DEVELOPMENT OF A MEDICAL INFERENCING SYSTEM USING DATA CLUSTERING

A Thesis submitted in partial fulfillment of the requirements *for the award of degree of*

Master of Engineering

in

Electronic Instrumentation and Control

Submitted by

Yuvraj Bhushan Khare

Roll No. 800951024

Under the Guidance of

Dr. Yaduvir Singh

Associate Professor

Department of Electrical and Instrumentation Engineering

Thapar University

(Established under the section 3 of UGC act, 1956)

Patiala, 147004, Punjab, India

July 2011

II

DECLARATION

I hereby certify that the work which is being presented in the thesis entitled **"Development of a Medical Inferencing System Using Data Clustering"** in partial fulfillment of award of degree of **Master of Engineering** in **Electronics Instrumentation and Control** submitted in Electrical and Instrumentation Engineering department, Thapar University, Patiala is an authentic record of my own work carried under the supervision of **Dr. Yaduvir Singh**, Associate Professor, Department of Electrical and Instrumentation Engineering, Thapar University, Patiala, Punjab.

Date: 28/06/11

Name: Yuvraj Bhushan Khare
Roll No: 800951024

I certify that the above statement made by the student is correct to the best of my knowledge and belief.

Date: 28/06/2011

28/06/2011

Dr. Yaduvir Singh
Associate Professor
Department of Electrical and Instrumentation Engineering
Thapar University, Patiala
Punjab

Countersigned By

S. Ghosh 30/6/11

Dr. Smarajit Ghosh
Head of Department
Department of Electrical and Instrumentation Engineering
Thapar University, Patiala
Punjab

Dr. S K Mohapatra
Dean of Academic Affairs
Thapar University, Patiala
Punjab

ABSTRACT

Now a day with the advance technology data collection is far easier than before. Data of relevant subjects are collected and stored for future analysis. The analysis can be online or offline. Data can be collected from different sources like industrial plant, wireless sensors, stock market, banking and financial institutions, and medical parameters of the patient. The data are collected with the help of different sensors or transducers and stored in digital format in different storage media. Data mining or knowledge discovery from raw data is very much important and a lot of research has been going on in this field. Knowledge discovery gives a better edge to a person to take better action.

Medical diagnosis is a very important area of research where with the help of engineering techniques diagnosis is made. A new approach of medical diagnosis uses the day to day monitoring of the patient's medical data to determine the type of disease, degree of seriousness of the disease. First of all the day to day medical monitoring data is taken and clustered, so that it can be arranged in a better way. Different data mining or soft computing techniques are implemented in the clustered data to discover hidden trends, knowledge in the data which helps to gain an insight view of the subject.

This thesis tries to develop a medical diagnosis system using the day to day patient monitoring data. First of all the data are acquired and clustered or grouped. Fuzzy based inferencing techniques are used to discover knowledge from the clustered data.

ACKNOWLEDGEMENT

I would like to express my gratitude to **Dr. Yaduvir Singh**, Associate Professor**,** Electrical and Instrumentation Engineering Department, Thapar University, Patiala for his patient guidance and support throughout this thesis work. I am truly very fortunate to have the opportunity to work with him. He has provided me help in technical writing and presentation style, and I found this guidance to be extremely valuable. I am very thankful to head of the Department, **Dr. Smarajit Ghosh,** for his encouragement, support and providing the facilities for the completion of this thesis. I am also thankful to entire faculty and staff members of Electrical and Instrumentation Engineering Department for their unyielding encouragement. I am greatly indebted to all my friends, who have graciously applied themselves to the task of helping me with ample morale support and valuable suggestions. Finally, I would like to extend my gratitude to all those persons who directly or indirectly helped me in the process and contributed towards this work.

Yuvraj Bhushan Khare

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
INTRODUCTION

1.1 Overview

Disease diagnosis is a major problem area for researchers for a long time. To accurately diagnosis a disease is of prime concern for a doctor. To help the medical personnel with the diagnosis tool, many engineering techniques have evolved in the past. There are various conventional methods of disease diagnosis, but application of soft computing technique with information technology has given a new dimension to this area. With the advancement in storage devices now the past records of the patient is easily available. To predict the future behavior of the patient by studying the past history is a very good technique to diagnosis a disease.

1.2 Motivation

In modern days continuous monitoring of different medical parameters like blood pressure, temperature, heart rate, respiratory rates, ECG, EEG, and EMG can be done easily and all these data can be stored for future analysis and diagnosis. But the huge amount of data needs a lot of storage space which is costly. And another aspect is that all data are not of importance, only a few relevant data are important which posses a big problem in disease diagnosis.

To make the disease diagnosis system more effective, the data should be filtered, conditioned and clustered. For clustering the data different contemporary and soft computing based data clustering techniques are available. After the data of different parameters are clustered, an inference mechanism is required to correlate two different parameters to a single cause. Fuzzy logic is a soft computing technique which is more like the human mind. So fuzzy logic based inference mechanism can be implemented for correlation.

1.3 Objectives of the Thesis

This thesis takes periodic medical data of patients taken at a fixed interval, filters, conditions and clusters the data. After clustering a fuzzy based inference mechanism is used to correlate different parameters to a common symptom. This result creates a decision support system for disease diagnosis system. The diabetes data is taken from UCI respiratory machine learning databases, University of California, Irvine, Department

of Information and computer Science available http://archive.ics.uci.edu/ml/datasets/Diabetes (Last accessed: 20/5/2011), contributed by Dr. Michael Kahn, Washington University, St. Louis, MO. The blood pressure data is taken from UCI respiratory machine learning databases, University of California, Irvine, Department of Information and computer Science available http://archive.ics.uci.edu/ml/datasets/BloodPressure (Last accessed: 20/5/2011), contributed by Dr. Michael Kahn, Washington University, St. Louis, MO.

1.4 Organization of the Thesis

Chapter 1 introduces the concept of disease diagnosis, gives the overview of the problem and chalks out the objectives of the thesis.

Chapter 2 gives a complete overview of data clustering techniques like square error clustering, K means clustering, hierarchical clustering, partitional clustering.

Chapter 3 gives a relevant literature review of previous work done on this topic.

Chapter 4 gives a complete overview of fuzzy logic system with different fuzzy inference mechanism, de fuzzification techniques.

Chapter 5 gives a complete over view of soft computing based data clustering techniques. It gives a detailed study of fuzzy C means algorithm.

Chapter 6 introduces a novel approach of disease diagnosis using data clustering and fuzzy inference mechanism.

Chapter 7 is dedicated for simulation, results and discussions.

Chapter 8 concludes the thesis with future scope of research.

CHAPTER 2

LITERATURE REVIEW

Scott C Newton et.al in their paper proposed a hybrid adaptive fuzzy leader clustering technique implemented in ART-I like structure to cluster speech, image and medical data [1].

Y M Sebzalli et.al, has proposed two techniques like principal component analysis (PCA) and fuzzy C means clustering to identify and develop operational strategy for manufacture of desired product in process industry. This research paper takes a case study of fluid catalytic cracking process used in refinery industry. The authors analyzed the problem by collecting three hundred data from the process site and applying principal component analysis and fuzzy c means clustering algorithm in the datasets [2].

Timo Ahvenlampi et.al, studied the controllability of kappa number in two cooking application. Kappa number is the quality measure of pulp cooking method. The authors investigated the clustering and fault diagnosis approach of cooking system [3].

Young-Hak Lee et.al has proposed an adaptive monitoring technique of real time industrial process to classify and distinguish operational changes. The proposed method extracts process knowledge and classifies process state changes. The case study taken by the authors is a refinery fired heater [36].

Skrjanc I has presented in his research paper a method of sensor fault detection in waste water treatment plant using Gustafson-Kessel fuzzy clustering algorithm. Different measurements like influent ammonia concentration, dissolved oxygen concentration in aerobic reactors are measured and analyzed [5].

C Lionberger et.al has proposed a novel method of online acquisition and clustering of GRETINA (Gamma Ray Energy Tracking In-beam Nuclear Array) which is an array of 28 36-segment germanium crystals [15].

Zhe Song et.al has proposed a data mining approach to develop a model for optimizing the efficiency of an electric utility boiler. The industrial boiler generates real time data used for clustering. The clustering algorithm learns and generates new knowledge used to update the control signature database. Based on the real-time boiler status, the optimization algorithm searches the control signature database for an optimal centroid controlling the process. Thus, the boiler performance is improved [16].

N Sujatha et.al has proposed in her research paper an innovative way to find out the web usage pattern by implementing modified K means algorithms and optimizing the cluster quality by using genetic algorithm based refinement algorithm. The modified K means algorithm and refinement algorithm based on genetic algorithm is applied in web access log collected from internet traffic archive (ITA) [33].

Osama Abu Abbas in his research paper compared different conventional and intelligent clustering algorithms according to the size of data sets, number of clusters, and type of datasets to find out the performance of the clustering algorithm, quality of clustering and accuracy of the clustering [23].

K Premalatha et.al in her research paper applied swarm intelligence technique like particle swarm intelligence (PSO) in cluster analysis. With application of PSO the optimal shape of cluster can be found out [24].

V Kavitha et.al, has given a literature review of clustering of time series data stream. A time series data is being generated at a unique speed from almost every application domain. These types of dataset are special type of dataset which has temporal ordering [26].

Hesam Izakian and Ajith Abraham proposed a hybrid fuzzy C means algorithm which implements fuzzy particle swarm optimization algorithm in fuzzy C means algorithm. This proposed hybrid algorithm make use of merit of both the algorithms and finds out optimal cluster structures [27].

S Kalyani and K S Swarup proposed a supervised fuzzy C means algorithm for security assessment and classification of power system. The proposed algorithm is tested on 39 bus New England and IEEE 57 bus systems. The classification results of supervised fuzzy C means algorithm is tested with method of least squared and multi layered perceptron classifiers [28].

Xian-Xia Zhang et.al, proposed a novel sensor placement technique by utilizing main feature of spatial distribution [29].

Mika Liukkonen et.al in their research paper described the dependencies between process variables and the concentrations of gaseous emission components. They also created multivariate nonlinear models describing their formation in the process. A

process model was created using self organizing map and was clustered using K means algorithm for determination of subsets [30].

Vasil Simeonov et.al, has proposed a novel method of water quality assessment of high mountain lakes in Pirin Mountain in Bulgaria by application of cluster analysis and principal component analysis. The authors have also studied the classification of dataset by using self organizing map [31].

Ibrahim Masood and Adnan Hassan proposed an ANN based control chart pattern recognition system for process plant monitoring and control. The feature based and wavelet denoise method is used for input representation [32].

CHAPTER 3

DATA CLUSTERING TECHNIQUES AND ALGORITHMS

3.1 Introduction

Clustering is a division of data into groups of similar objects. Representing the data by fewer clusters necessarily loses certain fine details, but achieves simplification. It models data by its clusters. Data modeling puts clustering in a historical perspective rooted in mathematics, statistics, and numerical analysis. From a machine learning perspective clusters correspond to hidden patterns, the search for clusters is unsupervised learning, and the resulting system represents a data concept. From a practical perspective clustering plays an outstanding role in data mining applications such as scientific data exploration, information retrieval and text mining, spatial database applications, Web analysis, CRM, marketing, medical diagnostics, computational biology, control and many other applications. It is important to understand the difference between clustering (unsupervised classification) and discriminant analysis (supervised classification). In supervised classification, we are provided with a collection of labeled (preclassified) patterns; the problem is to label a newly encountered, yet unlabeled, pattern. Typically, the given labeled (training) patterns are used to learn the descriptions of classes which in turn are used to label a new pattern. In the case of clustering, the problem is to group a given collection of unlabeled patterns into meaningful clusters. In a sense, labels are associated with clusters also, but these category labels are data driven; that is, they are obtained solely from the data.

Data mining adds to clustering the complications of very large datasets with very many attributes of different types. This imposes unique computational requirements on relevant clustering algorithms. A variety of algorithms have recently emerged that meet these requirements and were successfully applied to real-life data mining problems. In recent years, the dramatic rise in the use of the web and the improvement in process industries in general have transformed our society into one that strongly depends on information. The huge amount of data that is generated by this process contains important information that accumulates daily in databases and is not easy to extract. This data is scattered in between the upper and the lower limits, applying required control or strategy to this data requires lots of computational work, for having a better control strategy. The

clustered data gives us a better control efficiency and performance of our system rather than working with an unorganized scattered dataset. The field of data mining developed as a means of extracting information and knowledge from databases to discover patterns or concepts that are not evident The process usually consists of the following: transforming the data to a suitable format, cleaning it, and inferring or making conclusions regarding the data. Data analysis underlies many computing applications, either in a design phase or as part of their on-line operations. Data analysis procedures can be dichotomized as either exploratory or confirmatory, based on the availability of appropriate models for the data source, but a key element in both types of procedures (whether for hypothesis formation or decision-making) is the grouping, or classification of measurements based on either of the following:

(i) Goodness-of-fit to a postulated model,

(ii) Natural groupings (clustering) revealed through analysis.

Cluster analysis is the organization of a collection of patterns (usually represented as a vector of measurements, or a point in a multidimensional space) into clusters based on similarity. Cluster analysis is thus a tool of discovery. It may reveal associations and structure in data which, though not previously evident, nevertheless are sensible and useful once found. The results of cluster analysis may contribute to the definition of a formal classification scheme, such as a taxonomy for related process variables, parameters or objects; or suggest statistical models with which to describe control; or indicate rules for assigning new cases to classes for control and analysis purposes; or provide measures of definition, variations and change in what previously were only broad concepts. Clustering is the classification of similar objects into different groups, or more precisely, the partitioning of a data set into subsets (clusters), so that the data in each subset (ideally) share some common trait - often proximity according to some defined distance measure

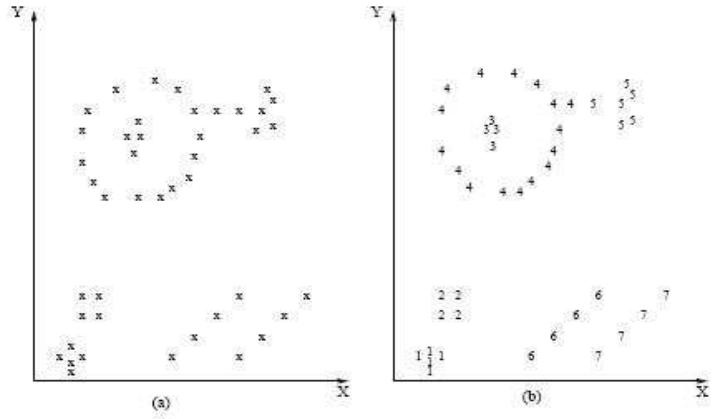

Figure 3.1: Depiction of Data Clustering

Clustering is useful in several exploratory pattern-analysis, grouping, decision-making, and machine-learning situations, including data mining, document retrieval, image segmentation, and pattern classification. However, in many such problems, there is little prior information (e.g., statistical models) available about the data, and the decision-maker must make as few assumptions about the data as possible. It is under these restrictions that clustering methodology is particularly appropriate for the exploration of interrelationships among the data points to make an assessment (perhaps preliminary) of their structure.

3.2 Issues of Clustering

The main requirements that a clustering algorithm should satisfy are the following

- Dealing with different types of attributes:

  There are various attributes of data that any clustering algorithm need to satisfy, the most general taxonomy being in common use distinguishes among numeric (continuous), ordinal, and nominal variables. A numeric variable can assume any value in R. An ordinal variable assumes a small number of discrete states, and these states can be compared.
- Scalability to large datasets:

  The data sets could be in any possible range, varying between large extremes and they need to be normalized by the clustering algorithm.
- Ability to work with high dimensional data:

The data could be multidimensional varying from 1, 2……n.; depending on the application data on which clustering is being applied.

- Ability to find clusters of irregular or arbitrary shape:

  The shape of clusters could be any arbitrary shapes. We prefer using Euclidean distance to get a circular shape of the clusters, but still shape of clusters can not be accurately defined

- Handling outliers:

  The data points on the boundary of clusters need to be handled; this is done in hierarchical methods by associating the boundary points to one of the clusters. While in fuzzy clustering, we associate membership functions to the points lying on the boundary of clusters.

- Time complexity :

  Complexity of the data points in terms of time has to be taken care of while clustering.

- Data order dependency:

  Dependency of data points on other variable can affect the clustering of data and there by the cluster centers too, so it has to be taken care before hand.

- Labeling or Assignment:

Other factors are also play a vital role in clustering of data, some of them are stated below:

- Reliance on a priori knowledge and user defined parameters
- Interpretability of results
- Minimal requirements for domain knowledge to determine input parameters;
- Ability to deal with noise
- Interpretability
- Usability

It can be shown that there is no absolute “best” criterion which would be independent of the final aim of the clustering. Consequently, it is the user which must supply this criterion, in such a way that the result of the clustering will suit their needs. For instance, we could be interested in finding representatives for homogeneous groups (data reduction), in finding “natural clusters” and describe their unknown properties (“natural”

data types), in finding useful and suitable groupings (“useful” data classes) or in finding unusual data objects (outlier detection).

3.3 Distance: A Measure of Similarity

Since similarity is fundamental to the definition of a cluster, a measure of the similarity between two patterns drawn from the same feature space is essential to most clustering procedures. Because of the variety of feature types and scales, the distance measure (or measures) must be chosen carefully. It is most common to calculate the dissimilarity between two patterns using a distance measure defined on the feature space. We will focus on the well-known distance measures used for patterns whose features are all continuous. The most popular metric for continuous features is the Euclidean distance An important component of a clustering algorithm is the distance measure between data points. If the components of the data instance vectors are all in the same physical units then it is possible that the simple Euclidean distance metric is sufficient to successfully group similar data instances. However, even in this case the Euclidean distance can sometimes be misleading. Figure shown below illustrates this with an example of the width and height measurements of an object. Despite both measurements being taken in the same physical units, an informed decision has to be made as to the relative scaling. As the figure shows, different scaling can lead to different clustering’s.

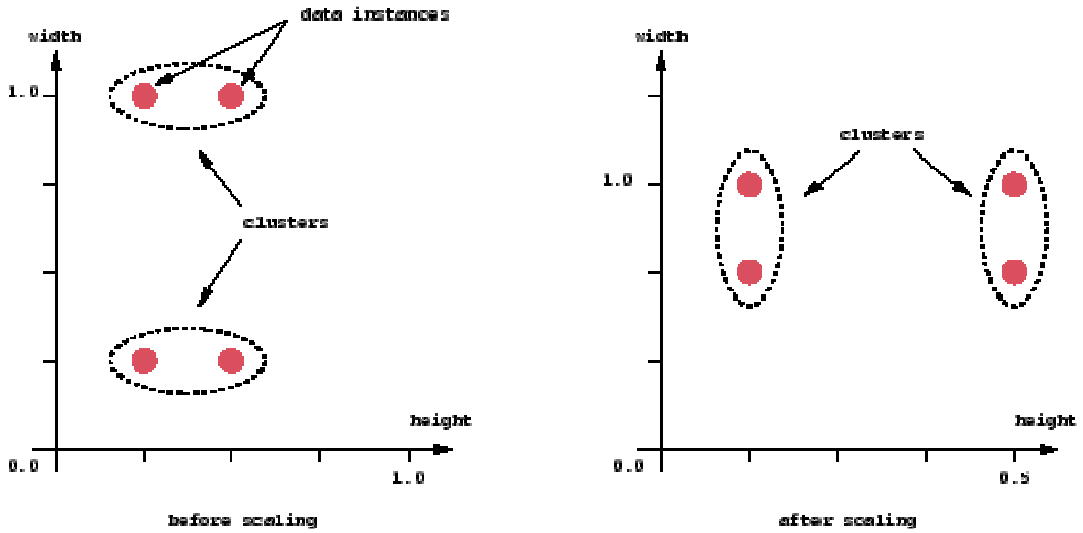

Figure 3.2: Distance criterion

Notice however that this is not only a graphic issue: the problem arises from the mathematical formula used to combine the distances between the single components of the data feature vectors into a unique distance measure that can be used for clustering purposes: different formulas leads to different clustering's. Again, domain knowledge need be used to guide the formulation of a suitable distance measure for each particular application.

3.3.1 Minkowski Metric

For higher dimensional data, a popular measure is the Minkowski metric,

$$d_p\left(x_i, x_j\right) = \left(\sum_{K=1}^{d} \left|x_{i,k} - x_{j,k}\right|^p\right)^{\frac{1}{p}} \quad (1)$$

Where *d* is the dimensionality of the data. The Euclidean distance is a special case where *p*=2, while Manhattan metric has *p*=1. However, there are no general theoretical guidelines for selecting a measure for any given application. Distance is most common to calculate the dissimilarity between two patterns using a distance measure defined on the feature space. The most popular metric for continuous features is the Euclidean distance

$$d_2(x_i, x_j) = (\sum_{k=1}^{d} (x_{i,k} - x_{j,k})^2)^{1/2} = \left\| x_i - x_j \right\|_2 \quad (2)$$

This is a special case of the Minkowski metric

$$d_p(x_i, x_j) = (\sum_{k=1}^{d} (x_{i,k} - x_{j,k})^p)^{1/p} = \left\| x_i - x_j \right\|_p \quad (3)$$

3.4 Distance Functions

The concept of dissimilarity (or distance) or dual similarity is the essential component of any form of clustering that helps us navigate through the data space and form clusters. By computing dissimilarity, we can sense and articulate how close together two patterns are and, based on this closeness, allocate them to the same cluster. Formally, the dissimilarity $d(\mathbf{x}, \mathbf{y})$ between **x** and **y** is considered to be a two-argument function satisfying the following conditions:

$d(x,y) \geq 0$ for every **x** and **y** (4)

$d(x,x) = 0$ for every **x** (5)

$d(x,y) = d(y,x)$ (6)

This list of requirements is intuitively appealing. We require a nonnegative character of the dissimilarity. The symmetry is also an obvious requirement. The dissimilarity attains a global minimum when dealing with two identical patterns, that is $d(x,x) = 0$.

Distance, (metric) is a more restrictive concept, as we require the triangular inequality to be satisfied; that is, for any pattern **x, y**, and **z** we have

$d(x,y) + d(y,z) \geq d(x,z)$ (7)

Table 1: Selected Distance Function between Patterns x and y.

