodologies, which then can be directly put into practice in the field. The methodology is tested only on a select group of images, using a C program developed in-house. It relies heavily on learning from previous experience, especially when deciding (1) how to skip consecutive frames, (2) how to relate pixel area information to physical surface area information, (3) how to correlate pixel gray-values to actual observed depth in defects, and (4) how to choose a correct perspective scaling and lighting correction factors. Further work is needed to formulate a set of rules, which will lend itself to easy automation and take care of the above issues.

CHAPTER 6

SUMMARY AND CONCLUSIONS

6.1 Conclusions

The primary objective of this dissertation is to propose novel methods in the fields of robust fuzzy clustering and cluster validation. A secondary objective is to prove the applicability of fuzzy clustering as a classification tool for problems in computational chemistry and image analysis.

It is well known that Fuzzy c-Means (FCM) and its variants, based on the minimization of the least squared error functional, are very susceptible to noise. Attempts made to robustify FCM almost always deal with a modification of the objective functional; however, several other robust techniques have been proposed which use statistical estimators and robust error functionals, and these have been discussed in brief in Chapter 2. For improving robustness, two novel fuzzy clustering procedures have been proposed in this work. The first implements a modified version of the FCM that uses a membership scaling function to achieve robustness. The function depends on the distances of the patterns from cluster centers and it has been shown that for outliers, repeated scaling results in large reduction of memberships in good clusters. The methodology is compared to the concept of noise cluster, and parallels are drawn between the two. The scheme is implemented on test data-sets and results compared with results on the same data reported in literature. The other novel scheme proposed in this dissertation implements a robust least trimmed squares estimator based on a feasible solution technique from the field of regression analysis. The least trimmed squares

estimator is a regressor that generates a best-fit line on a trimmed data-set. However, there exists no closed form solution scheme to implement the least trimmed squares clustering (or regression). The feasible solution scheme which implements the LTS fit is used along with FCM and is shown to impart robustness. The resulting partitioning scheme is called the Feasible Solution – Fuzzy Least Trimmed Squares (FS-FLTS) clustering. Several noisy data-sets from the literature have been tested using FS-FLTS, first under an assumption that the amount of contamination is known *a priori* and later with an unknown amount of contamination. The results of both MC and FS-FLTS schemes have been very encouraging. The contributions made here include, (1) development of easily-interpretable and implementable schemes, and (2) integration of a high breakdown robust statistical estimator into the standard FCM procedure.

The validation of partitions produced by clustering is also investigated in detail in this dissertation. A thorough review of validation procedures used in fuzzy and non-fuzzy clustering domains is presented in Chapter 3. A major drawback of almost all validation procedures is the lack of physical interpretability of the numerical results. Most validity schemes try to maximize or minimize a function which is related to the geometry of the partitions (size of clusters), or is related directly to assignment information (memberships), or a combination of the two. They rank the partitions by assigning each partition with a numeric label; however, it is only the nominal label that counts (the best partition is either the one with the smallest or the largest numeric value). The cluster validation scheme proposed in this dissertation checks for the validity of the partition by implementing a test of random position hypothesis. The concept is borrowed from the field of clustering tendency and its applicability in validating clusters is shown

here. The null hypothesis is either accepted or rejected based on the value of the Hopkins statistic, a function that characterizes the randomness within each cluster of every partition. The statistic is shown to take a value very close to 0.5 for extremely random clusters. The underlying assumption here is that natural clusters are random within themselves. If a cluster is structured (if there are clusters within this cluster), then it is not a natural cluster and hence, can be partitioned further. The test of random position hypothesis also assigns a numerical value to every partition (mean and variance of the Hopkins statistic over the c clusters). The optimal number of clusters is identified as the c where the mean Hopkins statistic takes a value close to 0.5 and the variance is close to zero. The numerical values now have significance and a definite meaning unlike other cluster validity schemes. The statistic also has the power to differentiate between completely random, highly structured and non-random, equally-spaced data.

Over the years, there has been considerable development in theories and methodologies in the field of fuzzy clustering. However, there is an unnatural dearth of applications and unless theories are applied in practice to solve real world problems, the field of research would not be self-sustaining. The recent versions of MATLAB include the Fuzzy Logic Toolbox which contains a GUI implementation of FCM. There are many opensource routines that implement many of the well-known clustering procedures but development of specific applications based on these routines are relatively hard to come by. A few applications of fuzzy clustering are mentioned in Chapter 1. In this dissertation, two elaborate applications are presented.