Distance Function	Formula and Comments
Euclidean distance	$d(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$
Hamming (city block) distance	$d(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{n}\lvert x_i - y_i\rvert$
Tchebyschev distance	$d(\mathbf{x},\mathbf{y}) = \max_{i=1,2,\ldots,n}\lvert x_i - y_i\rvert$
Minkowski distance	$d(\mathbf{x},\mathbf{y}) = \sqrt[p]{\sum_{i=1}^{n}(x_i - y_i)^p},\ p > 0$
Canberra distance	$d(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{n}\frac{\lvert x_i - y_i\rvert}{x_i + y_i}$, x_i and y_i are positive
Angular separation	$d(\mathbf{x},\mathbf{y}) = \frac{\sum_{i=1}^{n} x_i y_i}{\left[\sum_{i=1}^{n} x_i^2 \sum_{i=1}^{n} y_i^2\right]^{1/2}}$ Note: this is a similarity measure that expresses the angle between the unit vectors in the direction of **x** and **y**

In the case of continuous features (variables), we have a long list of distance functions each of these functions implies a different view of the data because of their geometry. The geometry is easily illustrated when we consider two features ($\mathbf{x} = [x_1\ x_2]^T$) and compute the distance of **x** from the origin. The contours of the constant distance show what type of geometric construct becomes a focus of the search for structure. Here we become aware that the Euclidean distance favors circular shapes of data clusters. With the distance functions come some taxonomy; the Minkowski distance comprises an infinite family of distances, including well-known and commonly used ones such as the Hamming, Tchebyschev, and Euclidean distances. The same effect shown in Figure can be achieved when the value of the power in the Minkowski distance is changed; one commonly used generalization is the Mahalanobis distance

$$d(x,y) = \mathbf{x}^T A^{-1} \mathbf{y} \qquad (8)$$

Here A is a positive definite matrix. By choosing this matrix, we can control the geometry of potential clusters by rotating the ellipsoid (off diagonal entries of A and changing the length of its axes (the elements lying on the main diagonal of the matrix).

3.5 Components of a Clustering Task

Typical pattern clustering activity involves the following steps

(1) Pattern representation (optionally including feature extraction and/or selection),

(2) Definition of a pattern proximity measure appropriate to the data domain,

(3) Clustering or grouping,

(4) Data abstraction (if needed), and

(5) Assessment of output (if needed).

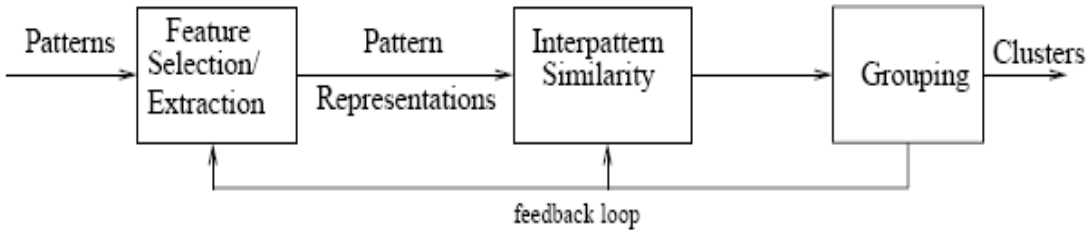

Figure 3.3: Stages of Clustering

Figure 3.3 above depicts a typical sequencing of the first three of these steps, including a feedback path where the grouping process output could affect subsequent feature extraction and similarity computations.

Pattern representation refers to the number of classes, the number of available patterns, data, and the number, type, and scale of the features available to the clustering algorithm. Some of this information may not be controllable by the practitioner.

Feature selection is the process of identifying the most effective subset of the original features to use in clustering.

Feature extraction is the use of one or more transformations of the input features to produce new salient features. Either or both of these techniques can be used to obtain an appropriate set of features to use in clustering.

Pattern proximity is usually measured by a distance function defined on pairs of patterns. A variety of distance measures are in use in the various communities. A simple distance measure like Euclidean distance can often be used to reflect dissimilarity between two patterns, whereas other similarity measures can be used to characterize the conceptual similarity between. The grouping step can be performed in a number of ways. The output clustering (or clustering) can be hard (a partition of the data into groups) or fuzzy (where each pattern has a variable degree of membership in each of the output clusters). Hierarchical clustering algorithms produce a nested series of partitions based on a criterion for merging or splitting clusters based on similarity. Partitional clustering algorithms identify the partition that optimizes (usually locally) a clustering criterion. Additional techniques for the grouping operation include probabilistic and graph-theoretic clustering methods.

Data abstraction is the process of extracting a simple and compact representation of a data set. Here, simplicity is either from the perspective of automatic analysis (so that a machine can perform further processing efficiently) or it is human-oriented (so that the representation obtained is easy to comprehend and intuitively appealing). In the clustering context, a typical data abstraction is a compact description of each cluster, usually in terms of cluster prototypes or representative patterns such as the centroid. How is the output of a clustering algorithm evaluated? What characterizes a ‘good’ clustering result and a ‘poor’ one? All clustering algorithms will, when presented with data,

produce clusters - regardless of whether the data contain clusters or not. If the data does contain clusters, some clustering algorithms may obtain 'better' clusters than others. The assessment of a clustering procedure's output, then, has several facets. One is actually an assessment of the data domain rather than the clustering algorithm itself— data which do not contain clusters should not be processed by a clustering algorithm. The study of cluster tendency, wherein the input data are examined to see if there is any merit to a cluster analysis prior to one being performed, is a relatively inactive research area, and will not be considered further in this survey.

Cluster validity analysis, by contrast, is the assessment of a clustering procedure's output. Often this analysis uses a specific criterion of optimality; however, these criteria are usually arrived at subjectively. Hence, little in the way of 'gold standards' exist in clustering except in well-prescribed sub domains. Validity assessments are objective and are performed to determine whether the output is meaningful. A clustering structure is valid if it cannot reasonably have occurred by chance or as an artifact of a clustering algorithm. When statistical approaches to clustering are used, validation is accomplished by carefully applying statistical methods and testing hypotheses. There are three types of validation studies. An external assessment of validity compares the recovered structure to an apriori structure. An internal examination of validity tries to determine if the structure is intrinsically appropriate for the data. A relative test compares two structures and measures their relative merit.

3.6 Clustering Procedure

The clustering procedure includes analysis, validation and visualization of clusters. These steps of clustering are described below

3.6.1 Cluster Analysis

The objective of cluster analysis is the classification of objects according to similarities among them, and organizing of data into groups. Clustering techniques are among the *unsupervised* methods, they do not use prior class identifiers. The main potential of clustering is to detect the underlying structure in data, not only for classification and pattern recognition, but for model reduction and optimization. Different classifications can be related to the algorithmic approach of the clustering techniques.

3.6.2 The data

Clustering techniques can be applied to data that is quantitative (numerical), qualitative (categorical), or a mixture of both. In this thesis, the clustering of quantitative data is considered. The data are typically observations of some physical process. Each observation consists of *n* measured variables, grouped into an *n*-dimensional row vector

$$x_k = [x_{k1}, x_{k2} \ldots\ldots\ldots\ldots\ldots x_{kn}]^T, x_k \in R^n \qquad (10)$$

A set of *N* observations is denoted by $X = \{x_k \,|\, k = 1,2,\ldots\ldots,N\}$ and is represented as an N X n matrix

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \cdots & \vdots \\ x_{N1} & x_{N2} & \cdots & x_{Nn} \end{bmatrix} \qquad (11)$$

In pattern recognition terminology, the rows of X are called patterns or objects, the columns are called the features or attributes, and X is called the pattern matrix. In this thesis, X is often referred to simply as the data matrix. The meaning of the columns and rows of X with respect to reality depends on the context. In medical diagnosis, for instance, the rows of X may represent patients, and the columns are then symptoms, or laboratory measurements for the patients. When clustering is applied to the modeling and identication of dynamic systems, the rows of X contain samples of time signals, and the columns are, for instance, physical variables observed in the system (position, velocity, temperature, etc.). In order to represent the system's dynamics, past values of the variables are typically included in X as well. In system identification, the purpose of clustering is to find relationships between independent system variables, called the regressors, and future values of dependent variables, called the regressands. One should,

however, realize that the relations revealed by clustering are just causal associations among the data vectors, and as such do not yet constitute a prediction model of the given system. To obtain such a model, additional steps are needed.

3.6.3 The clusters

Various definitions of a cluster can be formulated, depending on the objective of clustering. Generally, one may accept the view that a cluster is a group of objects that are more similar to one another than to members of other clusters. The term "similarity" should be understood as mathematical similarity, measured in some well-defined sense. In metric spaces, similarity is often defined by means of a distance norm. Distance can be measured among the data vectors themselves, or as a distance form a data vector to some prototypical object of the cluster. The prototypes are usually not known beforehand, and are sought by the clustering algorithms simultaneously with the partitioning of the data. The prototypes may be vectors of the same dimension as the data objects, but they can also be defined as "higher-level" geometrical objects, such as linear or nonlinear subspaces or functions. Data can reveal clusters of different geometrical shapes, sizes and densities. Algorithms that can detect subspaces of the data space are of particular interest for identification. The performance of most clustering algorithms is influenced not only by the geometrical shapes and densities of the individual clusters but also by the spatial relations and distances among the clusters. Clusters can be well separated, continuously connected to each other, or overlapping each other.

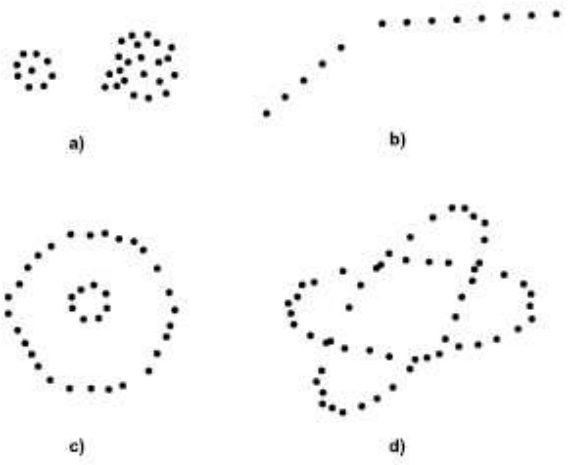

Figure 3.4: Clusters of different shapes and dimensions in R^2.

3.6.4 Cluster partition

Since clusters can formally be seen as subsets of the data set, one possible classification of clustering methods can be according to whether the subsets are fuzzy or crisp (hard). Hard clustering methods are based on classical set theory, and require that an object either does or does not belong to a cluster. Hard clustering in a data set X means partitioning the data into a specified number of mutually exclusive subsets of X. The number of subsets (clusters) is denoted by *c*. Fuzzy clustering methods allow objects to belong to several clusters simultaneously, with different degrees of membership. The data set X is thus partitioned into *c* fuzzy subsets. In many real situations, fuzzy clustering is more natural than hard clustering, as objects on the boundaries between several classes are not forced to fully belong to one of the classes, but rather are assigned membership degrees between 0 and 1 indicating their partial memberships. The discrete nature of hard partitioning also causes analytical and algorithmic intractability of algorithms based on analytic functions, since these functions are not differentiable. The structure of the partition matrix U=$[\mu_{i,k}]$.

3.6.5 Validation Parameters

Cluster validity refers to the problem whether a given fuzzy partition fits to the data at all. The clustering algorithm always tries to find the best fit for a fixed number of clusters and the parameterized cluster shapes. However this does not mean that even the best fit is meaningful at all. Either the number of clusters might be wrong or the cluster shapes might not correspond to the groups in the data, if the data can be grouped in a meaningful way at all. Two main approaches to determining the appropriate number of clusters in data can be distinguished:

- Starting with a sufficiently large number of clusters, and successively reducing this number by merging clusters that are similar (compatible) with respect to some predefined criteria. This approach is called compatible cluster merging.
- Clustering data for different values of *c*, and using *validity measures* to assess the goodness of the obtained partitions.
- This can be done in two ways

- ✓ The first approach is to define a validity function which evaluates a complete partition. An upper bound for the number of clusters must be estimated (c_{max}), and the algorithms have to be run with each $c \in \{2; 3; : : : ; cmax\}$. For each partition, the validity function provides a value such that the results of the analysis can be compared indirectly.

- ✓ The second approach consists of the definition of a validity function that evaluates individual clusters of a cluster partition. Again, *cmax* has to be estimated and the cluster analyses have to be carried out for *cmax*. The resulting clusters are compared to each other on the basis of the validity function. Similar clusters are collected in one cluster; very bad clusters are eliminated, so the number of clusters is reduced. The procedure can be repeated until there are *bad* clusters.

Different scalar validity measures have been proposed in the literature, none of them is perfect by oneself, and therefore we used several indexes in our Toolbox, which are described below

1. Partition Coefficient (PC): measures the amount of "overlapping" between clusters.

It is described below:

$$PC(c) = \frac{1}{N}\sum_{i=1}^{c}\sum_{j=1}^{N}(\mu_{ij})^2 \qquad (12)$$

Where μ_{ij} is the membership of data point *j* in cluster *i*. The disadvantage of PC is lack of direct connection to some property of the data themselves. The optimal number of cluster is at the maximum value.

2. Classification Entropy (CE): It measures the fuzziness of the cluster partition only, which is similar to the Partition Coefficient

$$CE(c) = -\frac{1}{N}\sum_{i=1}^{c}\sum_{j=1}^{N}\mu_{ij}log(\mu_{ij}), \qquad (13)$$

3. Partition Index (SC): is the ratio of the sum of compactness and separation of the clusters. It is a sum of individual cluster validity measures normalized through division by the fuzzy cardinality of each cluster.

$$SC(c) = \sum_{i=1}^{c} \frac{\sum_{j=1}^{N} (\mu_{ij})^m \left\| x_j - v_i \right\|^2}{N_i \sum_{k=1}^{c} \left\| v_k - v_i \right\|^2} \tag{14}$$

SC is useful when comparing different partitions having equal number of clusters. A lower value of SC indicates a better partition.

4. Separation Index (S): on the contrary of partition index (SC), the separation index uses a minimum-distance separation for partition validity.

$$S(c) = \frac{\sum_{i=1}^{c} \sum_{j=1}^{N} (\mu_{ij})^2 \left\| x_j - v_i \right\|^2}{N \min_{i.k} \left\| v_k - v_i \right\|^2} \tag{15}$$

5. Xie and Beni's Index (XB): it aims to quantify the ratio of the total variation within clusters and the separation of clusters.

$$XB(c) = \frac{\sum_{i=1}^{c} \sum_{j=1}^{N} (\mu_{ij})^m \left\| x_j - v_i \right\|^2}{N \min_{i.j} \left\| x_j - v_i \right\|^2} \tag{16}$$

The optimal number of clusters should minimize the value of the index.

6. Dunn's Index (DI): this index is originally proposed to use at the identification of "compact and well separated clusters". So the result of the clustering has to be recalculated as it was a hard partition algorithm.

$$DI(c) = \min_{i \in c} \{ \min_{j \in c i \neq j} \{ \frac{\min_{x \in C_i, y \in C_j} d(x, y)}{\max_{k \in c} \{ \max_{x, y \in C} d(x, y) \}} \} \} \tag{17}$$

The main drawback of Dunn's index is computational since calculating becomes computationally very expansive as c and N increase.

7. Alternative Dunn Index (ADI): The aim of modifying the original Dunn's index was that the calculation becomes simpler, when the dissimilarity function

between two clusters $(min_{x \in C_i, y \in C_j} d(x,y))$ is rated in value from beneath by the triangle-non equality:

$$d(x,y) \geq | d(y,v_j) - d(x,v_j) | \quad (18)$$

Where v_j is the cluster center of the *j*th cluster.

The only difference of *SC*, *S* and *XB* is the approach of the separation of clusters. In the case of overlapped clusters the values of *DI* and *ADI* are not really reliable because of re-partitioning the results with the hard partition method.

3.6.7 Taxonomy of clustering

At the very high end of the overall taxonomy we envision two main categories of clustering, known as hierarchical and objective function-based clustering. Different approaches to clustering data can be described with the help of the hierarchy shown in figure. At the top level, there is a distinction between hierarchical and partitional approaches (hierarchical methods produce a nested series of partitions, while partitional methods produce only one). These two clustering methods are further sub divided as shown in figure below.

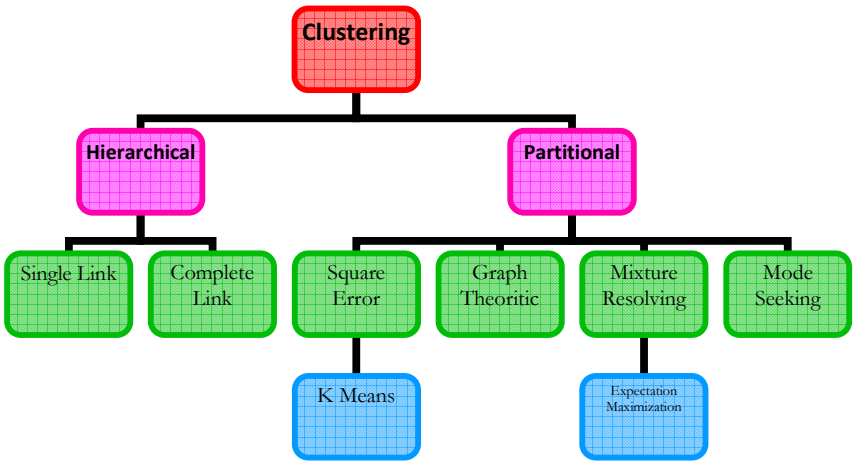

Figure 3.5: Taxonomy of Clustering Approaches

3.6.8 Types of Clustering

The taxonomy shown in Figure above must be supplemented by a discussion of cross-cutting issues that may (in principle) affect all of the different approaches regardless of their placement in the taxonomy.

3.6.8.1 Agglomerative vs. Divisive

This aspect relates to algorithmic structure and operation. An agglomerative approach begins with each pattern in a distinct (singleton) cluster, and successively merges clusters together until a stopping criterion is satisfied. A divisive method begins with all patterns in a single cluster and performs splitting until a stopping criterion is met.

3.6.8.2 Monothetic vs. Polythetic

This aspect relates to the sequential or simultaneous use of features in the clustering process. Most algorithms are polythetic; that is, all features enter into the computation of distances between patterns, and decisions are based on those distances. A simple monothetic algorithm considers features sequentially to divide the given collection of patterns. Here, the collection is divided into two groups using feature *x*1; the vertical broken line V is the separating line. Each of these clusters is further divided independently using feature *x*2, as depicted by the broken lines *H*1 and *H*2. The major problem with this algorithm is that it generates 2*d* clusters where *d* is the dimensionality of the patterns. For large values of *d* (*d* . 100 is typical in information retrieval applications), the number of clusters generated by this algorithm is so large that the data set is divided into uninterestingly small and fragmented clusters.

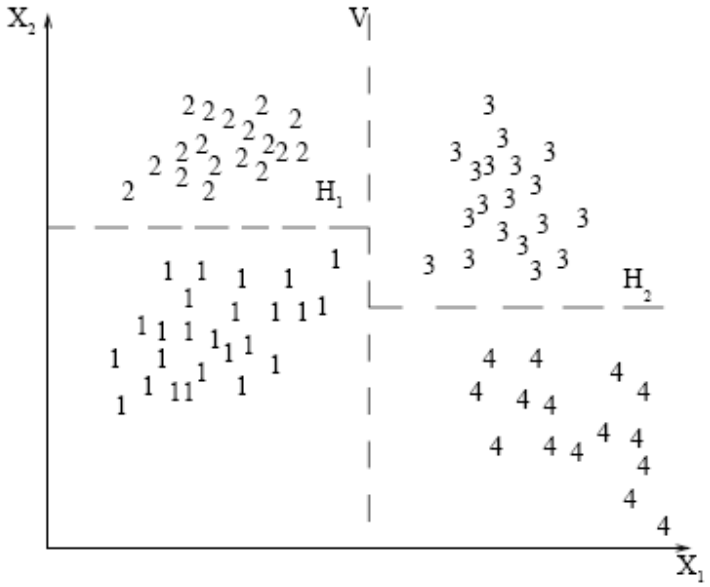

Figure 3.6: Monothetic partitional clustering

3.6.8.3 Hard vs. Fuzzy

A hard clustering algorithm allocates each pattern to a single cluster during its operation and in its output. A fuzzy clustering method assigns degrees of membership in several clusters to each input pattern. A fuzzy clustering can be converted to a hard clustering by assigning each pattern to the cluster with the largest measure of membership.

3.6.8.4 Deterministic vs. Stochastic

This issue is most relevant to partitional approaches designed to optimize a squared error function. This optimization can be accomplished using traditional techniques or through a random search of the state space consisting of all possible labeling.

3.6.8.5 Incremental vs. Non-incremental

This issue arises when the pattern set to be clustered is large, and constraints on execution time or memory space affect the architecture of the algorithm. The early history of clustering methodology does not contain many examples of clustering algorithms designed to work with large data sets, but the advent of data mining has fostered the development of clustering algorithms that minimize the number of scans through the pattern set, reduce the number of patterns examined during execution, or reduce the size of data structures used in the algorithm's operations.

3.6.9 Hierarchical Clustering Algorithms

The clustering techniques in this category produce a graphic representation of data. The construction of graphs (as these methods reveal the structure by considering each individual pattern) is done in two ways: bottom-up and top-down. The other names used reflect the way a structure is revealed. In the bottom-up mode known as an agglomerative approach, we treat each pattern as a single-element cluster and then successively merge the closest clusters. At each pass of the algorithm, we merge the two closest clusters. The process is repeated until we get to a single data set or reach a certain predefined threshold value. The top-down approach, known as a divisive approach, works in the opposite direction: we start with the entire set treated as a single cluster and keep splitting it into smaller clusters. Considering the nature of the process, these methods are

often computationally inefficient, with the possible exception of patterns with binary variables. The results of hierarchical clustering are usually represented in the form of dendrograms. Dendrograms are visually appealing graphical constructs: they show how difficult it is to merge two clusters. The distance scale shown at the right-hand side of the graph helps us quantify the distance between the clusters. This implies a simple stopping criterion: given a certain threshold value of the distance, we stop merging the clusters once the distance between them exceeds this threshold, meaning that merging two distinct structures does not seem to be feasible. The operation of a hierarchical clustering algorithm is illustrated using the two-dimensional data set in Figure. This figure depicts seven patterns labeled A, B, C, D, E, F, and G in three clusters. A hierarchical algorithm yields a dendrogram representing the nested grouping of patterns and similarity levels at which groupings change. The dendrogram can be broken at different levels to yield different clustering's of the data.

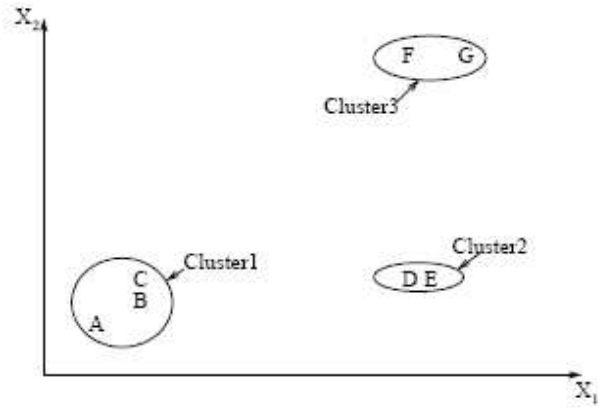

Figure 3.7: Points falling in three clusters.

An important issue is how to measure the distance between two clusters. Note that we have discussed how to express the distance between two patterns. Here, as each cluster may contain many patterns, computation of the distance is neither obvious nor unique.

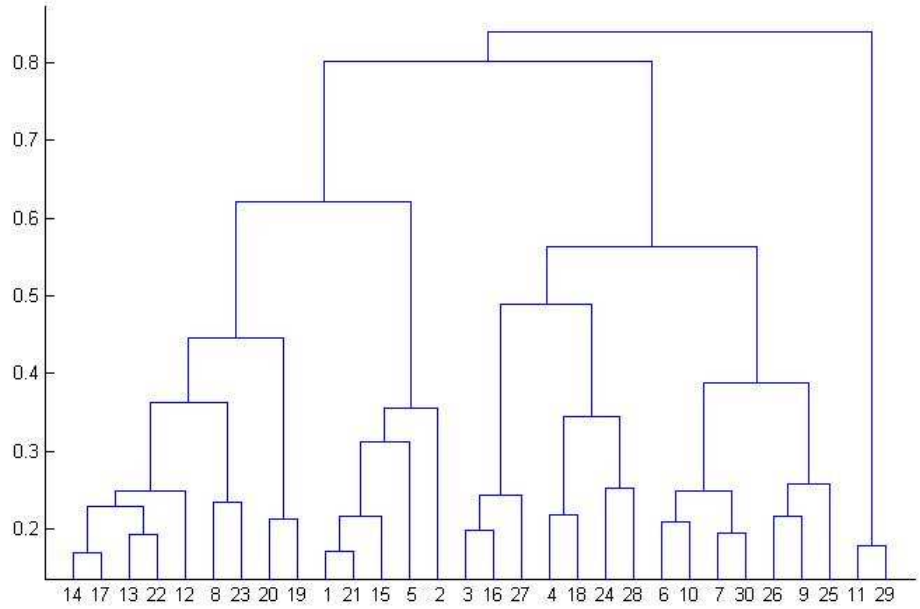

Figure 3.8: The dendrogram obtained using the single-link algorithm.

Consider two clusters, *A* and *B*, illustrated in Figure below. Let us describe the distance by *d*(*A*, *B*) and denote the number of patterns in *A* and *B* by *n*1 and *n*2, respectively. Intuitively, we can easily envision three typical ways of computing the distance between the two clusters.

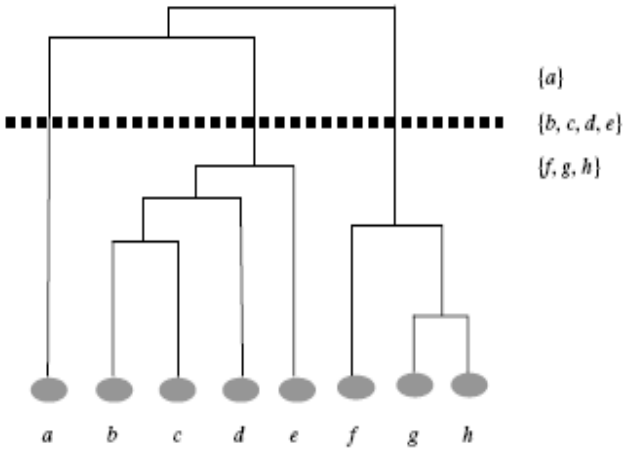

Figure 3.9: A dendrogram as a visualization of the structure of patterns**;**

3.6.10 Single-Link Method

The distance $d(A,B)$ is based on the minimal distance between the patterns belonging to A and B. It is computed in the form

$$d(A,B) = \min_{x \in A, y \in B} d(x,y)$$

In essence, the distance supports a sort of radically "optimistic" mode of expressing vicinity between clusters where we get involved the closest patterns located in different clusters. Clustering based on this distance is one of the most commonly used methods.

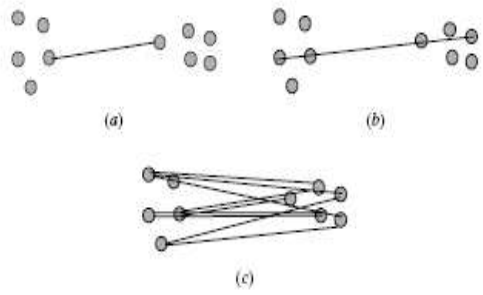

Figure 3.10: Two clusters and three main ways of computing the
(a) Single link, (b) complete link, and (c) group average link.

3.6.11 Complete-Link Method

This method is at the opposite end of the spectrum, as it is based on the distance between the two farthest patterns belonging to two clusters:

$$d(A,B) = \max_{x \in A, y \in B} d(x,y) \quad (19)$$

3.6.12 Group Average Link Method

In contrast to the two previous approaches, where the distance is determined on the basis of extreme values of the distance function, this method considers the average between the distances computed between all pairs of patterns, one from each cluster. We have

$$d(A,B) = \frac{1}{card(A)card(B)} \sum_{x \in A, y \in B} d(x,y) \quad (20)$$

Most hierarchical clustering algorithms are variants of the single-link, complete-link, and minimum-variance algorithms. Of these, the single-link and complete link algorithms are

most popular. These two algorithms differ in the way they characterize the similarity between a pair of clusters. In the single-link method, the distance between two clusters is the minimum of the distances between all pairs of patterns drawn from the two clusters (one pattern from the first cluster, the other from the second). In the complete-link algorithm, the distance between two clusters is the *maximum* of all pair wise distances between patterns in the two clusters. In either case, two clusters are merged to form a larger cluster based on minimum distance criteria.

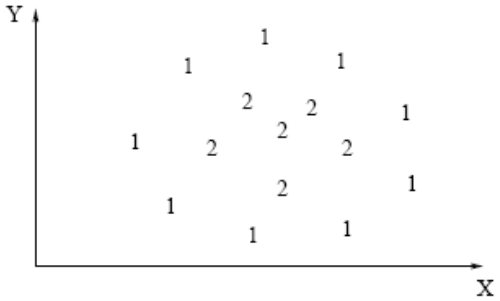

Figure 3.11: Two concentric clusters.

The complete-link algorithm produces tightly bound or compact clusters. The single-link algorithm, by contrast, suffers from a chaining effect. It has a tendency to produce clusters that are straggly or elongated. There are two clusters in Figures below, separated by a “bridge” of noisy patterns.

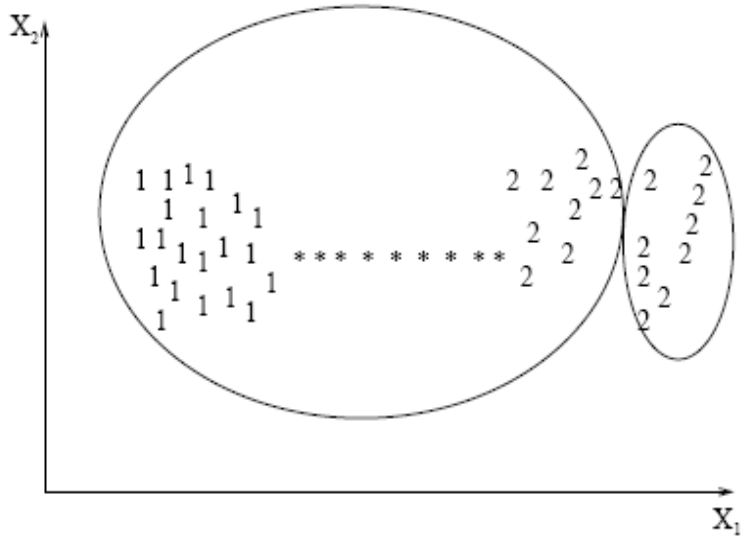

Figure 3.12: A single-link clustering of a pattern set containing two classes (1 and 2) connected by a chain of noisy patterns (*).

The single-link algorithm produces the clusters shown in figure 3.12, whereas the complete-link algorithm obtains the clustering shown in figure 3.13. The clusters obtained by the complete link algorithm are more compact than those obtained by the single-link algorithm; the cluster labeled 1 obtained using the single-link algorithm is elongated because of the noisy patterns labeled "*". The single-link algorithm is more versatile than the complete-link algorithm.

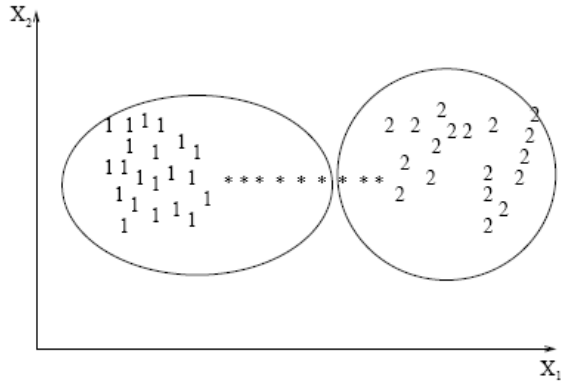

Figure 3.13: A complete-link clustering of a pattern set containing two classes (1 and 2) connected by a chain of noisy patterns (*).

However, from a pragmatic viewpoint, it has been observed that the complete link algorithm produces more useful hierarchies in many applications than the single-link algorithm.

3.7 Graph-Theoretic Clustering:

The best-known graph-theoretic divisive clustering algorithm is based on construction of the minimal spanning tree(MST) of the data , and then deleting the MST edges with the largest lengths to generate clusters. Figure below depicts the MST obtained from nine two-dimensional points. By breaking the link labeled CD with a length of 6 units (the edge with the maximum Euclidean length), two clusters ({A, B, C} and {D, E, F, G, H, I}) are obtained. The second cluster can be further divided into two clusters by breaking the edge EF, which has a length of 4.5 units. The hierarchical approaches are also related to graph-theoretic clustering.

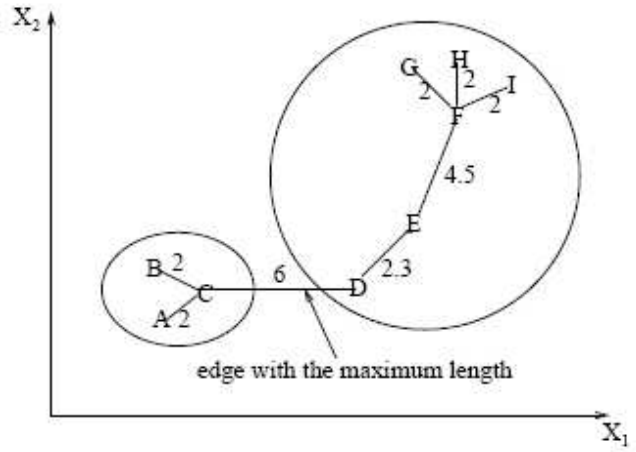

Figure 3.14: Use of the minimal spanning tree to form clusters.

Single-link clusters are sub-graphs of the minimum spanning tree of the data which are also the connected components. Complete-link clusters are maximal complete subgraphs, and are related to the node color ability of graphs. The maximal complete subgraph was considered the strictest definition of a cluster. A graph-oriented approach is used for non-hierarchical structures and overlapping clusters. The Delaunay graph (DG) is obtained by connecting all the pairs of points that are Voronoi neighbors. The DG contains all the neighborhood information contained in the MST and the relative neighborhood graph (RNG).

3.8 Nearest Neighbor Clustering

Since proximity plays a key role in our intuitive notion of a cluster, nearest neighbor distances can serve as the basis of clustering procedures. An iterative procedure; it assigns each unlabeled pattern to the cluster of its nearest labeled neighbor pattern, provided the distance to that labeled neighbor is below a threshold. The process continues until all patterns are labeled or no additional labeling occurs. The mutual neighborhood value (described earlier in the context of distance computation) can also be used to grow clusters from near neighbors.

3.9 Representation of Clusters

In applications where the number of classes or clusters in a data set must be discovered, a partition of the data set is the end product. Here, a partition gives an idea about the separability of the data points into clusters and whether it is meaningful to

employ a supervised classifier that assumes a given number of classes in the data set. However, in many other applications that involve decision making, the resulting clusters have to be represented or described in a compact form to achieve *data abstraction*. Even though the construction of a cluster representation is an important step in decision making, it has not been examined closely by research and the following representation schemes were suggested:

(1) Represent a cluster of points by their centroid or by a set of distant points in the cluster. Figure below depicts these two ideas.

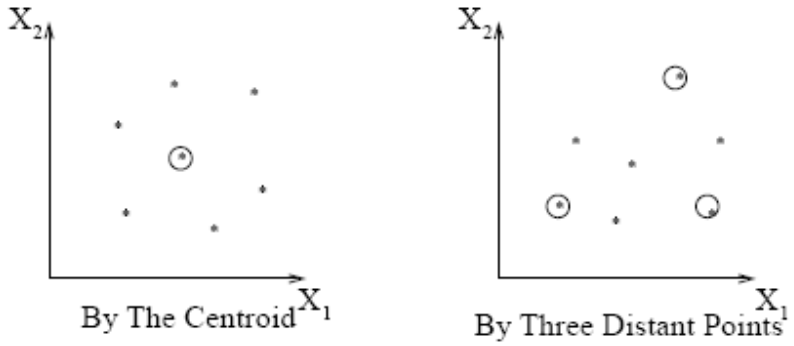

Figure 3.15: Representation of a cluster by points.

For example, the expression $[X_1>3]$ $[X_2<2]$ in Figure above stands for the logical statement '*X*1 is greater than 3' *and* '*X*2 is less than 2'. Use of the centroid to represent a cluster is the most popular scheme. It works well when the clusters are compact or isotropic. However, when the clusters are elongated or non-isotropic, then this scheme fails to represent them properly. In such a case, the use of a collection of boundary points in a cluster captures its shape well. The number of points used to represent a cluster should increase as the complexity of its shape increases. An important limitation of the typical use of the simple conjunctive concept representations is that they can describe only rectangular or isotropic clusters in the feature space. Data abstraction is useful in decision making because of the following:

(1) It gives a simple and intuitive description of clusters which is easy for human comprehension. In both conceptual clustering and symbolic clustering, this representation is obtained without using an additional step. These algorithms generate the clusters as well as their descriptions. A set of fuzzy rules can be

obtained from fuzzy clusters of a data set. These rules can be used to build fuzzy classifiers and fuzzy controllers.

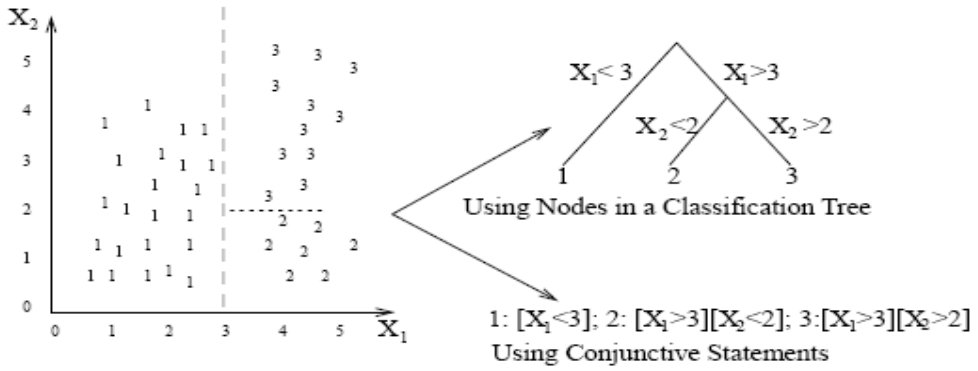

Figure 3.16: Representation of clusters by a classification tree

A partitional clustering like the *k*-means algorithm cannot separate these two structures properly. The single-link algorithm works well on this data, but is computationally expensive. So a hybrid approach may be used to exploit the desirable properties of both these algorithms. We obtain 8 sub clusters of the data using the (computationally efficient) *k*-means algorithm. Now the single-link algorithm can be applied on these centroids alone to cluster them into 2 groups. Here, a data reduction is achieved

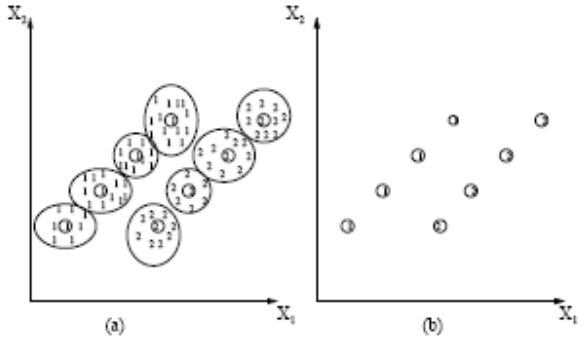

Figure 3.17: Data compression by clustering.

3.10 Square Error Clustering

The main objective of square error clustering is to obtain partition which for a fixed number of clusters minimizes the square error. Let there is a given set of n data in d

dimensions has to be partitioned in to K clusters $C = \{c_1, c_2, c_3 ..... c_k\}$ such that Ck has nk objects and each object is in one cluster only such that $\sum_{k=1}^{K} n_k = n$ (21)

The mean vector or center of cluster C_k is defined as the centroid of cluster

$$m^k = \frac{1}{n_k} \sum_{i=1}^{n_k} x_i^k \quad (22)$$

x_i^k is the i^{th} object belonging to cluster C_k. The square error for cluster C_k is the sum of the squared Euclidean distance between each object in C_k and its cluster center m^k. The squared error is represented by

$$e_k^2 = \sum_{i=1}^{n_k} \left(x_i^k - m^k \right)^T \left(x_i^k - m^k \right) \quad (23)$$

The square error of entire clustering containing K clusters is the sum of the within cluster variation and is represented by

$$E_K^2 = \sum_{k=1}^{K} e_k^2 \quad (24)$$

3.11 K-Means Clustering

Let $X = \{x_1, x_2, x_3 ..... x_n\}$ is a set of n dimensional objects to be clustered in to k clusters represented by $C = \{c_1, c_2, c_3 ..... c_k\}$. K Means clustering algorithm finds a partition such that the squared error between the empirical mean of the cluster and points in the cluster is minimized. Let μ_k is the mean of cluster c_k. The squared error between μ_k and c_k is defined as $J(c_k) = \sum_{x_i \in c_k} \left\| x_i - \mu_k \right\|^2$. (25)

The objective of K means algorithm is to minimize the sum of squared error over all K clusters. So the minimization function is written as $J(C) = \sum_{k=1}^{K} \sum_{x_i \in c_k} \left\| x_i - \mu_k \right\|^2$ (26)

Minimizing this objective function is known to be an NP-hard problem (even for K = 2). Thus K-means, which is a greedy algorithm, can only converge to a local minimum, even though recent study has shown with a large probability K-means could converge to the global optimum when clusters are well separated. K-means starts with an initial partition with K clusters and assign patterns to clusters so as to reduce the squared error.

Since the squared error always decrease with an increase in the number of clusters K (with J(C) = 0 when K = n), it can be minimized only for a fixed number of clusters. The main steps of K-means algorithm are as follows

3.12 Algorithm of K means algorithm

1. Select K points as initial centroids
2. **Repeat**
3. Form K clusters by assigning each point to its closest centroid
4. Recompute the centroid of each cluster
5. **Until** centroid do not change

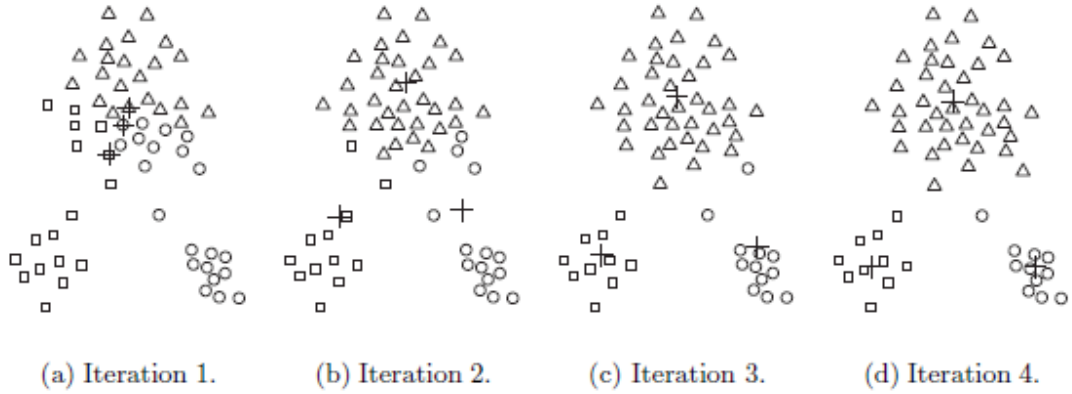

Figure 3.18: Different iteration to find out 3 clusters of a sample data

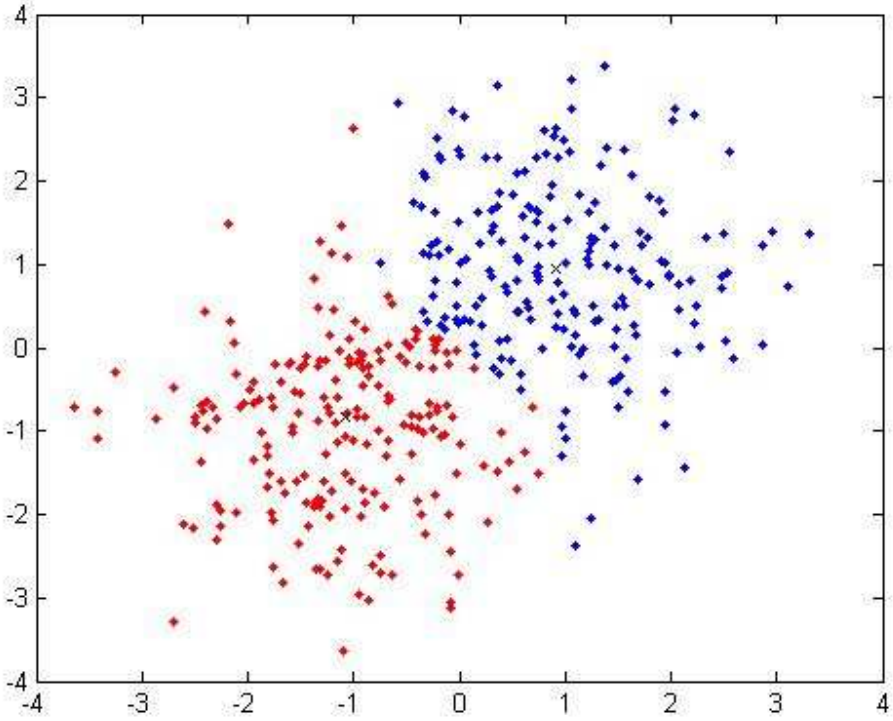

Figure 3.19: K-Means algorithm implemented in sample data

The simple K-means partitional clustering algorithm described above is computationally efficient and gives surprisingly good results if the clusters are compact, hyperspherical in shape and well-separated in the feature space. If the Mahalanobis distance is used in defining the squared error, then the algorithm is even able to detect hyperellipsoidal shaped clusters. Numerous attempts have been made to improve the performance of the basic K-means algorithm by

1. Incorporating a fuzzy criterion function, resulting in a fuzzy K-means (or c-means) algorithm,
2. Using genetic algorithms, simulated annealing, deterministic annealing, and tabu search to clustering Algorithms optimize the resulting partition
3. Mapping it onto a neural network for possibly efficient implementation.