In Chapter 4, a conformational analysis protocol is presented with emphasis on feature extraction, dimensionality reduction and fuzzy clustering. The objective of the

study is to classify over 700 conformers of a drug molecule into groups based on similarity in structure. Two novel feature extraction methodologies are proposed here – one consists of a set of selected atom locations and molecular planes, and the other comprises of a set of a selected pair of molecular planes. For ease of clustering and interpretation, the highly flexible molecule is superimposed on a fixed plane and conceptually divided into left and right sides. These sides are clustered separately using the newly-developed Fuzzy Relational Clustering (FRC) procedure. Unlike object-based clustering, relational clustering takes a proximity (dissimilarity or distance) matrix as in input. Clusters extracted using FRC are validated using cluster validity measures from the literature. The representative conformers found based on a structural proximity criteria, not only appear to be structurally distinct (occupying distinct regions in space), but also show a separability based on potential energy [107]. The clusters and the representatives are also in accordance with physically explainable phenomena, such as bond rotations etc.

From the point of view of cluster analysis, this study is novel and intellectually challenging – very few practical applications of large data fuzzy clustering exist in the literature. The complete cluster analysis protocol (like software development lifecycle) has been followed here, starting with intelligent feature extraction and concluding with a physical interpretation of clustering results. Any clustering procedure can partition a data-set (given a good reliable feature vector for clustering); the data could be a structured data or the data could be a random collection of entities. This study also raises an important clustering tendency question – *why find clusters and representatives when there are none to be found*? This is better illustrated in the B-side and full-molecule

clustering results of the molecular planes feature set. The visualization plots do not show the existence of clusters (however, when clustering on a 6-D space is visualized in a 2-D or 3-D space, there are bound to be inaccuracies; this is not to say that the clustering results are in error). However, even if it is assumed at this stage that there were no *natural clusters* to be found on the B-side or the full-molecule, FRC and subsequent cluster validity plots indicate the presence of three and five clusters respectively. Even if the data is completely random and there is no inherent substructure, clustering would give meaningful results within the bounds of randomness. A circular blob of data points can be clustered into four groups by drawing two diameter lines along the X and Y axis; the representatives found would still be the most dissimilar from each other. The same blob can also be divided into five clusters but obviously, four is a better partition than five in this case. Hence, even if there is no substructure to be found, clustering would produce partitions and validity measures would indicate (within the range of c) the so-called optimum value of c (in the case of the blob, $c = 4$). This value of c may or may not partition the data into *natural groups*, but given the limitations it certainly would produce the best partition. And if the partition is the best, the corresponding representative structures will also be the most dissimilar amongst each other. This is the bottom line of the analysis – the chemist in charge of further CoMFA studies is hardly interested in natural groups, she[1] only insists that she be given as distinct conformers as can be found, and that she does not intend to test more than a certain fixed number of conformers. (This decides the range of c to be tested, assumed either 12 or 14 in the study.)

In Chapter 5, an image analysis application is presented as a part of a broader project involving automated structural condition state assessment of pipelines. The

[1] Used in a non-gender context.

framework developed however is not unique to pipelines or underground infrastructure and can be extended to any infrastructure that requires repair, replacement, or rehabilitative maintenance based on acceptable (or rather unacceptable) structural conditions. The images considered in this study are extremely noisy. The image analysis implements a two-stage FCM procedure to mitigate the effects of noise. A single stage robust clustering algorithm can also be used instead of a two stage FCM. However, FCM has many desirable properties and these properties can be used to ones advantage when there is considerable knowledge about the data-set. In the study reported in Chapter 5, although the state of the pipe was unknown, the nature of the images was known beforehand. In other words, it was assumed that the representation in the image is known – the center is occupied by the dark pipe-end, the top and the bottom-center of the image contain textual legends, the only areas of interest are the left and the right lower halves of the image etc. A little knowledge such as this can then be used to extract features intelligently so that noise and inaccuracies can be handled. This is indeed a novel approach compared to other fuzzy and non-fuzzy segmentation approaches found in literature.