However, many of these so-called enhancements to the K-means algorithm are computationally demanding and require additional user-specified parameters for which no general guidelines are available. A combination of algorithmic enhancements to a square-error clustering algorithm and distribution of the computations over a network of

workstations can be used to cluster hundreds of thousands of multidimensional patterns in just a few minutes.

Summary

In this chapter we discuss some traditional data clustering techniques and issue of data clustering in many applications. A distance is mainly the basic tool for measuring the clustering algorithm and many distance formulas have been discussed. In this we really discuss that how the distance play a main role in distinguish the vector that which data vectors are in which clusters. At last we discuss the K Means algorithm which main objective is how to minimize the sum of squared error overall K cluster.

CHAPTER 4

OVERVIEW OF ARTIFICIAL INTELLIGENCE TECHNIQUES

4.1 Introduction

From beginning artificial intelligence is widely used in various domains so as to get a better solution. In majority of the cases, researchers got much better results when they applied artificial intelligence algorithms in various engineering problems. Engineering problems have shown remarkable enhancement in performance and also efficiency when different artificial intelligence techniques were applied in comparison to conventional techniques. There are three basic domains in artificial intelligence viz fuzzy logic, artificial neural network and optimization techniques.

There is a wide variety of engineering application. These algorithms and their techniques have been applied to almost every engineering discipline. Presently, these techniques are applied on data mining, image processing, bio informatics, digital signal processing, measurement of concrete beams, vibration analysis, machine vision, machine control, navigation and communication equipment.

4.2 Fuzzy Logic

Fuzzy Logic is extension of Boolean logic. It incorporates partial values of truth. Instead of sentences being "Completely True" or "Completely False," Here in fuzzy logic they are assigned a value which represents their degree of truthness. In fuzzy systems, values are indicated by a number called as truth value. It lies in the range from 0 to 1. 0.0 represents absolute falseness and 1.0 represents absolute truth. Fuzzification is generalization of theory from discrete to continuous. Fuzzy logic is important to artificial intelligence. Fuzzy logic allows computers to answer 'to a certain degree' unlike Boolean logic (one extreme or the other). Computers are allowed to think more 'human-like'. Nothing in our perception is extreme. However, it is true only to a certain degree. In fuzzy logic, machines think in degrees. It can solve problems in the cases where there is no simple mathematical model. Fuzzy logic solves highly nonlinear processes. Fuzzy logic uses expert knowledge to make decisions.

4.3 History of Fuzzy Logic

Fuzzy logic was first invented as a representation scheme. It acts as calculus for uncertain or vague notions. It allows more human-like interpretations. Fuzzy logic has put reasoning in machines by resolving intermediate categories between notations like true/false, hot/cold etc. Fuzzy logic is a problem-solving control system methodology. It lends itself to implementation in systems ranging from small, simple, embedded micro-controllers to large, multi-channel, networked PC or workstation-based data acquisition control systems etc. It can be implemented in software, hardware, or a combination of both. Fuzzy logic provides a simple way to arrive at a definite conclusion. Conclusion is based upon ambiguous or vague, noisy, imprecise, or missing input information. Fuzzy logic's approach to control problems simply mimics how a person will make efficient decisions much faster.

In 1965, Professor L.A. Zadeh of the University of California, Berkely presented his seminal paper outlining fuzzy theory. In this paper he introduced fuzzy set theory and operation, fuzzy logic based controller etc. In 1970, fuzzy logic theory began to produce result in Japan, China and Europe. In 1987 sixteen station subway railway system was built. It worked with a fuzzy logic-based automatic train operation control system in Sendai, Japan. The ride of train is so smooth that riders do not need to hold straps. Fuzzy controller makes 70 percent fewer judgment errors in acceleration and braking. Fuzzy logic is a powerful problem-solving methodology. It has myriad of applications in embedded information processing and control. Fuzzy provides remarkably simple and definite conclusions. Conclusions are made from vague, ambiguous and imprecise information. Fuzzy logic resembles human decision making. It has ability to work from approximate data. It finds precise solutions. Classical logic requires a deep understanding of a system, exact equations, and precise numeric values. Fuzzy logic provides an alternative way of thinking. Fuzzy logic allows modeling complex systems while using a higher level of abstraction that originates from knowledge and experience. Fuzzy logic expresses knowledge with subjective concept like bright red, very hot, long time, very quick etc. are mapped into exact numeric ranges.

4.4 Architecture of Fuzzy Logic

Fuzzy logic is new and novel paradigm for an alternative design methodology. The fuzzy logic is applied in developing both linear and non-linear control systems. Fuzzy logic provides an alternative solution to non-linear control. It is closer to real world. Membership functions, rules and the inference process results in improved performance, simpler implementation, and reduced design costs. It handles non-linearity very efficiently. By using fuzzy logic, designers can realize superior features, lower development costs, optimized and better end product performance. Products can be brought to market faster and also more cost-effectively. Fuzzy logic is gaining increasing acceptance for the past couple of years. There are over two thousand commercially available products which use Fuzzy logic like washing machines, high-current trains etc. Every application can potentially realize the benefits of Fuzzy logic. These benefits are simplicity, performance, productivity and lower cost.

Fuzzy logic is a simple and flexible. Fuzzy logic handles problems with imprecise, vague and incomplete data. Fuzzy logic can model nonlinear functions of arbitrary complexity. If plant model is not available, or if the system is changing, then Fuzzy produces better solutions than conventional control techniques. Fuzzy systems match any set of input-output data. The fuzzy logic toolbox makes it easy by supplying adaptive techniques like adaptive neuro-fuzzy inference systems (ANFIS) and fuzzy subtractive clustering. Fuzzy logic models are called as fuzzy inference system. These model consist of number of conditions i.e "if-then" rules. For designer who understands the system better, these rules can be easily written. Flexible membership function scheme make fuzzy systems quite straightforward to create. Also it simplifies the design of systems. It ensures that it can be very easily updated and can be maintained the system over time. Fuzzy logic has several unique features. It is inherently robust. It does not require precise, noise-free inputs. It can be programmed to fail safely if a feedback sensor quits or gets destroyed. The output control is a smooth control function for a wide range of input variations. Fuzzy logic controller processes user-defined rules governing the target control system. Fuzzy logic can be modified and tweaked easily in order to improve or drastically alter the control system performance. Sensors can easily be

incorporated into fuzzy logic system. Just appropriate governing rules need to be generated. Fuzzy logic is not limited to a few feedback inputs or control outputs. It is not necessary to measure or compute rate-of-change in parameters for implementation. Sensor data providing some indication of a system's actions and reactions is just sufficient. Sensors required are inexpensive and imprecise. Thus it keeps the overall system cost and its complexity low. Because of the rule-based operation, a large number of inputs can be very easily processed and also numerous outputs can be generated. Defining the rule base is complex especially if there are too many inputs and outputs and their interrelations need to be defined. For this purpose control system is broken into smaller chunks. Several smaller fuzzy logic controllers are used which are distributed on system. Each fuzzy logic controller has different responsibilities. Fuzzy logic can control nonlinear systems. Non linear systems are difficult or impossible to model mathematically.

It is quite important to define the control objectives and control criteria. What is to be controlled? What has to be done to control the system? What kind of response should be there? What are the possible failure modes in the systems? It is necessary to determine the input and output relationships. A minimum number of variables are chosen for input to the fuzzy logic inference engine typically error and rate-of-change-of-error. Using the rule-based structure of fuzzy logic, the control problem is broken down into a series of IF X AND Y THEN Z rules. Rules must define the desired system output response for the given system input conditions. Number and complexity of rules depends on the number of input parameters be processed. It also depends upon the number fuzzy variables associated with each parameter. It is preferred to use at least one variable and its time derivative. A single instantaneous error parameter should be used along with its rate of change. Fuzzy logic membership functions need to be created which define the meaning (values) of input / output terms that are used in the rules. System need to be tested, evaluated for results. Tune the rules and membership functions. Until satisfactory results are obtained, retest the system.

4.5 Type I Fuzzy Set

If *X* is the universe of discourse (or universal set) consisting of all elements x of concern in a particular context or application, then a fuzzy set *A* in *X* is defined as the set of ordered pairs.

$$A=\{x,\mu_A(x)\,|\,x\in A\subset X\} \tag{27}$$

$\mu_A(x)$ is called membership function of fuzzy set A

$$\mu_A : x\in A\rightarrow[0,1] \tag{28}$$

This concept provides a mathematical way of characterizing the fuzzy set. The membership function of an object specifies the degree of similarity with the fuzzy set.

$$\mu_{pos}(x)=\begin{cases}1 & x\geq 3\\ \dfrac{x-1}{2} & 1\geq x<3\\ 0 & else\end{cases} \tag{29}$$

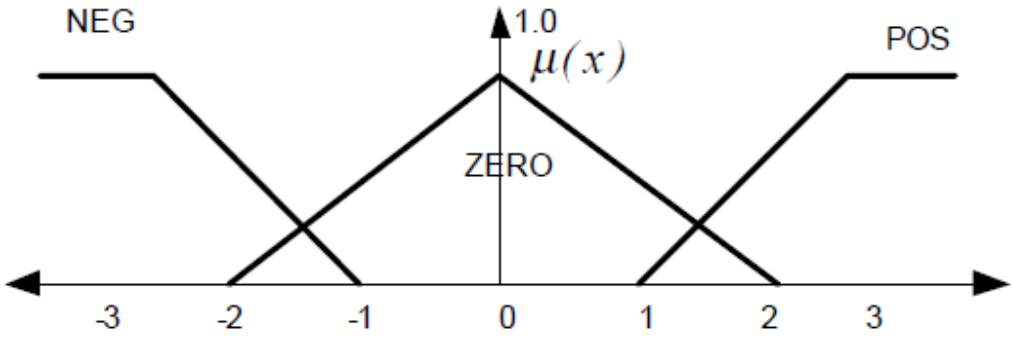

Figure 4.1: Fuzzy membership values

4.6 Operations of Fuzzy Logic

1. Equality: $A=B\Leftrightarrow\mu_A(x)=\mu_B(x)\forall x\in R_x$ (30)
2. Containment: $A\subset B\Leftrightarrow\mu_A(x)\leq\mu_B(x)\forall x\in R_x$ (31)
3. Fuzzy Union: $A\cup B\leftrightarrow\mu_{A\cup B}(x)=\max\left[\mu_A(x),\mu_B(x)\right]$ (32)
4. Fuzzy Intersection: $A\cap B\leftrightarrow\mu_{A\cap B}(x)=\min\left[\mu_A(x),\mu_B(x)\right]$ (33)
5. Fuzzy Complement: $\bar{A}\leftrightarrow\mu_{\bar{A}}(x)=1-\mu_A(x)$ (34)

Both the law of contradiction and law of excluded middle do not hold for the fuzzy sets.

4.7 Fuzzy **Membership Functions**

There are various types of membership function in fuzzy logic. Some standard membership functions are given here. Membership functions contain the membership values of elements in fuzzy set. Membership values can lie between 0 and 1.

4.7.1 Triangular Membership Function

It is given as

$$triangle(x,a,b,c)=\begin{cases}0 & x\le a\\ \dfrac{x-a}{b-a} & a\le x\le b\\ \dfrac{c-x}{c-b} & b\le x\le c\\ 0 & c\le x\end{cases} \tag{35}$$

$$triangle(x,a,b,c)=max\left(min\left(\frac{x-a}{b-a},\frac{c-x}{c-b}\right),0\right) \tag{36}$$

4.7.2 Trapezoidal Membership Function

It is given as

$$trapezoid(x;a,b,c)=max\left(min\left(\frac{x-a}{b-a},1,\frac{d-x}{d-c}\right),0\right) \tag{37}$$

$$trapezoid(x:a,b,c,d)=\begin{cases}0 & x\le a\\ \dfrac{x-a}{b-a} & a\le x\le b\\ 1 & b\le x\le c\\ \dfrac{d-x}{d-c} & c\le x\le d\\ 0 & d\le x\end{cases} \tag{38}$$

4.7.3 Gaussian Membership Function

It is given as

$$gaussian(x;c,\theta)=e^{-\frac{1}{2}\left(\frac{x-c}{\sigma}\right)^2} \tag{39}$$

4.7.4 Bell Membership Function

It is given as

$$bell(x, a, b, c) = \frac{1}{1 + \left|\frac{x - c}{a}\right|^{2b}} \quad (40)$$

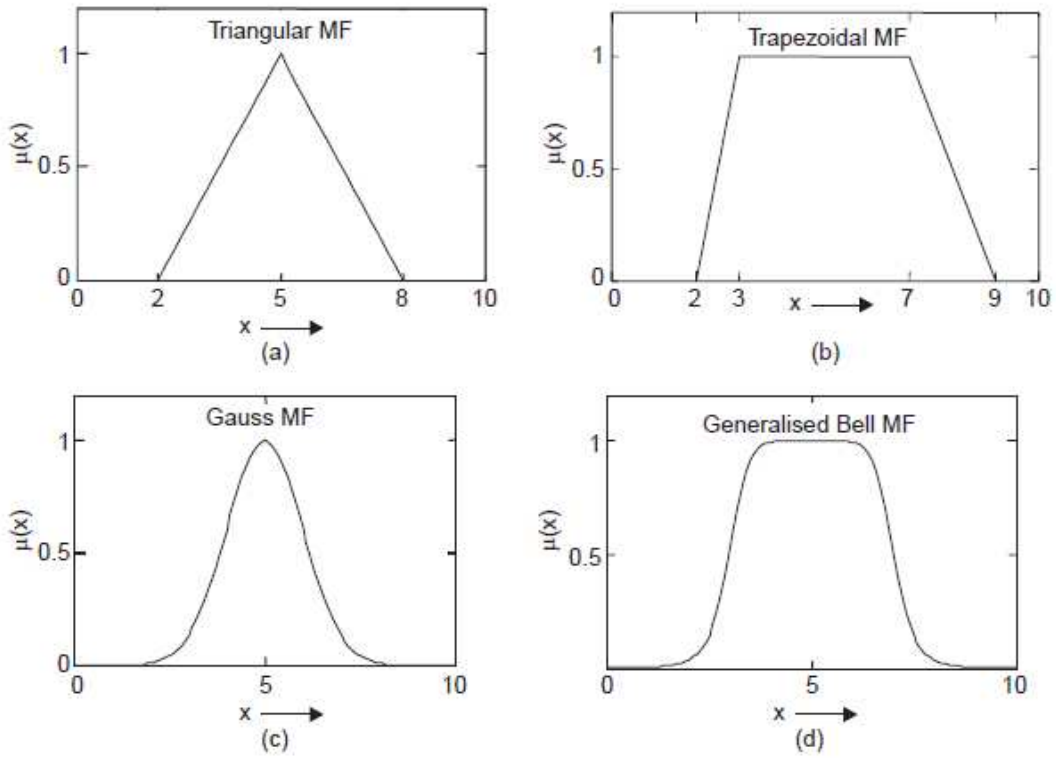

Figure 4.2: Various types of fuzzy logic membership functions
(a) Triangular Membership
(b) Trapezoidal Membership
(c) Gauss Membership
(d) Generalized Bell Membership

4.8 Fuzzy Inference System

Fuzzy inference systems (FIS) are rule-based systems. It is based on fuzzy set theory and fuzzy logic. FIS are mappings from an input space to an output space. FIS allows constructing structures which are used to generate responses (outputs) for certain stimulations (inputs). Response of FIS is based on stored knowledge (relationships between responses and stimulations). Knowledge is stored in the form of a rule base. Rule base is a set of rules. Rule base expresses relations between inputs of system and its expected outputs.

Knowledge is obtained by eliciting information from specialists. These systems are usually known as fuzzy expert systems. Another common denomination for FIS is fuzzy knowledge-based systems. It is also called as data-driven fuzzy systems. FIS are usually divided in two categories viz. multiple input and multiple output (MIMO) systems and Multiple Input and Single Output (MISO) systems, the system returns several outputs based on the inputs which it receives. Multiple input and single output (MISO) systems are those where only one output is returned from multiple inputs. MIMO systems are decomposed into a set of MISO systems which work in parallel. In terms of inference process there are two main classes of FIS viz. the Mamdani-type FIS and the Takagi-Sugeno- Kang (TSK) type FIS. TSK FIS is also called as Sugeno FIS.

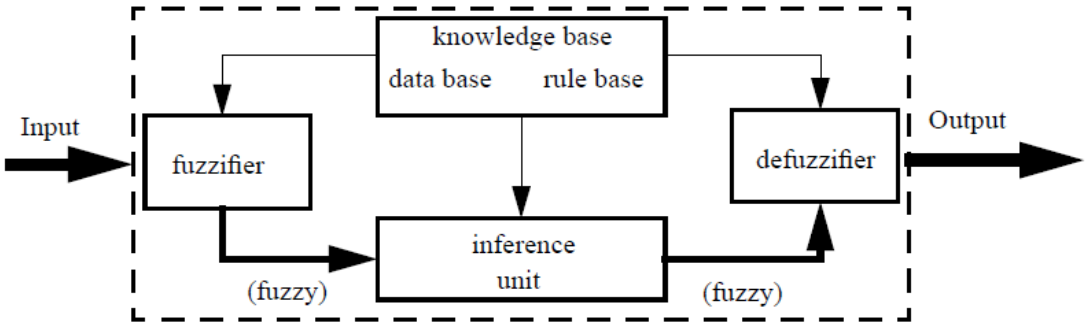

Figure 4.3: Block diagram of fuzzy logic System

4.9 Mamdani Type FIS

In mamdani based fuzzy inference system, inputs and output have an If-then rules. A typical rule in a mamdani fuzzy model is: IF X is Negative Big AND Y is Negative Small THEN Z is Zero

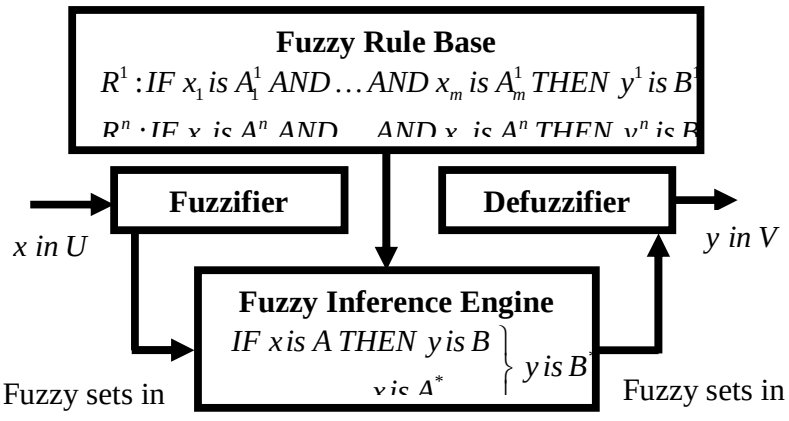

Figure 4.4: Mamdani based fuzzy inference system

4.10 Sugeno Type FIS

Sugeno-type systems are used to model any inference system in which output membership functions are either linear or constant. This fuzzy inference system was introduced in 1985. It is also called as Takagi-Sugeno-Kang. Sugeno output membership functions (z) are either linear or constant. A typical rule in a Sugeno fuzzy model is:

If Input 1 = x and Input 2 = y, then Output is z = ax + by + c

For a zero-order Sugeno model, the output level z is a constant (a=b =0).

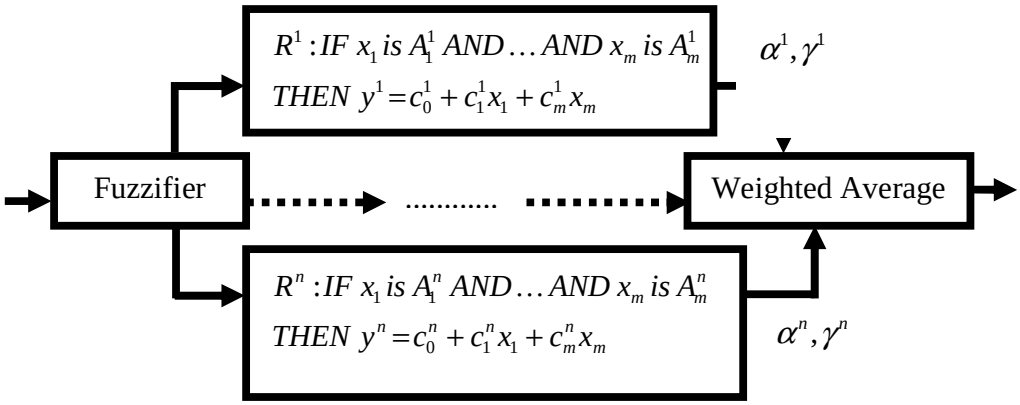

Figure 4.5: Sugeno type fuzzy inference System

Both Sugeno and Mamdani FIS can be used to perform the similar tasks. Rule base and fuzzification remain same for the variables. There are various defuzzifiers that can be chosen for a Mamdani FIS. These defuzzifiers also originate similar results in a Sugeno FIS. There is a certain overlap between both types of systems. Mamdani FIS is more widely used. It is used for decision support applications, because its intuitive and interpretable nature. Consequents of the rules in a Sugeno FIS do not have a direct semantic mean. This means that they are not linguistic terms. Also, this interpretability is partially lost. Sugeno FIS rules consequents can have many parameters per rule as per input values. Thus, Sugeno FIS gets translated into more degrees of freedom in its design as compared to Mamdani FIS. Thus it provides more flexibility. Many parameters can be used in the consequents of the rules of a Sugeno FIS. A zero order Sugeno FIS can reasonably approximate a Mamdani FIS. In computational terms, a Sugeno FIS is more efficient than a Mamdani FIS. It is so because; Sugeno FIS does not involve

computationally expensive defuzzification process. Also, a Sugeno FIS always generates continuous surfaces. The continuity of the output surface is quite important. Any existence of discontinuities will result in similar inputs originating substantially different outputs. It will be a situation which is undesirable from the control/ monitoring perspective. Because of continuous structure of output functions, a Sugeno FIS is also better and adequate for functional analysis than a Mamdani FIS.

4.11 Fuzzy Cartesian product and Composition

Let R be a relation that relates elements from universe X to universe Y. Let S be the relation that relates elements from universe Y to universe Z. Let T relates the same elements in universe that R contains to the same elements in the universe Z that S contains. The two methods of the composition operation are:

– Max–min composition,

– Max–product composition.

The max–min composition is defined by the set-theoretic and membership function–theoretic expressions:

$$T = RoS \tag{41}$$

$$\chi_T(x,z) = \underset{y \in Y}{V}(\chi_R(x,y) \wedge \chi_S(y,s)). \tag{42}$$

The max–product composition is defined by the set-theoretic and membership function-theoretic expressions:

$$T = RoS \tag{43}$$

$$\chi_T(x,z) = \underset{y \in Y}{V}(\chi_R(x,y) \bullet \chi_S(y,s)). \tag{44}$$

Let A be a fuzzy set on universe X and B be a fuzzy set on universe Y, then the Cartesian product between fuzzy sets A and B will result in a fuzzy relation R which is contained with the full Cartesian product space or

$$A \times B = R \subset X \times Y, \tag{45}$$

where the fuzzy relation R has the membership function.

$$\mu_R(x,y) = \mu_{A \times B}(x,y) = \min\left(\mu_A(x), \mu_B(y)\right). \tag{46}$$

Each fuzzy set could be thought of as a vector of membership values; each value associated with a particular element in each set. For example, for fuzzy set A that has

four element, hence column vector size 4×1 and for fuzzy set (vector) B that has the five element, hence the row vector of 1×5. The resulting fuzzy relationship R will be represented by a matrix of size 4×5 i.e. R will have four rows and five columns.

4.12 Defuzzification Techniques

Defuzzification converts the fuzzy outputs back to crisp values. There are different defuzzification methods given as

1. Max Membership $\mu_c(z^*) \geq \mu_c(z)$ for all $z \in Z$ (47)

2. Centroid $$z^* = \frac{\int \mu_c(z).z dz}{\int \mu_c(z) dz} \quad (48)$$

3. Weighted average $$z^* = \frac{\sum \mu_c(z).z}{\sum \mu_c(z)} \quad (49)$$

4. Mean-Max $$z^* = \frac{a+b}{2} \quad (50)$$

5. Center of Sum $$z^* = \frac{\int_z z^* \sum_{k=1}^{n} \mu_c(z) dz}{\int_z \sum_{k=1}^{n} \mu_c(z) dz} \quad (51)$$

6. Center of Largest Area $$z^* = \frac{\int \mu_c(z) z dz}{\int \mu_c(z) dz} \quad (52)$$

4.13 Principles of Fuzzy Logic

Fuzzy logic is not really “fuzzy”. A fuzzy controller has a set of rules that is used to calculate the final control action. Each rule is linguistic expression about the control action to be taken in response to a given set of process conditions. The condition may include ‘AND’ and ‘OR’ condition, e.g. If set point is positive big and error change is positive small than actuator output is negative big.

The step for designing a simple fuzzy logic control system is as follows:-

1. Identify the variables (Input, states, and output) of the plant.
2. Partition the universe of discourse or the interval spanned by each variable into a number of fuzzy subsets, assigning each a linguistic labels.(subset includes all the elements in the universe).

3. Assign or determine a membership function for each fuzzy subset.
4. Assign the fuzzy relationship between the ‘inputs’ or the ‘states’ fuzzy subsets on the one hand and the ‘output’ fuzzy subsets on the otherhand, thus forming the rule base.
5. Choose appropriate scaling factor for the input and variables in order to normalize the variable to the [0,1] or [-1,1] interval.
6. Fuzzily the inputs to the controller.
7. Use fuzzy appropriate reasoning to input the output contributed from each rule.
8. Aggregate the fuzzy outputs recommended by each rule.
9. Apply defuzzification to form a crisp output.

Summary

In this chapter fuzzy logic is discussed in which we discuss all the architecture of the fuzzy logic in which how to create rules, making membership function, different types of membership function, type I fuzzy set, fuzzy inference system, types of fuzzy system, fuzzy Cartesian product and max-min composition. At last we discussed the defuzzifications techniques and the method to get the crisp value from the fuzzy value.

CHAPTER 5
AI BASED DATA CLUSTERING TECHNIQUES

5.1 Introduction

In traditional clustering algorithm, one object is assigned in to only one cluster. This is valid till the clusters are disjoint and separate. But if the clusters are touching each other or they are overlapping, then one object can belong to more than one cluster. In this case fuzzy clustering comes in to existence.

In fuzzy clustering, one object can be clustered in more than one cluster according to the degree of membership function.

Let a set of objects $X = \{x_1, x_2, x_3.....x_n\}$ has to be clustered in to $C = \{c_1, c_2, c_3.....c_k\}$. $\delta(x, C_i)$ denote the similarity between object x and cluster C_i. The membership function for object x and cluster C_i is represented by the following equation

$$fc_i(x) = \frac{P_i \delta(x, C_i)}{\sum_{k=1}^{K} P_k \delta(x, C_k)} \qquad (53)$$

$P_k = \frac{n_k}{n}$ is the relative size of cluster C_k. This membership function is non negative. Membership function can also be expressed in terms of Euclidian distance. This is represented in following equation

$$fc_k(x) = \frac{1 - \left(\frac{1}{\beta}\right) d\left(x, m^k\right)}{K - \left(\frac{1}{\beta}\right) \sum_j d\left(x, m^j\right)} \qquad (54)$$

$d\left(x, m^k\right)$ represent the Euclidian distance between vector x and centroid m^k of cluster C_k. β denotes the belongingness.

Traditional clustering approaches generate partitions; in a partition, each pattern belongs to one and only one cluster. Hence, the clusters in a hard clustering are disjoint. Fuzzy clustering extends this notion to associate each pattern with every cluster using a membership function. The output of such algorithms is a clustering, but not a partition. Contrary to other methods of clustering, the fuzzy clustering methods provide a number

of membership values that indicate the degree of membership of the different samples to the different groups. These values can be very important for understanding the data and for assessing how natural the groups are. Other methods like for instance hierarchical clustering methods give crisp groups as result, i.e. the membership values are either 0 or 1.

The binary character of partitions described so far may not always be a convincing representation of the structure of data. Consider the set of two-dimensional patterns illustrated in figure 5.1. While we can easily detect three clusters, their character is different. The first one is quite compact, with highly concentrated patterns. The other two exhibit completely different structures. They are far less condensed, with several patterns whose allocation to a given cluster may be far less certain. In fact, we may be tempted to allocate them to two clusters with varying degrees of membership. This simple and appealing idea forms a cornerstone of fuzzy sets—collections of elements with partial membership in several categories. As illustrated in Figure below, the two identified patterns could easily belong to several clusters.

These situations of partial membership occur quite often. Structures (clusters) may not be well separated for a variety of reasons. There may be noise or lack of discriminatory power of the feature space in which the patterns are represented. Some patterns could be genuine outliers. Some of them could be borderline cases and thus are difficult to classify. As a result, they may require far greater attention. A clustering algorithm that could easily provide detailed insight into the membership grades of the patterns could be a genuine asset. Let us assume that this is true and that the partition matrix now consists of grades of membership distributed in the unit interval.

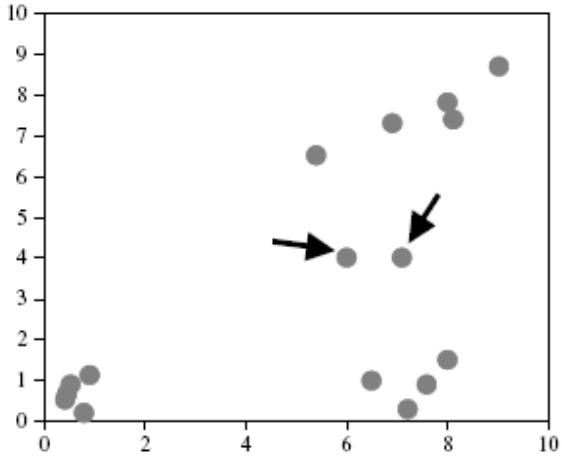

Figure 5.1: Three clusters with patterns of partial membership (belongingness) in the clusters. The patterns of borderline character are pointed to by the arrows.

For the data in Figure above, the partition matrix comes with the entries shown. The results are highly appealing, and they fully reflect our intuitive observations: patterns 6 and 7 have a borderline character, with membership grades in one of the clusters at the 0.5 level. The values in the partition matrix quantify the effect of partial membership.

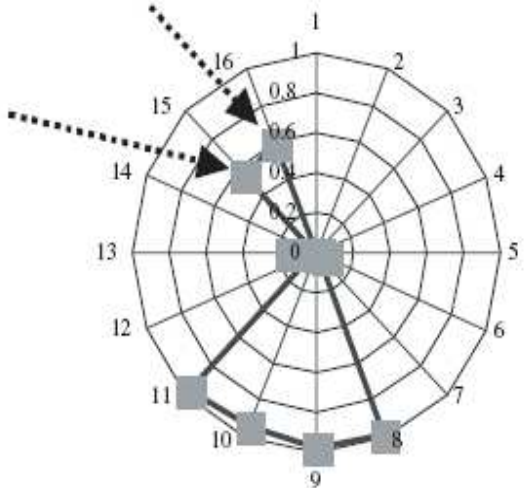

Figure 5.2: Star diagram of the fuzzy partition matrix

It seems similar to the one guiding the optimization of the Boolean (two-valued) partition matrix with only one significant exception: we consider *U* to be a fuzzy partition, viz., a matrix with the entries confined to the unit interval that satisfies two important requirements:

- The clusters are nontrivial. For each cluster (i = 1, 2, . . ., c) we end up with a nonempty construct that does not include all patterns
- The total membership grades sum to 1, so the distribution of belongingness is equal to 1.

The membership values are here denoted by uij and are collected in a matrix denoted by U. Each line in U corresponds to a sample and each column to a group. The u_{ij} values in each line sum to 1. This means that the membership values of each sample (i) to the different groups (j) sum to 1. The number of samples is denoted by N and the number of groups by C. The number C has to be fixed during fuzzy clustering. However, different choices of C can be tested and the one with the best results can be selected. Indices have been developed for studying the quality of the splitting. As for any other clustering method, fuzzy clustering is based on a distance measure. In classical applications and theory, the distances are either Euclidean or Mahalanobis distances (in the whole space or in a subspace). For this introductory section we assume that the distance is Euclidean or Mahalanobis with the same fixed covariance matrix for all groups. There exist several fuzzy clustering algorithms. For example, if we consider the so called fuzzy k-means algorithm. This is based on minimizing the following criterion:

$$J = \sum_{j=1}^{c} \sum_{i=1}^{N} u_{ij}^{N} D_{ij}^{2} \qquad (55)$$

The user can determine the parameter m ≥ 1 in the exponent of u. An m equal to 1 gives crisp subsets, while larger values of m give fuzzy subgroups which also may be more robust to outliers. A much used value is m=2. This will be used for the rest of the thesis. Note that minimization of the J-criterion is natural because small values of u combined with large values of D (and vice versa) are favored

5.2 Fuzzy C Means

Fuzzy C Means (FCM) is a feature clustering technique wherein each feature point belongs to a cluster by some degree that is specified by a membership grade. These kind of clustering algorithms are known as objective function based clustering. Given M dimensional database of size N where N is the total number of feature vectors and M is the dimension of each feature vector. FCM assigns every feature vector a membership grade for each cluster. The problem is to partition the database based on some fuzziness

criteria using membership values. To find membership values, the partition matrix U of size N x C is calculated that defines membership degrees of each feature vector. The values 0 and 1 in U indicate no membership and full membership respectively. Grades between 0 and 1 indicate that the feature point has partial membership in a cluster. Looking at the picture, we may identify two clusters in proximity of the two data concentrations. We will refer to them using “A” and “B”. In the first approach shown in this tutorial - the k-means algorithm - we associated each datum to a specific centroid; therefore, this membership function looked like this:

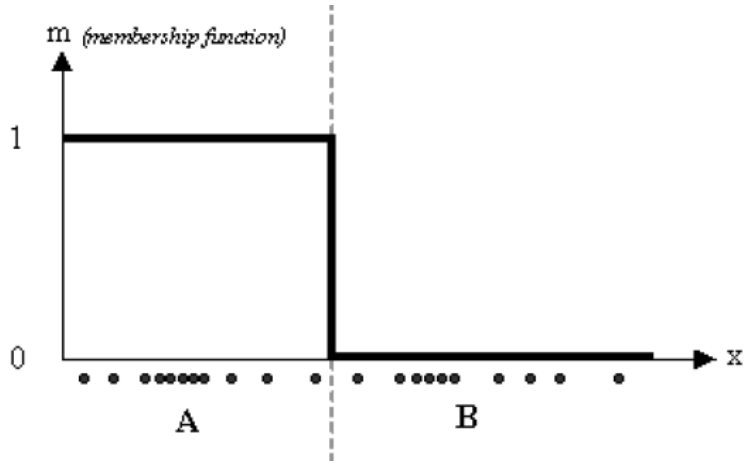

Figure 5.3: Hard or crisp clustering of data

In the FCM approach, instead, the same given datum does not belong exclusively to a well defined cluster, but it can be placed in a middle way. In this case, the membership function follows a smoother line to indicate that every datum may belong to several clusters with different values of the membership coefficient.

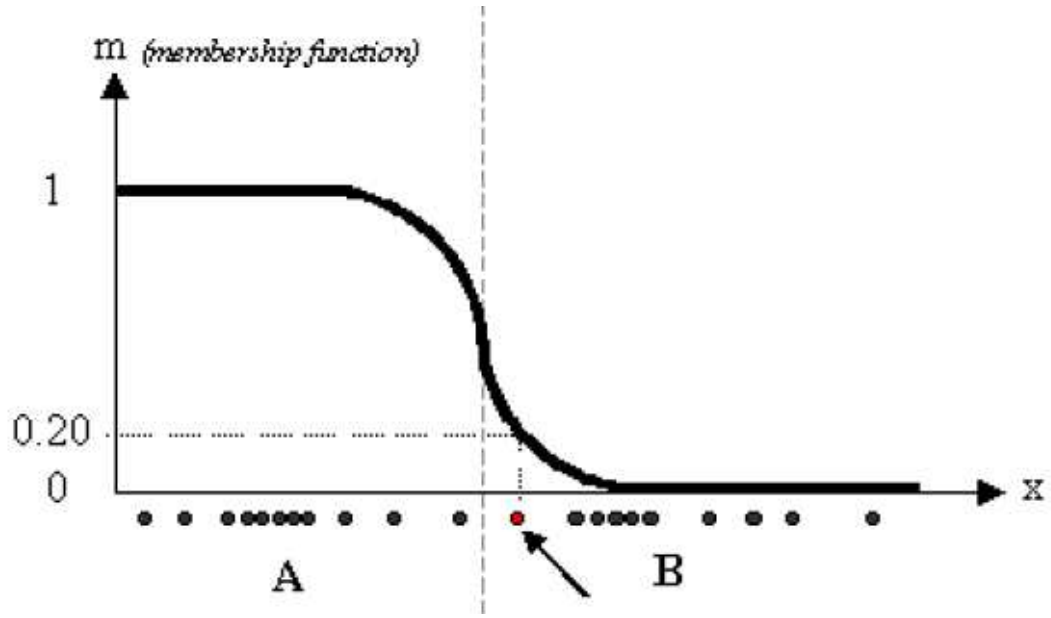

Figure 5.4: Membership of data in fuzzy clustering

The following steps are involved in training the database using FCM technique

5.2.1 Initialization of the partition matrix

Initially a fuzzy partition matrix U is generated that is of size N x c, where c is number of clusters and N is total number of feature vectors. Subject to the constraint that $\sum_{j=1}^{c} U_{ij} = 1$

i = {1,2,3 -------- N}

5.2.2 Calculation of fuzzy centers

The fuzzy centers are calculated using the partition matrix generated

$$C_j = \frac{\sum_{i=1}^{N} U_{ij}{}^{m} x_i}{\sum_{i=1}^{N} U_{ij}{}^{m}} \qquad (56)$$

where $m \geq 1$ is a fuzzification exponent. The larger the value of *m* the fuzzier the solution will be. This indicates the number of iterations that is required for clustering. xi is ith feature vector. The value of *i* ranges from 1 to *N* (total number of templates in the database).

5.2.3 Updating membership and cluster centers

FCM is an iteration loop. The method of clustering is based on minimization of the objective function defined by $J = \sum_{i=1}^{N}\sum_{j=1}^{C} U_{ij}^{m} \left\| x_i - c_j \right\|^2$ (57)

Uij describes the degree of member of feature set (*xi*) with cluster *cj*. ||*|| represents norm between *xi* and cluster center *cj* given by $\left\| x_i - c_j \right\|^2 = \left(x_i - c_j \right)^T A \left(x_i - c_j \right)$ (58)

where *A* is identity matrix for Euclidean distance used here. At every iteration the membership matrix is updated using

$$U_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{\left\| x_i - c_j \right\|}{\left\| x_i - c_j \right\|} \right)^{\frac{2}{m-1}}} \quad (59)$$

The revised membership matrix is used for updating the cluster centers.

The iteration will stop when $\max_{ij} \left\{ \left| U_{ij}^{m+1} - U_{ij}^{m} \right| \right\} < \varepsilon$ (60)

where ε is a termination criteria. The value of ε ranges between 0 and 1.

5.3 Algorithm

1. Fix 1 < m < ∞, initial partition matrix U (N x c) and termination criteria
2. Calculate fuzzy cluster centers
3. Update membership matrix
4. Calculate change in membership function $\Delta = \left\| U^{m+1} - U^{m} \right\| = \max_{ij} \left| U_{ij}^{m+1} - U_{ij}^{m} \right|$

   If $\Delta \le \varepsilon$ then set m = m+1 and go to step 2

   Or else stop

5.4 Possibilistic C-Means (PCM)

PCM constitutes a more robust algorithm that relaxes the constraint causing the relative definition of membership degrees in FCM. The fir coefficients they are based on, usually denoted tir, measure the absolute resemblance between data points and cluster centers.

$$t_{ir} = \left(1 + \left(\frac{d_{ir}^2}{\eta_r} \right)^{\frac{1}{m-1}} \right)^{-1} \quad (61)$$

where ηr is a parameter that evaluates the cluster diameter and can be defined a priori or defined from initialisations. Outliers, that are far away from all clusters are then associated with small weights for all clusters and thus do not influence their parameters. PCM suffers from a coincident cluster problem: in some cases, clusters are confounded whereas natural subgroups in the data are overlooked. Moreover, it has been shown that the objective function global minimum is obtained when all clusters are coincident. Satisfying results are obtained with PCM because the optimization scheme leads to local minima and not the global minimum. This property is not satisfying from a theoretical point of view.

5.5 Possibilistic Fuzzy C-Means

To solve the PCM coincident cluster problem, Pal et al. propose to combine PCM and FCM: they argue that both possibilistic and membership coefficients are necessary to perform clustering, respectively to reduce the outlier influence and to assign data points to clusters. Therefore, they take into account both relative and absolute resemblance to cluster centres. In the PFCM algorithm, the combination is performed through a weighted sum, in the form $f_{ir} = au_{ir}^{m1} + bt_{ir}^{m2}$ (62)

Here uir are the membership degrees defined in eq. (58) and tir the possibilistic co-efficient defined in eq. (62) with ηr replaced by ηr/b. a, b, m1 and m2 are user-defined parameters.

5.6 Other approaches

There exist many other approaches to solve the merging cluster problem or that of the outlier sensitivity: the cluster repulsion method e.g. includes in the objective function an additional term to impose repulsion between clusters and prevent their merging. The noise clustering algorithm has a rejecting process for outliers or noisy data, McLachlan and Peel use tstudent distributions to better model outliers thanks to heavier tailed distributions.

Summary

In this chapter we discussed the artificial intelligent based data clustering techniques and its main features. Fuzzy C Means (FCM) is a feature clustering technique wherein each feature point belongs to a cluster by some degree that is specified by a membership grade. Possibillistic C Means (PCM) constitutes a more robust algorithm that relaxes the

constraint causing the relative definition of membership degrees in FCM. Also there is some other techniques in which distance is the main concerned.