This dissertation looks at the domain of cluster analysis both from theoretical and application point of view. In each, problems have been identified which have then been approached from with a fundamental *first-principle* frame of mind. This is not to say that the wheel was reinvented every time a wheel was required. Usable concepts have been borrowed from other domains and have been successfully applied or integrated into already existing methodologies, yet using a fundamentally distinct approach.

6.2 Future Research Directions

There are several issues that remain unresolved in cluster analysis. There are also many issues that should have been covered (or researched and elaborated) in this dissertation but have not been because of a myriad of different reasons. Such unresolved issues form the crux of the future research thrust.

Clustering and development of clustering procedures have received undue importance (and not without a reason) compared to feature extraction, clustering tendency and, cluster validation studies. A clustering scheme is easy to formulate mathematically even if few have closed form solutions. However, feature extraction and cluster validation still remain inexact sciences. It requires as much of a left brain, as right, to formulate the *best* feature set or to extract truly *wonderful* features to cluster a data-set on. It can well be debated if feature extraction can ever be automated. Because of this vagueness in interpretation and representation, feature extraction is a field which would gain immensely from the incorporation of fuzzy sets. For an introduction to fuzzy feature extraction the reader is referred to [128] and [129].

The issue of natural vs. artificial clusters, investigated in brief in the last section, provides another interesting perspective to cluster validation. The cluster validity plots presented in Chapter 4 almost all follow a curious trend – in the (perceived) absence of natural groups, the plots for F, F', and H are all either monotonically decreasing (F and F') or monotonically increasing (H). This leads to the hypothesis that if the objective is to uncover natural groups, then one needs to look at these three measures first. Modeling of this behavior is a topic of future research.

A related issue is that of visualization of data. The need for clustering data partly arises because data in higher dimensions are almost impossible to visualize. There are also certain feature spaces that are hard to visualize. A parallel focus of research should be better data visualization techniques and automation of data processing tasks based on data visualization. Data visualization tools can also be helpful in validation results of clustering.

REFERENCES

1. Ackermann, R., *An Introduction to Many-Valued Logics*. London: Routledge and Kegan Paul, 1967.

2. Rescher, N., *Many-Valued Logic*. New York: McGraw Hill, 1969.

3. Zinovev, A. A., *Philosophical Problems of Many-Valued Logic.* Dordrecht, The Netherlands: Reidel, 1963.

4. Mamdani, E. H., Assilian, S., "An experiment in linguistic synthesis with a fuzzy logic controller," *International Journal of Man-Machine Studies*, vol. 7(1), pp. 1-13, 1975.

5. Ruspini, E., "A new approach to clustering," *Information and Control*, vol. 15, pp. 22-32, 1969.

6. Roubens, M., "Pattern classification problems and fuzzy sets," *Fuzzy Sets and Systems*, vol.1, pp. 239-253, 1978.

7. Bezdek, J. C., *Pattern Recognition with Fuzzy Objective Function Algorithms*. New York: Plenum, 1981.

8. Bellman, R., Zadeh, L. A., "Decision making in a fuzzy environment," *Management Science*, vol. 17, pp. B141-B164, 1970.

9. Zadeh, L. A., "Soft computing and fuzzy logic," *Software IEEE*, vol. 11(6), pp. 48-56, 1994.

10. Wiedey, G., Zimmerman, H. -J., "Media selection and fuzzy linear programming," *Journal of the Operations Research Society*, vol. 29, pp. 1071-1084, 1978.

11. Zadeh, L. A., "The concept of a linguistic variable and its application in approximate reasoning," *Memorandum ERL-M 411*, Berkeley, 1973.

12. Baldwin, J. F., "A new approach to approximate reasoning using a fuzzy logic," *Fuzzy Sets and Systems*, vol. 2, pp. 309-325, 1979.

13. Bharitkar, S., Kyriakakis, C., "A cluster centroid method for room response equalization at multiple locations," in *IEEE Workshop on Applications of Signal Processing, Audio and Acoustics*, 2001, pp. W2001.1-W2001.4.

14. Marzouk, M., Moselhi, O., "On the use of fuzzy clustering in construction simulation," in *Winter Simulation Conference,* 2001, pp. 1547-1555.

15. Zadeh, L. A., "Fuzzy Sets," *Information and Control*, vol. 8, pp. 338-353, 1965.

16. Goguen, J. A. "L-Fuzzy sets," *Journal of Mathematical Modelling and Algorithms*, vol. 18, pp. 145-174, 1967.

17. Goguen, J. A., "The logic of inexact concepts," *Synthese*, vol. 19, pp. 325-373, 1969.

18. Dubois, D., Prade, H., "Fuzzy real algebra: Some results," *Fuzzy Sets and Systems,* vol. 2, pp. 327-348, 1979.

19. Dubois, D., Prade, H., *Fuzzy Sets and Systems: Theory and Applications*. New York: Academic Press, 1980.

20. Sugeno, M., "Measures and fuzzy integrals – A survey," in *Fuzzy Automata and Decision Processes*, Gupta, M. M., Saridis, G. N., Gaines, B. R., Eds. New York: North-Holland, 1977, pp. 89-102.

21. Kolmogorov, A., *Foundations of the Theory of Probability*. New York: Chelsea, 1950.

22. Zimmermann, H. -J., *Fuzzy Set Theory and Its Applications*. Dordrecht, The Netherlands: Kluwer-Nijhoff, 1985.

23. Shafer, G., *A Mathematical Theory of Evidence*. Princeton, New Jersey: Princeton Univ. Press, 1976.

24. Pawlak, Z., "Rough sets," *International Journal of Computer and Information Sciences*, vol. 11, pp. 341-356, 1982.

25. Shackle, G. L. S. *Decision, Order and Time in Human Affairs*, London, U.K.: Cambridge Univ. Press, 1961.

26. Zadeh, L. A., "Fuzzy sets as a basis for a theory of possibility," *Fuzzy Sets and Systems,* vol. 1, pp. 3-28, 1978.

27. Bezdek, J. C., "Fuzzy models – What are they and why?," *IEEE Transactions on Fuzzy Systems*, vol. 1(1), pp. 1-6, 1993.

28. Bezdek, J. C., "The thirsty traveler visits Gamont: A rejoinder to "Comments on fuzzy sets – What are they and why?"," *IEEE Transactions on Fuzzy Systems,* vol. 2(1), pp. 43-45, 1994.

29. Wang, P. -C., Leou, J. -J., "New fuzzy hierarchical algorithms," *Journal of Information Science and Engineering*, vol. 9(3), pp. 461-489, 1993.

30. MacQueen, J., "Some methods for classification and analysis of multivariate observations," in *Fifth Berkeley Symposium on Mathematical Statistics and Probability*, 1967, pp. 281-297.

31. Jain, A. K., Dubes, R. C., *Algorithms for Clustering Data*. Englewood Cliffs, New Jersey: Prentice Hall, 1988.

32. Jolion, J. -M., Rosenfeld, A., "Cluster detection in background noise, *Pattern Recognition*, vol. 22, pp. 603-607, 1989.

33. Weiss, I., "Projective invariants of shapes," in *DARPA Image Understanding Workshop*, Cambridge, MA, 1988, pp. 1125-1134.

34. Davé, R. N., "Characterization and detection of noise in clustering," *Pattern Recognition Letters*, vol. 12, pp. 657-664, 1991.

35. Davé, R. N., Sen, S., "On generalizing the noise clustering algorithms," in *Proceedings of the 3rd Conference of the International Fuzzy Systems Association (IFSA III)*, 1997, pp. 205-210.

36. Keller, A., "Fuzzy clustering with outliers," in *Proceedings of the 19th Conferences of the North American Fuzzy Information Processing Society (NAFIPS 2000)*, 2000, pp. 143-147.

37. Davé, R. N., Krishnapuram, R., "Robust clustering methods: A unified view," *IEEE Transactions on Fuzzy Systems*, vol. 5(2), pp.270-293, 1997.

38. Huber, P. J., *Robust Statistics*. New York: Wiley, 1981.

39. Kersten, P. R., "Fuzzy order statistics and their application to fuzzy clustering," *IEEE Transactions on Fuzzy Systems*, vol. 7(6), pp. 708-712, 1999.

40. Łęski, J., "Towards a robust fuzzy clustering," *Fuzzy Sets and Systems*, vol. 137, pp. 215-233, 2003.