CHAPTER 6
DISEASE DIAGNOSIS USING DATA CLUSTERING AND FUZZY INFERENCE MECHANISM

6.1 Introduction

Medical diagnosis refers to the process of attempting to determine or identify a possible disease or disorder and the opinion reached by the process. A medical diagnosis is an attempt at classification. Just as chemists attempt to classify naturally occurring elements into a periodic table and biologists attempt to classify plants and animals into species and genii so too do physicians attempt to classify disease into separate and distinct categories that allow medical decisions about treatment and prognosis to be made. Physicians usually begin the diagnostic process by observing the patient for specific signs and symptoms and by taking a specific history, e.g. how did these signs and symptoms come about? Specific signs, symptoms and historical clues allow the physician to perform a specific physical examination and order specific diagnostic imaging. The provider usually formulates a "short list" of likely diagnoses and may obtain further testing to confirm or rule-out competing diagnoses before providing treatment. Diagnosis may be performed by health care providers such as a physician, dentist, podiatrist, nurse practitioner, physical therapist or physician assistants. Medical tests commonly performed are: measuring blood pressure, measuring glucose in body, checking the pulse rate, listening to the heart with a stethoscope, performance tests such as treadmill ambulation, vital capacity, balance tests, pathological and neurological tests such as reflexes, sensation and muscle testing, urine test, fecal tests, saliva tests, blood test, medical imaging, electrocardiogram, hydrogen breath test and occasionally biopsy. Diagnosis and etiology are often used synonymously, especially since germ theory began to link causative agents with disease. Later, the development of antibiotics allowed physicians to treat the cause (pneumonia bacilli) and cure the disease. This effective linkage of diagnosis with etiology is widely accepted, even by physicians. New taxonomies of medical classification, however, do not require the etiology of disease in order to treat the patient. For instance, a common disorder such as pneumonia was nevertheless used as a diagnosis before the germ theory was accepted, and the disease

was defined as a complex of many symptoms consisting of cough, sputum production, fever and chills. Later, as the actual cause was assigned to micro-organisms, the term diagnosis included the causality, e.g., pneumococcal pneumonia, suggesting not only a spectrum of symptoms but also a cause for the symptoms. Widespread disagreement exists between medical and psychiatric practitioners as to whether causalities for various diseases and disorders are known or not. If causalities are assumed to be known, then authentic cures can be obtained by correcting the causal abnormalities. If causalities are assumed to be unknown, then palliative treatments to reduce symptoms are the best treatments possible. A provider's job is to know the human body and its functions in terms of normality (homeostasis). The four cornerstones of diagnostic medicine, each essential for understanding homeostasis, are: anatomy (the structure of the human body), physiology (how the body works), pathology (what can go wrong with the anatomy and physiology) and psychology (thought and behaviour). Once the provider knows what is normal and can measure the patient's current condition against those norms, she or he can then determine the patient's particular departure from homeostasis and the degree of departure. This is called the **diagnosis**. Once a diagnosis has been reached, the provider is able to propose a management plan, which will include treatment as well as plans for follow-up. From this point on, in addition to treating the patient's condition, the provider educates the patient about the causes, progression, outcomes, and possible treatments of his ailments, as well as providing advice for maintaining health. It should be noted however, that medical diagnosis in psychology or psychiatry is problematic. Apart from the fact that there are differing theoretical views toward mental conditions and that there are few "lab" tests available for various major disorders (e.g., clinical depression), a causal analysis with respect to symptomatology and disorder/disease is not always possible. As a result, most if not all mental conditions, function as both symptoms as well as disorders. There are often functional descriptions provided for psychological disorders and these are vulnerable to circular reasoning due to the etiological fuzziness inherent of these diagnostic categories.

In this thesis we have taken two very common diseases named as Diabetes and Blood pressure which is very common in people and their symptoms are also very common. So first of all we know little bit about these diseases and then later on we find

how serious is patient who is suffering from these two diseases using data clustering and fuzzy max-min composition.

6.2 Diabetes

Diabetes means your blood glucose, also called blood sugar, is too high. Your blood always has some glucose in it because your body needs glucose for energy to keep you going. But too much glucose in the blood isn't good for your health. Glucose comes from the food you eat and is also made in your liver and muscles. Your blood carries the glucose to all the cells in your body. Insulin is a chemical, also called a hormone, made by the pancreas. The pancreas releases insulin into the blood. Insulin helps the glucose from food get into your cells. If your body doesn't make enough insulin, or if the insulin doesn't work the way it should, glucose can't get into your cells. It stays in your blood instead. Your blood glucose level then gets too high, causing pre-diabetes or diabetes. Pre-diabetes is a condition in which blood glucose levels are higher than normal but not high enough for a diagnosis of diabetes. People with pre-diabetes are at increased risk for developing type 2 diabetes and for heart disease and stroke. The good news is, if you have pre-diabetes, you can reduce your risk of getting diabetes. With modest weight loss and moderate physical activity, you can delay or prevent type 2 diabetes and even return to normal glucose levels. People can get diabetes at any age. Type 1, type 2, and gestational diabetes are the three main kinds. Type 1 diabetes, formerly called juvenile diabetes or insulin-dependent diabetes, is usually first diagnosed in children, teenagers, or young adults. With this form of diabetes, the beta cells of the pancreas no longer make insulin because the body's immune system has attacked and destroyed them. Treatment for type 1 diabetes includes taking insulin and possibly another inject able medicine, making wise food choices, being physically active, taking aspirin daily—for some— and controlling blood pressure and cholesterol Type 2 diabetes, formerly called adult-onset diabetes or noninsulin-dependent diabetes, is the most common form of diabetes. People can develop type 2 diabetes at any age—even during childhood. This form of diabetes usually begins with insulin resistance, a condition in which fat, muscle, and liver cells do not use insulin properly. At first, the pancreas keeps up with the added demand by producing more insulin. In time, however, it loses the ability to secrete enough insulin in response to meals. Being overweight and inactive increases the chances of developing

type 2 diabetes. Treatment includes using diabetes medicines, making wise food choices, being physically active, taking aspirin daily—for some—and controlling blood pressure and cholesterol. Some women develop gestational diabetes during the late stages of pregnancy. Although this form of diabetes usually goes away after the baby is born, a woman who has had it is more likely to develop type 2 diabetes later in life. Gestational diabetes is caused by the hormones of pregnancy or a shortage of insulin. After many years, diabetes can lead to serious problems with your eyes, kidneys, nerves, and gums and teeth. But the most serious problem caused by diabetes is heart disease. When you have diabetes, you are more than twice as likely as people without diabetes to have heart disease or a stroke. If you have diabetes, your risk of a heart attack is the same as someone who has already had a heart attack. Both women and men with diabetes are at risk. You may not even have the typical signs of a heart attack. You can reduce your risk of developing heart disease by controlling your blood pressure and blood fat levels. If you smoke, talk with your doctor about quitting. Remember that every step toward your goals helps! See “Why Taking Care of Your Diabetes Is Important” to learn how you can try to prevent or delay long-term problems. The best way to take care of your health is to work with your health care team to keep your blood glucose, blood pressure, and cholesterol in your target range. Everyone’s blood has some glucose in it. In people who don’t have diabetes, the normal range is about 70 to 120. Blood glucose goes up after eating but 1 or 2 hours later returns to the normal range.

6.3 Hypoglycemia

Hypoglycemia also called low blood sugar, occurs when your blood glucose (blood sugar) level drops too low to provide enough energy for your body's activities. In adults or children older than 10 years, hypoglycemia is uncommon except as a side effect of diabetes treatment, but it can result from other medications or diseases, hormone or enzyme deficiencies, or tumors. Glucose, a form of sugar, is an important fuel for your body. Carbohydrates are the main dietary sources of glucose. Rice, potatoes, bread, tortillas, cereal, milk, fruit, and sweets are all carbohydrate-rich foods. After a meal, glucose molecules are absorbed into your bloodstream and carried to the cells, where they are used for energy. Insulin, a hormone produced by your pancreas, helps glucose enter cells. If you take in more glucose than your body needs at the time, your

body stores the extra glucose in your liver and muscles in a form called glycogen. Your body can use the stored glucose whenever it is needed for energy between meals. Extra glucose can also be converted to fat and stored in fat cells. When blood glucose begins to fall, glucagon, another hormone produced by the pancreas, signals the liver to break down glycogen and release glucose, causing blood glucose levels to rise toward a normal level. If you have diabetes, this glucagon response to hypoglycemia may be impaired, making it harder for your glucose levels to return to the normal range.

6.4 Hyperglycemia

Hyperglycemia, or high blood sugar, is a condition in which an excessive amount of glucose circulates in the blood plasma. This is generally a glucose level higher than 10 mmol/l (180 mg/dl), but symptoms may not start to become noticeable until even higher values such as 15-20 mmol/l (270-360 mg/dl). However, chronic levels exceeding 7 mmol/l (125 mg/dl) can produce organ damage. High blood sugar levels happen when the body either can't make insulin (type 1 diabetes) or can't respond to insulin properly (type 2 diabetes). The body needs insulin so glucose in the blood can enter the cells of the body where it can be used for energy. In people who have developed diabetes, glucose builds up in the blood, resulting in hyperglycemia. Having too much sugar in the blood for long periods of time can cause serious health problems if it's not treated. Hyperglycemia can damage the vessels that supply blood to vital organs, which can increase the risk of heart disease and stroke, kidney disease, vision problems, and nerve problems in people with diabetes. These problems don't usually show up in kids or teens with diabetes who have had the disease for only a few years. However, these health problems can occur in adulthood in some people with diabetes, particularly if they haven't managed or controlled their diabetes properly. Blood sugar levels are considered high when they're above your target range. Your diabetes health care team will let you know what your target blood sugar levels are.

6.5 Diabetes Management Plan

Intensive diabetes management--keeping your blood glucose as close to the normal range as possible to prevent long-term complications--can increase the risk of hypoglycemia. If your goal is tight control, talk to your health care team about ways to prevent hypoglycemia and how best to treat it if it does occur.

Table 2: Blood Glucose levels

Normal and target blood glucose ranges (mg/dL)	
Normal blood glucose levels in people who do not have diabetes	
Upon waking (fasting)	70 to 110
After meals	70 to 140
Target blood glucose levels in people who have diabetes	
Before meals	90 to 130
1 to 2 hours after the start of a meal	less than 180
Hypoglycemia (low blood glucose)	70 or below

6.6 Symptoms of Diabetes

1. Frequent Urination
2. Excessive Thirst
3. Hunger
4. Weight Loss
5. Fatigue
6. Blurry Vision
7. Old aches and Pains
8. Dry mouth
9. Dry or itchy skin
10. Excessive or usual infection
11. Numbness
12. Slow-healing wounds
13. Excessive eating

14. Altered mental status

15. Vaginal yeast infection (in a female)

16. Impotence (in a male).

6.7 Various Reasons for Which Diabetes Occur

1. Genetic Causes

2. Environmental factors

3. Due to Over-weight

4. High-alcohol Intakes

5. Sedentary life-styles

6. Increasing Age

7. Pregnancy in women

8. Hypertension

9. Serum lipids and lipoproteins.

6.8 Blood Pressure

Blood pressure (BP) is the pressure exerted by circulating blood upon the walls of blood vessels, and is one of the principal vital signs. During each heartbeat, BP varies between a maximum (systolic) and a minimum (diastolic) pressure. The mean BP, due to pumping by the heart and resistance to flow in blood vessels, decreases as the circulating blood moves away from the heart through arteries. Blood pressure drops most rapidly along the small arteries and arterioles, and continues to decrease as the blood moves through the capillaries and back to the heart through veins. Gravity, valves in veins, and pumping from contraction of skeletal muscles, are some other influences on BP at various places in the body. The term *blood pressure* usually refers to the pressure measured at a person's upper arm. It is measured on the inside of an elbow at the brachial artery, which is the upper arm's major blood vessel that carries blood away from the heart. A person's BP is usually expressed in terms of the systolic pressure over diastolic pressure (mmHg), for example 140/90. The following US classification of blood pressure applies to adults

aging 18 and older. It is based on the average of seated BP readings that were properly measured during 2 or more office visits. In the UK, hypertension is considered when a patient's reading is above 140/90 mmHg. According to the American Heart Association the following are the blood pressure categories

Table 3: Classification of blood pressure for adults

Classification of blood pressure for adults		
Category	**systolic, mmHg**	**diastolic, mmHg**
Hypotension	< 90	< 60
Normal	**90 - 119**	**60 - 79**
Prehypertension	120 – 139	or 80 – 89
Stage 1 Hypertension	140 – 159	or 90 – 99
Stage 2 Hypertension	160 - 179	or 100 - 109
Hypertensive Crisis	≥ 180	or ≥ 110

While average values for arterial pressure could be computed for any given population, there is often a large variation from person to person; arterial pressure also varies in individuals from moment to moment. Additionally, the average of any given population may have a questionable correlation with its general health, thus the relevance of such average values is equally questionable. However, in a study of 100 subjects with no known history of hypertension, an average blood pressure of 112/64 mmHg was found, which the normal values are. Various factors influence a person's average BP and variations. Factors such as age and gender influence average values. In children, the normal ranges are lower than for adults and depend on height. As adults age, systolic

pressure tends to rise and diastolic tends to fall. In the elderly, BP tends to be above the normal adult range, largely because of reduced flexibility of the arteries. Also, an individual's BP varies with exercise, emotional reactions, sleep, digestion and time of day. Differences between left and right arm BP measurements tend to be random and average to nearly zero if enough measurements are taken. However, in a small percentage of cases there is a consistently present difference greater than 10 mmHg which may need further investigation, e.g. for obstructive arterial disease. The risk of cardiovascular disease increases progressively above 115/75 mmHg. In the past, hypertension was only diagnosed if secondary signs of high arterial pressure were present, along with a prolonged high systolic pressure reading over several visits. Regarding hypotension, in practice blood pressure is considered too low only if noticeable symptoms are present. Clinical trials demonstrate that people who maintain arterial pressures at the low end of these pressure ranges have much better long term cardiovascular health. The principal medical debate concerns the aggressiveness and relative value of methods used to lower pressures into this range for those who do not maintain such pressure on their own. Elevations, more commonly seen in older people, though often considered normal, are associated with increased morbidity and mortality.

Table 4: Average blood pressure ranges (mm/Hg)

1 year	6 to 9 years	Adults
95/65	100/65	110/65 to 140/90

6.9 Mean Arterial Pressure

The mean arterial pressure (MAP) is the average over a cardiac cycle and is determined by the cardiac output (CO), systemic vascular resistance (SVR), and central venous pressure (CVP), MAP = (CO – SVR) + CVP (63)

MAP can be approximately determined from measurements of the systolic pressure P_{sys} and the diastolic pressure P_{dias} while there is a normal resting heart rate,

$$MAP = P_{dias} + \frac{1}{3}\left(P_{sys} - P_{dias}\right) \quad (64)$$

6.10 Pulse Pressure

The up and down fluctuation of the arterial pressure results from the pulsatile nature of the cardiac output, i.e. the heartbeat. The pulse pressure is determined by the interaction of the stroke volume of the heart, compliance (ability to expand) of the aorta and resistance to flow in the arterial tree. By expanding under pressure, the aorta absorbs some of the force of the blood surge from the heart during a heartbeat. In this way the pulse pressure is reduced from what it would be if the aorta wasn't compliant. The loss of arterial compliance that occurs with aging explains the elevated pulse pressure found in elderly patients. The pulse pressure can be simply calculated from the difference of the measured systolic and diastolic pressures, $P_{pulse} = P_{sys} - P_{dias}$ (65)

6.11 Vascular pressure wave

Modern physiology developed the concept of the vascular pressure wave (VPW).This wave is created by the heart during the systole and originates in the ascending aorta. Much faster than the stream of blood itself, it is then transported through the vessel walls to the peripheral arteries. There the pressure wave can be palpated as the peripheral pulse. As the wave is reflected at the peripheral veins it runs back in a centripetal fashion. Where the crests of the reflected and the original wave meet, the pressure inside the vessel is higher than the true pressure in the aorta. This concept explains why the arterial pressure inside the peripheral arteries of the legs and arms is higher than the arterial pressure in the aorta, and in turn for the higher pressures seen at the ankle compared to the arm with normal ankle brachial pressure index values.

6.12 Measurement

Arterial pressure is most commonly measured via a sphygmomanometer, which historically used the height of a column of mercury to reflect the circulating pressure. BP values are generally reported in millimeters of mercury (mmHg), though aneroid and electronic devices do not use mercury. For each heartbeat, BP varies between systolic and diastolic pressures. Systolic pressure is peak pressure in the arteries, which occurs near the end of the cardiac cycle when the ventricles are contracting. Diastolic pressure is minimum pressure in the arteries, which occurs near the beginning of the cardiac cycle when the ventricles are filled with blood. An example of normal measured values for a

resting, healthy adult human is 120 mmHg systolic and 80 mmHg diastolic (written as 120/80 mmHg, and spoken as "one-twenty over eighty"). Systolic and diastolic arterial BPs are not static but undergo natural variations from one heartbeat to another and throughout the day (in a circadian rhythm). They also change in response to stress, nutritional factors, drugs, disease, exercise, and momentarily from standing up. Sometimes the variations are large. Hypertension refers to arterial pressure being abnormally high, as opposed to hypotension, when it is abnormally low. Along with temperature, respiratory rate, and pulse rate, BP is one of the four main vital signs routinely monitored by medical professionals and healthcare providers. Measuring pressure invasively, by penetrating the arterial wall to take the measurement, is much less common and usually restricted to a hospital setting.

6.13 Symptoms of Blood Pressure

1. Stressed
2. Sedentary
3. Bloated
4. Weakness
5. Failing
6. Tiredness
7. Dizzy
8. Fainting
9. Coma.
10. Headache
11. Blurred Vision
12. Nausea and Vomiting
13. Chest pain and Shortness of Breadth
14. Fatigue
15. Depression

16. Thirst.

6.14 Various Reasons for Which Blood pressure Occur

1. Moderate or severe bleeding.

2. Severe inflammation of organs inside the body.

3. A slow heart rate. (Bradycardia).

4. Genetic Causes.

5. Abnormally fast heart rate (Tachycardia)

6. Alcohol and nancotics also can cause blood pressure.

7. Postural (orthostatic) hypotension.

8. Micturition syncope.

9. Septicemia.

6.15 Infrencing

In this thesis we mainly focus the clustering and Fuzzy-Infrencing and by doing this I strongly correlate the seriousness of the patients by taking some common symptoms of these two diseases. In this first we have raw data of both the disease in which we have to filter out the irrelevant data and for this we used data clustering. There are various clustering techniques: Traditional and AI based which we already discussed and after clustering we have correlated these two diseases by using fuzzy composition i.e. max-min and Cartesian product. Let us take R and S matrix of 4×4 in which Diabetes and Blood pressure data due to common symptoms are stored. Now here we take a fractional value by dividing from the highest value from the data of diabetes and blood pressure.

6.16 Common symptoms for Diabetes and Blood Pressure are

1. Fatigue $\rightarrow S_1$

2. Blurred Vision $\rightarrow S_2$

3. Old Aches and Pains $\rightarrow S_3$

4. Depression $\rightarrow S_4$

6.17 Causes for the Diabetes

1. Genetic Causes $\rightarrow x_1$

2. Overweight $\rightarrow x_2$

3. Increasing in Age $\rightarrow x_3$

4. Pregnancy in woman $\rightarrow x_4$

Now R matrix is:

$$R = \begin{array}{c} \\ x_1 \\ x_2 \\ x_3 \\ x_4 \end{array} \begin{pmatrix} \overset{S_1}{0.5} & \overset{S_2}{0.4} & \overset{S_3}{0.2} & \overset{S_4}{0.1} \\ 0.7 & 0.3 & 0.8 & 0.2 \\ 0.9 & 0.7 & 0.6 & 0.8 \\ 0.8 & 0.3 & 0.4 & 0.5 \end{pmatrix}$$

6.18 Causes for the Blood Pressure

1. Genetic Causes

2. Slow heart Rate

3. Fast heart rate

4. High Alcohol Intake.

Now matrix S is:

$$S = \begin{array}{c} \\ S_1 \\ S_2 \\ S_3 \\ S_4 \end{array} \begin{pmatrix} \overset{Y_1}{0.4} & \overset{Y_2}{0.8} & \overset{Y_3}{0.2} & \overset{Y_4}{0.4} \\ 0.5 & 0.6 & 0.5 & 0.3 \\ 0.6 & 0.7 & 0.8 & 0.4 \\ 0.3 & 0.6 & 0.9 & 0.3 \end{pmatrix}$$

Summary

In this chapter we mainly diagnosis disease by using data clustering technique using fuzzy inferencing. The two types of disease i.e. diabetes and blood pressure and there measurement and ranges are discussed due to various reasons and symptoms due to

which we developed the matrix in which the data of both the disease due to common symptoms are entered. At last we have to find the fuzzy max-min and Cartesian product of these two matrixes and by knowing this we concluded the seriousness of patient.

CHAPTER 7
RESULTS AND DISCUSSIONS

7.1 Diabetes Clustering

In this graph we show how the data of diabetes forms a cluster in various regions. Its aim is to merge a cluster which exhibit high “similarity” (low “dissimilarity”). Specifically, the cluster pair that, according to a preselected dissimilarity measure between sets (cluster), exhibits the lowest dissimilarity is determined. We simply load the data by changing the data into .dat file and then load the data. Here the X and Y axis represent the data1 and data2 of the diabetes. The program of this graph is shown in appendix A.