41. Siegel, A. F., "Robust regression using repeated medians," *Biometrika*, vol. 69, pp. 242-244, 1982.

42. Rousseeuw, P. J., "Least median of squares regression," *Journal of the American Statistical Association*, vol. 79, pp. 871-880, 1984.

43. Rousseeuw, P. J., Leroy, A. M., *Robust Regression and Outlier Detection*. New York: Wiley, 1987.

44. Kim, J., Krishnapuram, R., Davé, R. N., "Application of the least trimmed squares technique to prototype-based clustering," *Pattern Recognition*, vol. 17, pp. 633-641, 1996.

45. Beni, G., Liu, X., "A least biased fuzzy clustering method," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16(9), pp. 954-960, 1994.

46. Krishnapuram, R., Keller, J., "A possibilistic approach to clustering," *IEEE Transactions on Fuzzy Systems*, vol. 1, pp. 98-110, 1993.

47. Krishnapuram, R., Keller, J., "Fuzzy and possibilistic clustering methods for computer vision," in *Neural Fuzzy Systems*, Mitra, S., Gupta, M., Kraske, W., Eds. pp. 133-159, 1994.

48. Pal, N. R., Pal, K., Bezdek, J. C., "A mixed c-means clustering model," in *Proceedings of the 6th IEEE International Conference on Fuzzy Systems (FUZZ-IEEE'97)*, 1997, pp. 11-21.

49. Chintalapudi, K. K., Kam, M., "A noise-resistant fuzzy c-means algorithm for clustering," in *Proceedings of the 7th IEEE International Conference on Fuzzy Systems (FUZZ-IEEE'98)*, 1998, pp. 1458-1463.

50. Krishnapuram, R., Keller, J. M., "The possibilistic c-means algorithm: Insights and recommendations," *IEEE Transactions on Fuzzy Systems*, vol. 4, pp. 385-393, 1996.

51. Wie, L. M., Xie, W. X., "Rival checked fuzzy c-means algorithm," *Acta Electronica Sinica*, vol. 28, pp. 63-66, 2000.

52. Fan, J. L., Zhen, W. Z., Xie, W. X., "Supressed fuzzy c-means clustering algorithms," *Pattern Recognition Letters*, vol. 24, pp. 1607-1612, 2003.

53. Marazzi, A., "Algorithms and programs for robust linear regression," in *Directions in Robust Statistics and Diagnostics: Part I*, Stahel, W., Weisberg, S., Eds. New York: Springer-Verlag, New York, 1991, pp. 183-199.

54. Atkinson, A. C., Weisberg, S., "Simulated annealing for the detection of multiple outliers," in *Directions in Robust Statistics and Diagnostics: Part I*, Stahel, W., Weisberg, S., Eds. New York: Springer-Verlag, 1991, pp. 7-20.

55. Hawkins, D. M., "The feasible solution algorithm for least trimmed squares regression," *Computational Statistics and Data Analysis*, vol. 17, pp. 185-196, 1994.

56. Xu, L., Oja, E., Kultanen, P., "A new curve detection method: Randomized Hough transform," *Pattern Recognition Letters*, vol. 11, pp. 331-338, 1990.

57. Xie, X. L., Beni, G., "A validity measure for fuzzy clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13(8), pp. 841-847, 1991.

58. Hartigan, J. A., *Clustering Algorithms*. New York: Wiley, 1975.

59. Berry, M. J. A., Linoff, G., *Data Mining Techniques for Marketing, Sales and Customer Support*. New York: Wiley, 1996.

60. Hubert, L. J., Schultz, J., "Quadratic assignment as a general data-analysis strategy," *British Journal of Mathematical and Statistical Psychology*, vol. 29, pp. 191-241, 1976.

61. Goodman, L. A., Kruskal, W. H., "Measures for association for cross-classifications," *Journal of the American Statistical Association*, vol. 49, pp. 732-764, 1954.

62. Bezdek, J. C., "Numerical taxonomy with fuzzy sets," *Journal of Mathematical Biology*, vol. 1, pp. 57-71, 1974.

63. Bezdek, J. C., "Mathematical models for systematics and taxonomy," in *Proceedings of the 8th International Conference on Numerical Taxonomy*, Estabrook, G., Ed. San Francisco, CA: Freeman, 1975, pp. 143-166.

64. Windham, M. P., "Cluster validity for fuzzy clustering algorithms," *Fuzzy Sets and Systems*, vol. 5(2), pp. 177-185, 1981.

65. Windham, M. P., "Cluster validity for fuzzy c-means clustering algorithm," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 4(4), pp. 357-363, 1982.

66. Libert, G., Roubens, M., "New experimental results in cluster validity of fuzzy clustering algorithms," in *New Trends in Data Analysis and Applications*, Eds Janssen, J., Macrotorchino, J. -F., Proth, J. -M., Eds. Amsterdam, The Netherlands: North Holland, 1983, pp. 205-218.

67. Windham, M. P., Bock, H., Walker, H. F., "Clustering information from convergence rate," in Proceedings of the *2nd Conference of the International Federation Classification Society*, Washington D. C., 1989, p. 143.

68. Gunderson, R., "Application of fuzzy ISODATA algorithms to star tracker pointing systems," in *Proceedings of the 7th Triennial World IFAC Congress*, Helsinki, Finland, 1978, pp. 1319-1323.

69. Shannon, C. E., "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379-423, 1948.

70. Dunn, J. C., "A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters," *Journal of Cybernetics*, vol. 3, pp. 32–57, 1973.

71. Dunn, J. C., "Indices of partition fuzziness and detection of clusters in large data sets," in *Fuzzy Automata and Decision Processes*, New York: Elsevier, 1977.

72. Fukuyama, Y., Sugeno, M., "A new method of choosing the number of clusters for the fuzzy c-means method," In *Proceedings of the 5th Fuzzy Systems Symposium*, 1989, pp. 247-250.

73. Gath, I., Geva, A. B., "Unsupervised optimal fuzzy clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 773-781, 1989.

74. Davé, R. N., "Validating fuzzy partitions obtained through c-shells clustering, *Pattern Recognition Letters*, vol. 17, pp. 613-623, 1996.

75. Bezdek, J. C., Hathaway, R. J., "VAT: A tool for visual assessment of (cluster) tendency," *In Proceedings of the 2nd International Joint Conference on Neural Networks (IJCNN'02)*, 2002, pp. 2225-2230.

76. Bezdek, J. C., Hathaway, R. J., "Visual cluster validity displays for prototype generator clustering methods," In *Proceedings of the 12th IEEE International Conference on Fuzzy Systems (FUZZ-IEEE'03)*, 2003, pp. 875-880.

77. Naus, J. I., "Approximations for distributions of scan statistics," *Journal of the American Statistical Association*, vol. 17, pp. 177-183, 1982.

78. Mead, R., "A test for spatial pattern at several scales using data from a grid of contiguous quadrats," *Biometrics*, vol. 30, pp. 295-308, 1974.

79. Ripley, B. D., "Modelling spatial patterns." *Journal of Royal Statistical Society*, vol. B39, pp. 172-212, 1977.

80. Hopkins, B., "A new method of determining the type of distribution of plant individuals," *Annals of Botany*, vol. 18, pp. 213-226, 1954.

81. Holgate, P., "Some new tests of randomness," *Journal of Ecology*, vol. 53, pp. 261-266, 1965.

82. Holgate, P., "Tests of randomness based on distance measures," *Biometrika*, vol. 52, p. 345, 1965.

83. Besag, J. E., Gleaves, J. T., "On the detection of spatial pattern in plant communities," *Bulletin of the International Statistical Institute*, vol. 45, pp. 153-158, 1973.

84. Eberhardt, L. L., "Some developments in distance sampling," *Biometrics*, vol. 23, pp. 207-216, 1967.

85. Cox, T. F., Lewis, T., "A conditioned distance ratio method for analyzing spatial patterns," *Biometrika*, vol. 63, pp. 483-491, 1976.

86. Panayirci, E., Dubes, R. C., "A test for multidimensional clustering tendency," *Pattern Recognition*, vol. 6(4), pp. 433-444, 1983.

87. Jain, A. K., Dubes, R., Feature definition in pattern recognition with small sample size, *Pattern Recognition*, vol. 10, pp. 85-97, 1978.