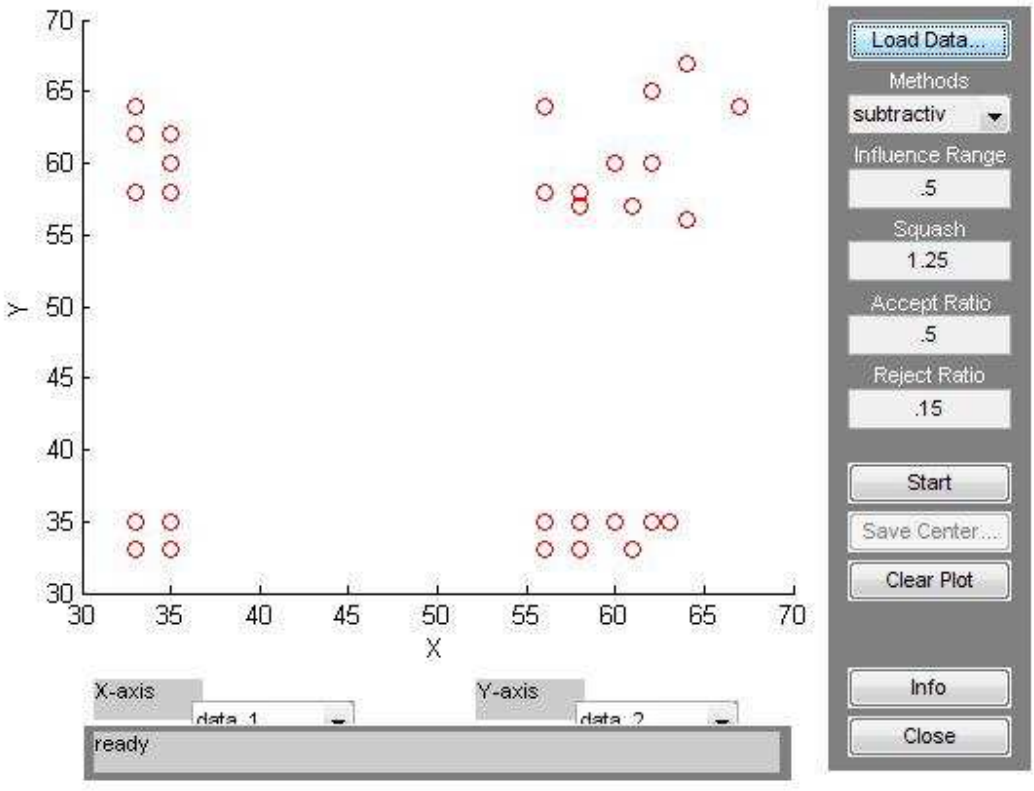

Figure 7.1: Diabetics Clustering.

7.2 Diabetics K-Means

This comes under Hard Clustering Algorithm and in this each data vector belongs exclusively to a single cluster. It is suitable for unraveling compact cluster and is a fast iterative algorithm as it requires only a few iterations to converge and the computations required at each iteration are not complicated. Thus it poses as a candidate for processing large data sets. We simply load the data of diabetes through .dat file. The program written in matlab is shown in appendix B.

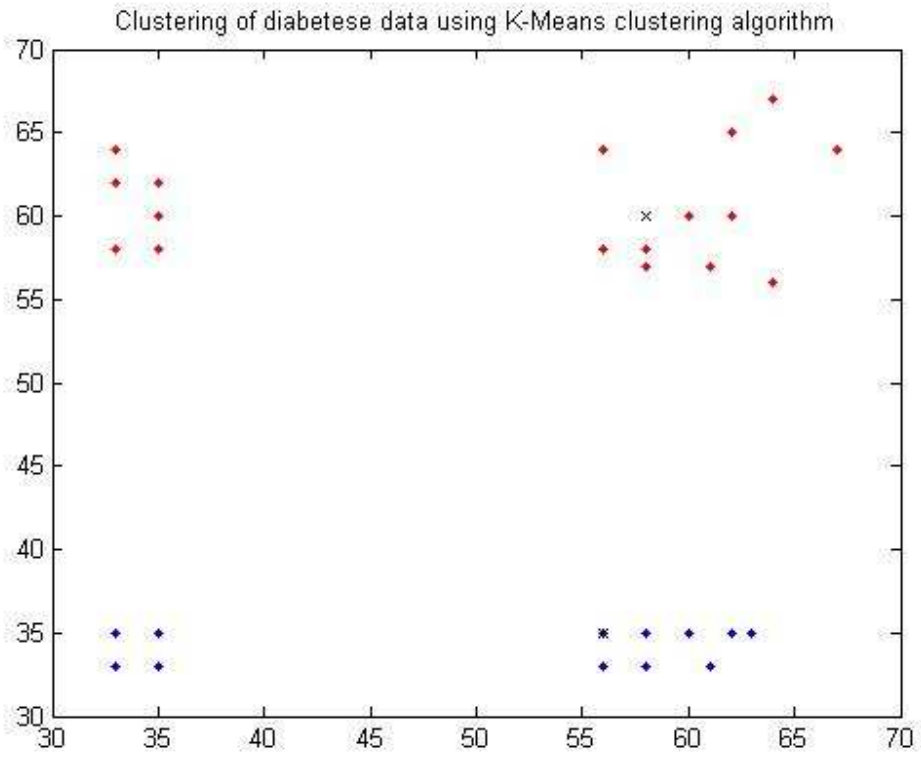

Figure 7.2: K-Means Clustering Algorithm of Diabetic data

7.3 Diabetics Dendrograms

This is so called proximity dendrograms as in this dissimilarity (similarity) a distance measure between clusters has been adopted. This has a tree like structure, which shows the dissimilarity dendrograms of the clustering hierarchy after applying the single link algorithm. The proximity dendrogams is a useful tool in visualizing information

concerning a clustering hierarchy. Its usefulness becomes more apparent in cases where the number of data points is large. The program written in matlab is shown in appendix C.

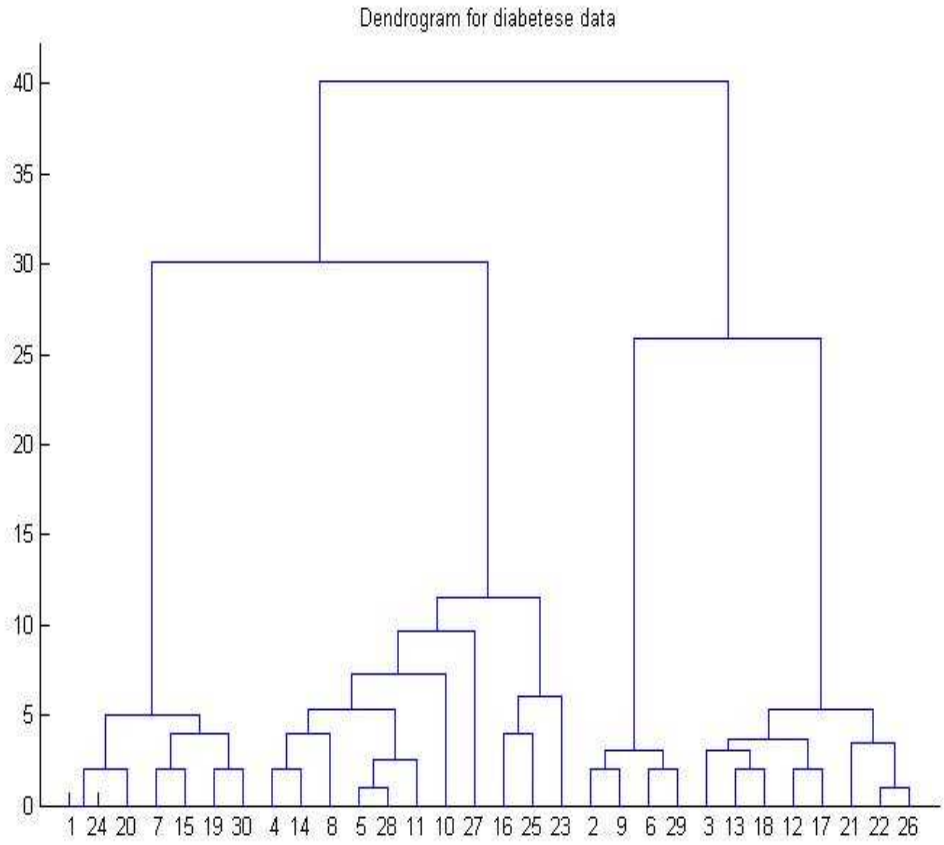

Figure 7.3: Dendrograms for Diabetes data.

7.4 Fuzzy C-Means

This comes under Nonhard Clustering Algorithm. In this each data vector may belong to (or may be compatible with) more than one cluster up to a certain number. The aim of FCM is to move each of the available parameter vectors towards region in the data space that is dense in data points. In this the grade of memberships of the data vectors in cluster is computed, taking into account the (squared Euclidean) distances. We simply load the data through .dat file. The program written in matlab is shown in appendix D.

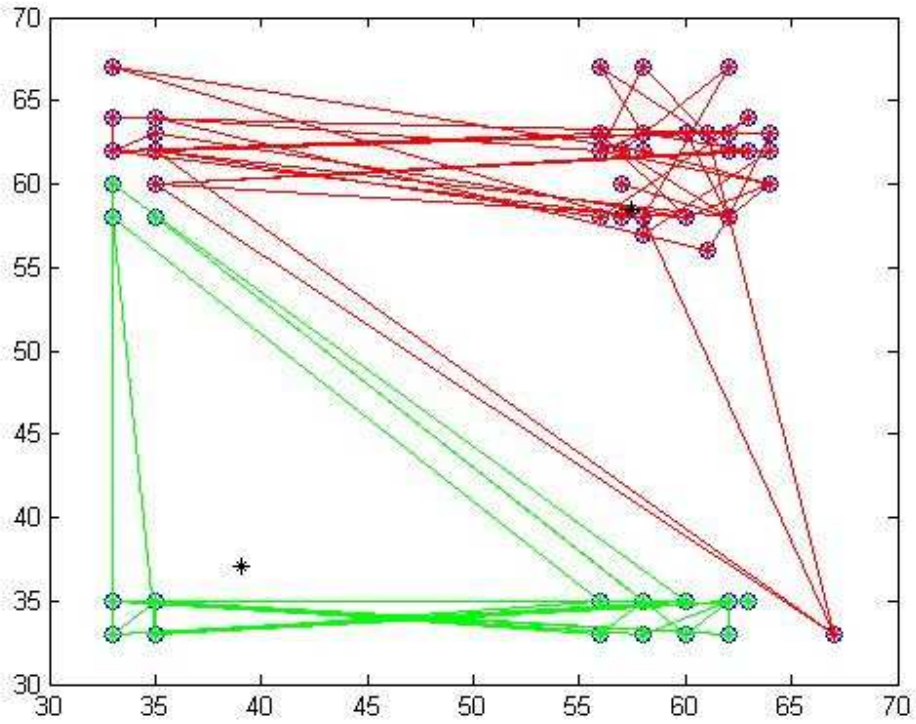

Figure 7.4: Diabetes Fuzzy C-Means.

7.5 Diabetes Error

In this plot we are calculating the SSE of each of the cluster. Sum of square error is the error between all the data points with their centroid, from this diagram we observe the density of the cluster. The square error shoule be minimum.

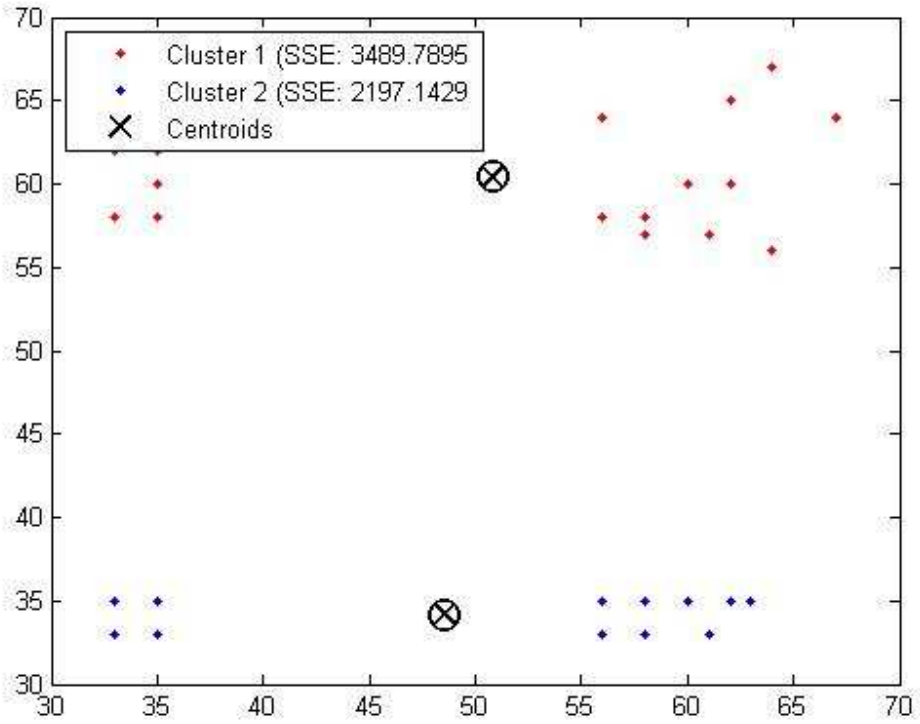

Figure 7.5: Clustering with sum of square error

7.6 Blood Pressure Clustering

In this graph we show how the data of blood pressure forms a cluster in various regions. Its aim is to merge a cluster which exhibit high "similarity" (low "dissimilarity"). Specifically, the cluster pair that, according to a preselected dissimilarity measure between sets (cluster), exhibit the lowest dissimilarity is determined. We simply load the data by changing the data into .dat file and then load the data. Here the X and Y axis represent the data1 and data2 of the diabetes. The program of this graph is shown in appendix F.

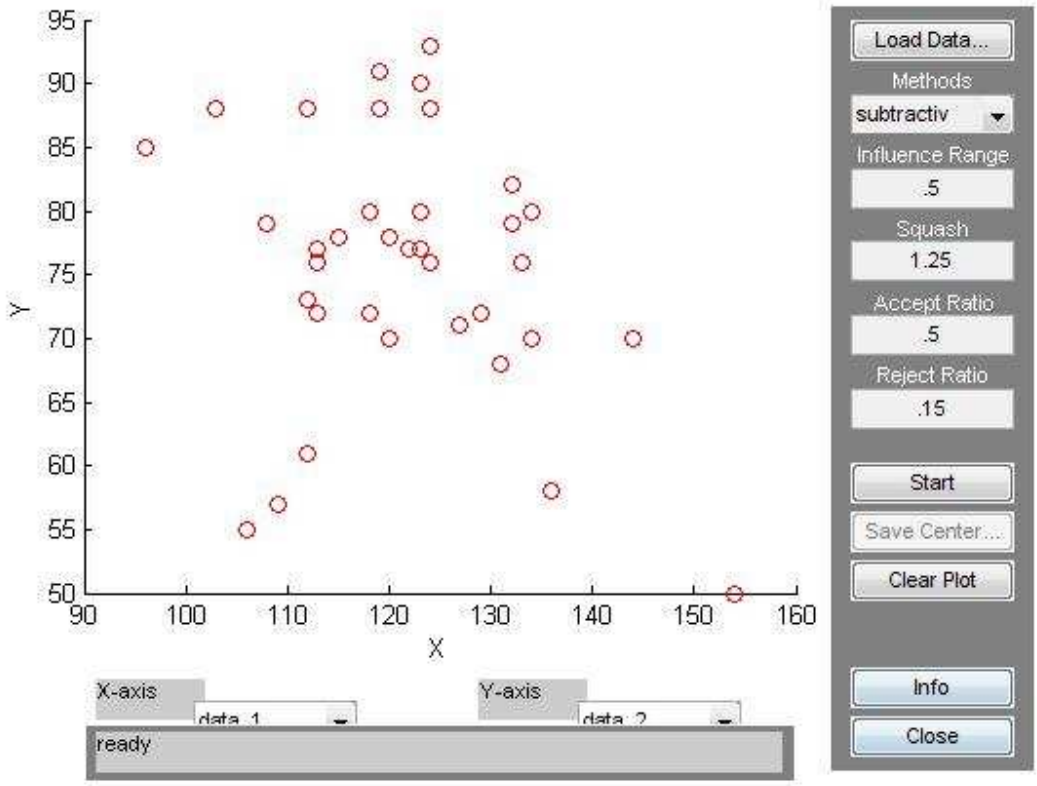

Figure 7.6: Blood Pressure Clustering.

7.7 Blood Pressure K-Means

This comes under Hard Clustering Algorithm and in this each data vector belongs exclusively to a single cluster. It is suitable for unraveling compact cluster and is a fast iterative algorithm as it requires only a few iterations to converge and the computations required at each iteration are not complicated. Thus it poses as a candidate for processing large data sets. We simply load the data of diabetes through .dat file. The program written in matlab is shown in appendix G.

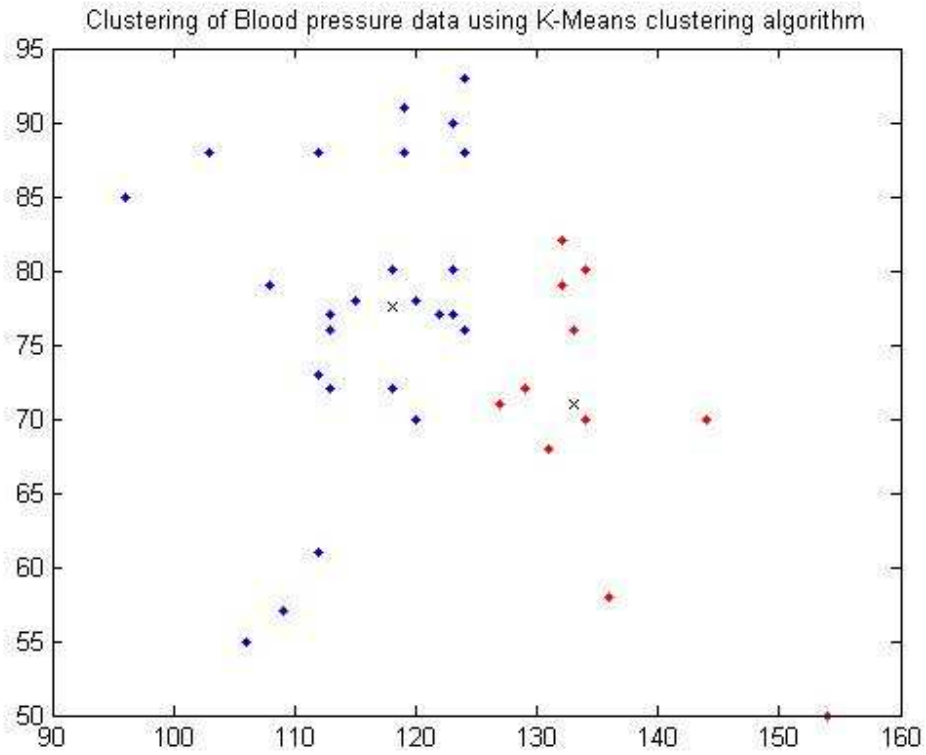

Figure 7.7: K-Means Algorithm for Blood Pressure.

7.8 Blood Pressure Dendrograms:

This is so called proximity dendrograms as in this dissimilarity (similarity) a distance measure between clusters has been adopted. This has a tree like structure, which shows the dissimilarity dendrograms of the clustering hierarchy after applying the single link algorithm. The proximity dendrogams is a useful tool in visualizing information concerning a clustering hierarchy. Its usefulness becomes more apparent in cases where the number of data points is large. The program written in matlab is shown in appendix H.

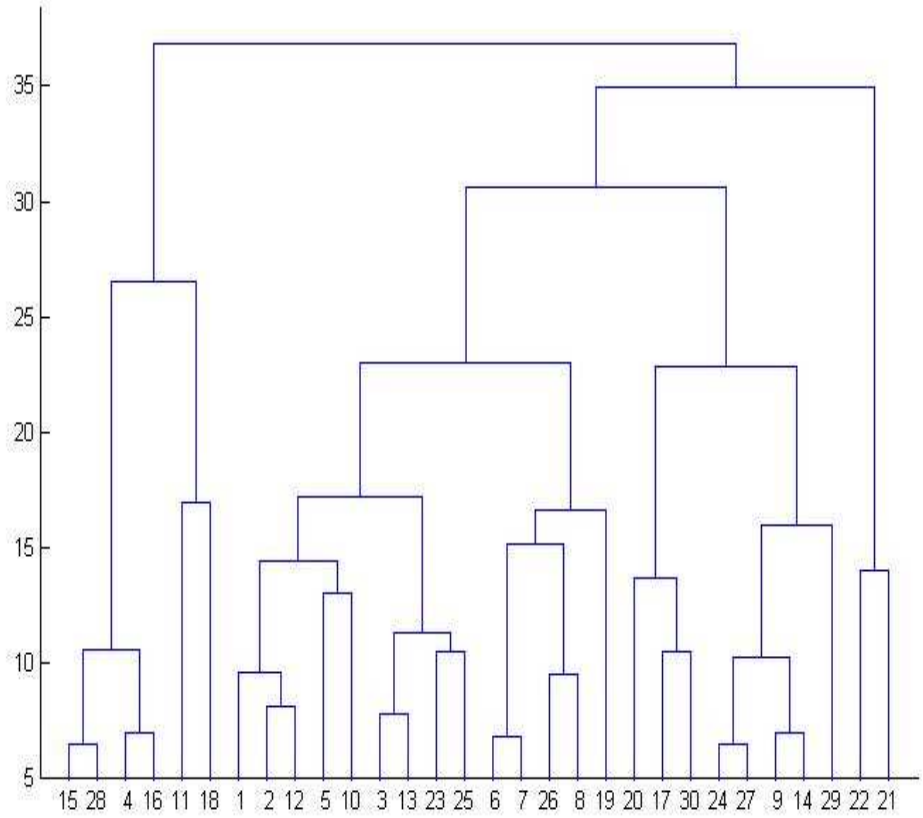

Figure 7.8: Blood Pressure Dendrograms.

7.9 Fuzzy C-Means

This comes under Nonhard Clustering Algorithm. In this each data vector may belong to (or may be compatible with) more than one cluster up to a certain number. The aim of FCM is to move each of the available parameter vectors towards region in the data space that is dense in data points. In this the grade of memberships of the data vectors in cluster is computed, taking into account the (squared Euclidean) distances. We simply load the data through .dat file. The program written in matlab is shown in appendix I.