88. Bellman, R., *Adaptive Control Processes*. Princeton, New Jersey: Princeton Univ. Press, 1961.

89. Joliffe, I.T. (1986): *Principal Component Analysis*. New York: Springer-Verlag, 1986.

90. Fukunaga, K. (1972): *Introduction to Statistical Pattern Recognition*. New York: Academic Press, 1972.

91. Schiffman, S. S., Reynolds, M. L., Young, F. W. *Introduction to Multidimensional Scaling*. New York: Academic Press, 1981.

92. Blum, A. L., Langley, P., "Selection of relevant features and examples in machine learning," *Artificial Intelligence*, vol. 97, pp. 245-271, 1997.

93. Quinlan, J. R., "Induction of decision trees," *Machine Learning*, vol. 1, pp. 81-106, 1986.

94. Baim, P. W., "A method for attribute selection in inductive learning systems," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10(6), pp. 888-896, 1988.

95. Rauber, T. W., Steiger-Gargao, A. S., "Feature selection of categorical attributes based on contingency table analysis," in *Proceedings of the 5th Portuguese Conference on Pattern Recognition*, Porto, Portugal, 1993.

96. Bril, F. Z., Brown, D. E., Worthy, N. W., "Fast genetic selection of features for neural network classifiers," *IEEE Transactions on Neural Networks*, vol. 3, pp. 324-328, 1992.

97. Raymer, M. L., Punch, W. F., Goodman, E. D., Kuhn, L. A., Jain, A. K., "Dimensionality reduction using genetic algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 4(2), pp. 164-171, 2000.

98. Shenkin, P. S., McDonald, D. Q., "Cluster analysis of molecular conformations," *Journal of Computational Chemistry*, vol. 15, pp. 899-916, 1994.

99. Murray-Rust, P., Raftery, J., "Computer analysis of molecular geometry, Part IV: Classification of differences in conformation," *Journal of Molecular Graphics*, vol. 3, pp. 50-59, 1985.

100. Chema, D., Goldblum, A., "The nearest neighbor method - Finding families of conformations within a sample," *Journal of Chemical Information and Computer Sciences*, vol. 43, pp. 208-217, 2003.

101. Feher, M., Schmidt, J. M., "Metric and multidimensional scaling: Efficient tools for clustering molecular conformations," *Journal of Chemical Information and Computer Sciences*, vol. 41, pp. 346-353, 2001.

102. Feher, M., Schmidt, J. M., "Fuzzy clustering as a means of selecting representative conformers and molecular alignments," *Journal of Chemical Information and Computer Sciences*, vol. 43, pp. 810-818, 2003.

103. Prisinzano, T., Rice, K. C., Baumann, M. H., Rothman, R. B., "Development of neurochemical normalization ("Agonist substitution") therapeutics for stimulant abuse: Focus on the dopamine uptake inhibitor, GBR12909," *Current Medicinal Chemistry - Central Nervous System Agents*, vol. 4(1), pp. 47-59, 2004.

104. Pai, D. S., "Analysis of molecular conformations using relative planes," *M.S. Thesis*, New Jersey Institute of Technology, Newark, New Jersey, 2004.

105. Wishart, D., "K-means clustering with outlier deletion for data mining with mixed variables and missing values," in *Exploratory Data Analysis in Empirical Research*, Schwaiger, M., Optiz, O., Eds. Berlin: Springer, 2002, pp. 841-847.

106. Davé, R. N., Sen, S., "Robust fuzzy clustering of relational data," *IEEE Transactions on Fuzzy Systems*, vol. 10, pp. 713-727, 2002.

107. Misra, M., Banerjee, A., Davé, R. N., Venanzi, C. A., "Novel feature extraction technique for fuzzy relational clustering of a flexible dopamine reuptake inhibitor," *Journal of Chemical Information and Modeling*, vol. 45(3), pp. 610-623, 2005.

108. Pandit, D., Venanzi, C. A., (Unpublished results), 2005.

109. Sinha, S. K., Knight, M. A., "Intelligent system for condition monitoring of underground pipelines," *Computer-Aided Civil Infrastructure*, vol. 19, pp. 42-53, 2004.

110. Sinha, S. K., Fieguth, P. W. Polak, M. A., "Computer vision techniques for automatic structural assessment of underground pipes, *Computer-Aided Civil Infrastructure*, vol.18, pp. 95-112, 2003.

111. Sinha, S. K., Karray, F., "Classification of underground pipe scanned images using feature extraction and neuro-fuzzy algorithm," *IEEE Transactions on Neural Networks*, vol. 13, pp. 393-401, 2002.

112. Sinha, S. K., Karray, F., Fieguth P. W., "Underground pipe cracks classification using image analysis and neuro-fuzzy algorithm," in *Proceedings of the IEEE International Symposium on Intelligent Control/ Intelligent Systems and Semiotics*, Cambridge, MA, 1999, pp. 399-404.

113. Fieguth, P. W. and Sinha, S. K., "Automated analysis and detection of cracks in underground pipelines," In *Proceedings on the IEEE International Conference on Image Processing*, Kobe, Japan, 1999, pp. 395-399.

114. Moselhi, O., Shehab-Eldeen, T., "Automated detection of surface defects in water and sewer pipes," *Automation in Construction*, vol. 8, pp. 581-588, 1999.

115. Broadhurst, S. J., Cockerham, G., Taylor, N., Pridmore, T., "Automatic task modeling for sewer studies," *Automation in Construction*, vol. 5, 61-71, 1996.

116. Bezdek, J. C., Keller, J. M., Krishnapuram, R., Pal, N. R., *Fuzzy Models and Algorithms for Pattern Recognition and Image Processing*, Boston, MA: Kluwer, 1999.

117. Haralick, R. M., Shapiro, L. G., *Computer and Robot Vision*, Vol. I, Reading, MA: Addison Wesley, 1992.

118. Pal, N. R., Pal, S. K., "A review of image segmentation techniques," *Pattern Recognition*, vol. 26(9), pp. 1277-1294, 1993.

119. Bezdek, J. C., Sutton, M. A., "Image processing in medicine," in *Applications of Fuzzy Systems*, Zimmerman, H. J., Ed. Norwell, MA: Kluwer, 1999.

120. Prewitt, J. M., "Object enhancement and extraction in picture processing and psychopictorics," Lipkin, B. S., Rosenfeld, A., Eds. New York: Academic Press, 1970, pp. 75-149.

121. Boujemaa, N., Stamon, G., Lemoine, J., "Fuzzy iterative image segmentation with recursive merging," in *Proceedings of the SPIE Conference on Visual Communication and Image Processing*, 1992, pp. 1271-1281.

122. Krishnapuram, R., Lee, J., "Fuzzy-set based hierarchical networks for information fusion in computer vision," *Neural Networks*, vol. 5, pp. 335-350, 1992.

123. Keller, J. M., Chen, Z., "Learning in fuzzy neural networks utilizing adaptive hybrid operators," in *Proceedings of the International Conference on Fuzzy Logic and Neural Networks*, Iizuka, Japan, 1992, pp. 85-87.

124. Campbell, G., Rogers, K., Gilbert, J., "PSET – System for quantitative sewer assessment," In *Proceedings of the International No-Dig Conference*, Hamburg Germany, 1995, pp. 455-462.

125. Ridler, T. W., Calvard, S., "Picture thresholding using an iterative selection method," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 8, pp. 630-632, 1978.

126. Meegoda, J. N., Juliano, T. M., Ayoola, G. M., Dhar, S., "Inspection, cleaning, condition assessment and prediction of remaining service life of CSCPs," Paper No. 04-4426, *83rd Annual Meeting of the Transportation Research Board*, Washington D. C., January 2004.

127. Meegoda, J. N., Juliano, T. M., Abdel-Malek, L., Ratnaweera, P., "A framework for inspection, maintenance and replacement of corrugated steel culvert pipes," Paper No. 05-1219, *84th Annual Meeting of the Transportation Research Board*, Washington D. C., January 2005.

128. Gomes, R. N., Lee, L. L., "Feature extraction based on fuzzy set theory for handwriting recognition," in *Proceedings of the 6th International Conference on Document Analysis and Recognition*, 2001, pp. 655-659.

129. Philip, K. P., Dove, E. L., Stanford, W., Chandran, K. B., McPherson, D. D., Gotteiner, N. L., "The fuzzy Hough Transform-feature extraction in medical images," *IEEE Transactions on Medical Imaging*, vol. 13(2), pp. 235-240, 1994.