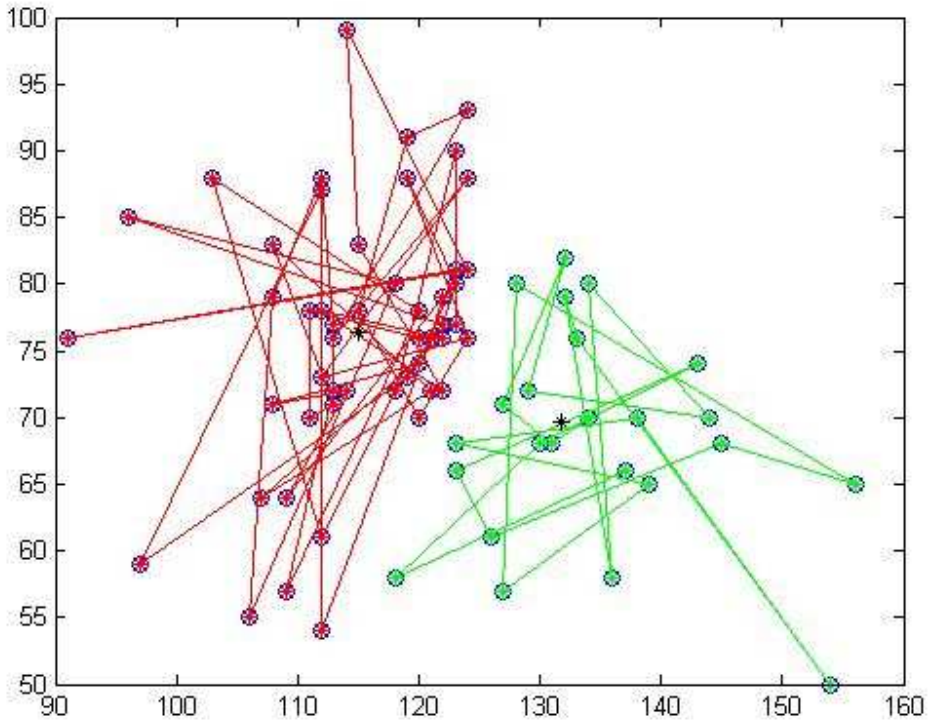

Figure 7.9: Blood Pressure Fuzzy C-Means

7.10 Blood Pressure Error

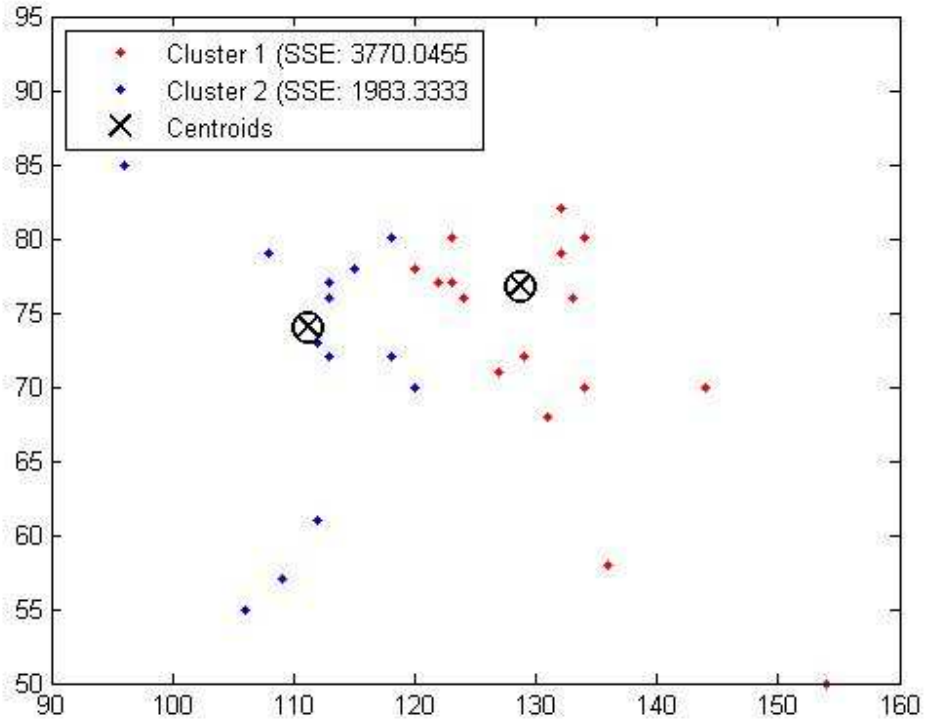

Figure 7.10: Clustering blood pressure data and sum of square error

In this plot we are calculating the SSE of each of the cluster. Sum of square error is the error between all the data points with their centroid, from this diagram we observe the density of the cluster. The square error shoule be minimum.

7.11 Fuzzy Max-Min

The t matrix given as:

t =

0.2000	0.4000	0.2000	0.2000
0.4800	0.5600	0.6400	0.3200
0.3600	0.7200	0.7200	0.3600
0.3200	0.6400	0.4500	0.3200

T matrix gives the data of the patient seriousness due to the two diseases i.e. Diabetes and Blood pressure occur due to the common symptoms. So here we conclude

that due to x_1 and y_1 the patient seriousness is 20%, due to x_1 and y_2 the patient seriousness is 40%.....and so on. We simply draw a conclusion from this t matrix is that how we know the patient seriousness due to various reasons for which disease occur and by knowing this a necessary measures should be taken. The program is written in matlab and is shown in appendix K.

7.12 Fuzzy Cartesian Product

The k matrix given as:

k =

0.2000	0.4000	0.2000	0.2000
0.4800	0.5600	0.6400	0.3200
0.3600	0.7200	0.7200	0.3600
0.3200	0.6400	0.4500	0.3200

k matrix gives the data of the patient maximum seriousness due to the two diseases i.e. Diabetes and Blood pressure occur due to the common symptoms. In this matrix we draw a conclusion that the maximum seriousness of the patient due to the x_1 and y_1 is 20 % and due to x_1 and y_2 is 40% and so on. We simply draw a conclusion that it is the maximum seriousness of the patient due to various reasons for which disease occur and by knowing this a necessary measures should be taken. The program is written in matlab and is shown in appendix L.

CHAPTER 8

CONCLUSION AND FUTURE SCOPE

This thesis gives a novel idea of disease diagnosis system using data clustering and fuzzy max-min inferencing system. The clinical data collected from continuous monitoring of the patient can be clustered to group similar kind of data and from these similar kinds of data a inferencing can be drawn based on the symptoms and disease. The inferencing mechanism used in this thesis is fuzzy based inference mechanism. Fuzzy logic is the kind of logic which depicts human intelligence and hence the fuzzy based inference is a superior choice from ant other inference mechanism.

Data clustering is a main part because it reduces the huge dimension of data, groups similar kind of data and help to improve the erroneous, time series data. There are different contemporary as well as intelligent data clustering algorithm like K Means, hard C Means, Fuzzy C Means, Gustafson-Kessel clustering method. All clustering methods work on the distance formula. There are different distance formulas available in the literature, but two of the distance functions are used. Those are Euclidian distance and Manhattan distance or taxi cab distance. This thesis takes the medical data (blood pressure and diabetes) from the UCI respiratory machine learning database, University of California, Irvine. The data is clustered using K Means and Fuzzy C-Means algorithm. After clustering the data, fuzzy based inferencing mechanism is implemented in the data to get the desired diagnosis results.

There is a lot of future scope of this research. Different soft computing based approach like self organizing map (SOM), linear vector quantization (LVQ) and associative resonance theory (ART) can be used to effectively cluster the data. To optimize the shape of the resulting cluster and validate the clusters different optimization and genetic programming can be used. This thesis takes the data of only two kind of disease. In future more disease can be added so that the complexity of the system can be increase many fold.

REFERENCES

[1] Scott C Newton, Surya Pemmaraju and Sunanda Mitra, "Adaptive Fuzzy Leader Clustering of Complex Data Sets in Pattern Recognition," IEEE Transactions on Neural Network, vol. 3, no. 5, 1992, pp. 794-800

[2] Y M Sebzalli and X Z Wang, "Knowledge Discovery From Process Operational Data Using PCA and Fuzzy Clustering," Engineering Applications of Artificial Intelligence, vol. 14, 2001, pp. 607-616

[3] Timo Ahvenlampi and Urpo Kortela, "Clustering Algorithm in Process Monitoring and Control Application to Continuous Digester," Informatica, vol. 29, 2005, pp. 101-109

[5] Skrjanc I., "Fuzzy Model Based Detection of Sensor Faults in Waste Water Treatment Plant," in Proceedings of 5th WSEAS International Conference on Computational Intelligence, Man-Machine Systems and Cybernetics, 2006, pp. 195-199

[6] Johnnie W. Huang and Rob J. Roy, "Multiple-Drug Hemodynamic Control Using Fuzzy Decision Theory," IEEE Transactions on Biomedical Engineering, vol.45, no.2, February 1998.

[7] Mrutyunjaya Panda and Manas Ranjan Patra, "Ensemble Voting System for Anamoly Based Network Intrusion Detection," International Journal of Recent Trends in Engineering, vol. 2, no. 5, Nov 2009, pp. 8-13

[8] Chris Fraley and Adriane E. Raftery, "How Many Clusters? Which Clustering method? Answer Via Model-based cluster Analysis", The Computer Journal, vol.41, No.8, 1998.

[9] A.K. Jain, M.N. Murty, P.J.Flynn, "Data Clustering: A Review", ACM Computing Surveys,vol.31, no.3, September 1999

[10] G.Zahlmonn, M.Stherf, A.Wegner, M.Obermoier, M.Mertz, "Situation Assessment of Glaucoma using a hybrid Fuzzy Neural Network", IEEE Engineering in Medicine and Biology, 0739-5175, January/febuary 2000.

[11] Sudeep Sarkar, Member, IEEE, and Padmanabhan Soundararajan, "Supervised Learning of Large Perceptual organisation: Graph Spectral Partitioning and Learning Automata", IEEE Transaction on pattern analysis and Machine

Intelligence, vol.22,no.5, May 2000.

[12] R. I. John and P. R. Innocent, "Modelling Uncertainty in Clinical Diagnosis using Fuzzy logic", "Modelling Uncertainty in Clinical Diagnosis using fuzzy logic",IEEE Transactions on systems ,man and cybernetics-part B cybernetics vol.35,no.6 December 2005.

[13] John W. Sheppard and Stephyn G. W. Butcher, Mark A. Kaufman and Craig MacDougall, "Not-So-Naïve Bayesian Networks and Unique Identification in Developing Advanced Diagnostics", 0-7803-9546-8/06/$20.00 © 2006 IEEE Paper 1572.

[14] Andrew Hamilton-Wright, Daniel W. Stashuk, "Transparent Decision Support Using Statistical Reasoning and Fuzzy Inference", IEEE Transactions on knowledge and Data Engineering vol.18 no.8 August 2006.

[15] C Lionberger and M Cromaz, "Control of Acquisition and Cluster Based Online Processing of Gretina Data," Proceedings of ICALEPCS 07, 2007, pp. 93-95

[16] Zhe Song and Andrew Kusiak, "Constraint Based Control of Boiler Efficiency: A Data Mining Approach," IEEE Transactions on Industrial Informatics, vol. 3, no. 1, Feb 2007, pp. 73-83

[17] Juan E. Moreno, Oscar Castillo, Juan R. Castro, Luis G. Martínez, Patricia Melin, "Data Mining for extraction of fuzzy IF-THEN rules using Mamdani and Takagi-Sugeno-Kang FIS", Engineering Letters, 15:1, EL_15_1_13 15 August 2007.

[18] Mei-Hui Wang and Chang-Shing Lee, Huan-Chung Li and Wei-Min Ko, "Ontology-based Fuzzy Inference Agent for Diabetes Classification", 1-4244-1214-5/07/$25.00 ©B2007 IEEE

[19] Markos G. Tsipouras, Costas Voglis, and Dimitrios I. Fotiadis. "A Framework for Fuzzy Expert System Creation—Application to Cardiovascular Diseases", IEEE Transaction on Biomedical Engineering, vol.54, no.11, November 2007.

[20] Markos G. Tsipouras, Dimitrios I. Fotiadis, Themis P. Exarchos, Anna P. Kotsia, Konstantinos V. Vakalis, Katerina K. Naka, and Lampros K. Michalis, "Automated Diagnosis of Coronary Artery Disease Based on Data Mining and Fuzzy Modeling",IEEE Transactions on Information Technology in Biomedicine, vol.12,no.4,july 2008

[21] Dustin Dunsmuir1, Jeremy Daniels, Christopher Brouse, Simon Ford, J. Mark Ansermino, “A Knowledge Authoring Tool For Clinical Decision Support”, Journal of Clinical Monitoring and Computing (2008).

[22] Kemal Polat , Salih Gu¨nes, Ahmet Arslan, “A cascade learning system for classification of diabetes disease: Generalized Discriminant Analysis and Least Square Support Vector Machine”, Expert Systems with Applications 34 (2008) 482–487.

[23] Osama Abu Abbas, “Comparisons Between Data Clustering Algorithms,” The International Arab Journal of Information Technology, vol. 5, no. 3, 2008, pp. 320-325

[24] K Premalatha and A M Natarajan, “A New Approach for Data Clustering Based on PSO with Local Search,” Computer and Information Science, vol. 1, no. 4, 2008, pp. 139-145.

[25] Voula C. Georgopoulos, Chrysotomos D. Stylios, “Diagnosis Support using Fuzzy Cognitive Maps combined with Genetic Algorithms”, 31st Annual International Conference of the IEEE EMBS Minneapolis, Minnesota, USA, September 2-6, 2009.

[26] V Kavitha and M Punithavalli, “Clustering Time Series Data Stream- A Literature Review,” International Journal of Computer Science and Information Security, vol. 8, no. 1, 2010, pp. 289-294

[27] Izakian, H., Abraham, A., “Fuzzy C-Means and Fuzzy Swarm for Fuzzy Clustering Problem,” Expert Systems with Applications, 2010, doi: 10.1016/j.eswa.2010.07.112

[28] S Kalyani and k S Swarup, “Supervised Fuzzy C Means Clustering Techniques for Security Assessment and Classification of Power System,” International Journal of Engineering, Science and Technology, vol. 2, no. 3, 2010, pp. 175-185

[29] Xian-Xia Zhang et.al, “Spatially Constrained Fuzzy Clustering Based Sensor Placement for Spatiotemporal Fuzzy Control System,” IEEE Transaction on Fuzzy Systems, vol. 18, no. 5, 2010, pp. 946-957

[30] Mika Liukkonen et.al, “Analysis of Flue Gas Emission Data From Fluidized Bed Combustion Using Self-Organizing Map,” Applied Computational Intelligence

and Soft Computing, Hindawi Publishing Corporation, 2010, pp. 1-8

[31] Vasil Simeonov et.al, "Lake Water Monitoring Data Assessment By Multivariate Statistics," Journal of Water Resource and Protection, vol. 2, 2010, pp. 353-361

[32] Ibrahim Massod and Adnan Hassan, "Issues in Development of ANN-Based Control Chart Pattern Recognition Schemes," European Journal of Scientific Research, vol. 39, no. 3, 2010, pp. 336-355

[33] N Sujatha and K Iyakutty, "Refinement of Web Usage Data Clustering From K-Means with genetic Algorithm," European Journal of Scientific Research, vol. 42, no. 3, 2010, pp. 478-490

[34] Chang-Shing Lee, Mei-Hui Wang, "A Fuzzy Expert System for Diabetes Decision Support Application", IEEE Transactions on systems, man and cybernetics-part B: cybernetics, vol.41,no.1 february 2011.

[35] T. Padma, P. Balasubramanie, "Domain experts' knowledge-based intelligent decision support system in occupational shoulder and neck pain therapy", Applied Soft Computing 11 (2011) 1762–1769.

APPENDIX

Appendix-A [Matlab code for Diabetic Clustering]

% GUI based k-means for Diabetese data

findcluster('d:\Imp\thesis\test1.dat');

Appendix-B [Matlab Code for Diabetic K MEANS]

```
% K-means clustering of Diabetese data
clc
clear all
close all
X = load('E:\Imp\thesis\bp2.dat');
opts = statset('Display','final');
[cidx, ctrs] = kmeans(X, 2, 'Distance','city', 'Replicates',5,
'Options',opts);
plot(X(cidx==1,1),X(cidx==1,2),'r.', X(cidx==2,1),X(cidx==2,2),'b.',
ctrs(:,1),ctrs(:,2),'kx');
title('Clustering of Diabetese data using K-Means clustering
algorithm');
```

Appendix-C [Matlab Code for Diabetics Dendrograms]

```
% Dendrogram for Diabetics data:
X = load('E:\Imp\thesis\bp1.dat');
Y = pdist(X,'cityblock');
Z = linkage(Y,'average');
[H, T] = dendrogram(Z);
```

Appendix-D [Matlab Code for Fuzzy C MEANS]

```
% fuzzy C means for Diabetics data
data = load('E:\Imp\thesis\test2.dat');
[center,U,obj_fcn] = fcm(data,2);
plot(data(:,1), data(:,2),'o');
hold on;
maxU = max(U);
% Find the data points with highest grade of membership in cluster 1
index1 = find(U(1,:) == maxU);
% Find the data points with highest grade of membership in cluster 2
index2 = find(U(2,:) == maxU);
line(data(index1,1),data(index1,2),'marker','*','color','g');
line(data(index2,1),data(index2,2),'marker','*','color','r');
% Plot the cluster centers
plot([center([1 2],1)],[center([1 2],2)],'*','color','k')
hold off;
```

Appendix-E [Matlab Code for Diabetes SSE]

```
clc
close all
clear all
xy = [rand(100,2)*2+3; rand(100,2)];
% Visualize clouds
scatter(xy(:,1),xy(:,2))
close
% Cluster
[idx,c,sse] = kmeans(xy,2);
% Plot clouds with centroids and legend
plot(xy(idx==1,1),xy(idx==1,2),'r.','MarkerSize',12)
hold on
plot(xy(idx==2,1),xy(idx==2,2),'b.','MarkerSize',12)
plot(c(:,1),c(:,2),'kx','MarkerSize',12,'LineWidth',2)
plot(c(:,1),c(:,2),'ko','MarkerSize',12,'LineWidth',2)
legend(['Cluster 1 (SSE: ' num2str(sse(1))],...
    ['Cluster 2 (SSE: ' num2str(sse(2))],...
    'Centroids', 'Location','NW')
```

Appendix-F [Blood Pressure Clustering]

% GUI based k-means for Blood Pressure data

findcluster('d:\Imp\thesis\bp1.dat')

Appendix- G [Matlab Code for Blood Pressure K MEANS]

```
% K-means clustering of Blood Pressure data
clc
clear all
close all
X = load('E:\Imp\thesis\bp2.dat');
opts = statset('Display','final');
[cidx, ctrs] = kmeans(X, 2, 'Distance','city', 'Replicates',5,
'Options',opts);
plot(X(cidx==1,1),X(cidx==1,2),'r.', X(cidx==2,1),X(cidx==2,2),'b.',
ctrs(:,1),ctrs(:,2),'kx');
title('Clustering of Blood pressure data using K-Means clustering
algorithm');
```

Appendix-H [Matlab Code for Blood Pressure Dendrograms]

```
% Dendrogram for Blood Pressure data:
X = load('E:\Imp\thesis\bp1.dat');
Y = pdist(X,'cityblock');
Z = linkage(Y,'average');
[H, T] = dendrogram(Z);
```

Appendix-I [Matlab Code for Fuzzy C MEANS]

```
% fuzzy C means for Blood Pressure data
data = load('E:\Imp\thesis\bp2.dat');
[center,U,obj_fcn] = fcm(data,2);
plot(data(:,1), data(:,2),'o');
hold on;
maxU = max(U);
% Find the data points with highest grade of membership in cluster 1
index1 = find(U(1,:) == maxU);
% Find the data points with highest grade of membership in cluster 2
index2 = find(U(2,:) == maxU);
line(data(index1,1),data(index1,2),'marker','*','color','g');
line(data(index2,1),data(index2,2),'marker','*','color','r');
% Plot the cluster centers
plot([center([1 2],1)],[center([1 2],2)],'*','color','k')
hold off;
```

Appendix- J [Matlab Code for Blood Pressure SSE]

```
clc
close all
clear all
xy = [rand(100,2)*2+3; rand(100,2)];
% Visualize clouds
scatter(xy(:,1),xy(:,2))
close
% Cluster
[idx,c,sse] = kmeans(xy,2);
% Plot clouds with centroids and legend
plot(xy(idx==1,1),xy(idx==1,2),'r.','MarkerSize',12)
hold on
plot(xy(idx==2,1),xy(idx==2,2),'b.','MarkerSize',12)
plot(c(:,1),c(:,2),'kx','MarkerSize',12,'LineWidth',2)
plot(c(:,1),c(:,2),'ko','MarkerSize',12,'LineWidth',2)
legend(['Cluster 1 (SSE: ' num2str(sse(1))],...
    ['Cluster 2 (SSE: ' num2str(sse(2))],...
    'Centroids', 'Location','NW')
```

Appendix- K [Matlab Code for Fuzzy max-min]

```
% The Matlab program for finding fuzzy relation using fuzzy max-min
%method is Program
%enter the two vectors whose relation is to be found
R=input('enter the first vector');
S=input('enter the second vector');
% find the size of two vectors
[m,n]=size(R)
[a,b]=size(S)
if(n==a)
    for i=1:m
        for j=1:b
            c=R(i,:)
            d=S(:,j)
```

```
            f=d
            %find the minimum of two vectors
            e=min(c,f)
            %find the maximum of two vectors
            h(i,j)=max(e);
        end
    end
    %print the result
    display('the fuzzy relation between two vectors is');
    display(h)
else
    display('The fuzzy relation cannot be found')
end
```

Appendix- L [Matlab Code for Fuzzy Cartesian Product]

```
% The Matlab program for the max-product method is shown below
%Program
%enter the two input vectors
R=input('enter the first vector');
S=input('enter the second vector');
%find the size of the two vector
[m,n]=size(R)
[a,b]=size(S)
if(n==a)
    for i=1:m
        for j=1:b
            c=R(i,:);
            d=S(:,j);
            [f,g]=size(c);
            [h,q]=size(d);
            %finding product
            for l=1:g
                e(1,l)=c(1,l)*d(l,1);
            end
            %finding maximum
            t(i,j)=max(e);
        end
    end
else
    display('Cannot be find min-max');
end
```

ANNEXURE

Name – Yuvraj Bhushan Khare

DOB - 26/11/1984

Contact: +91 9501313762

EDUCATIONAL AND PROFESSIONAL CREDENTIALS

M. E. (Electronics Instrumentation & Control Engineering) 2011
Thapar University 8.34

B. Tech (Electronics & Instrumentation Engineering) 2008
Uttar Pradesh Technical University 63.82%

12th Standard 2002
K.V. No.3 Jhansi CBSE, 73%

10th Standard 2000
K.V. No.3 Jhansi CBSE, 63.4%

Publication :

1. "PID Controller of Heat Exchanger System", International Journal of Computer Application (0975-8887) Volume 8-No.6 October 2010.
2. "Internal Model Based PID Control of Shell and Tube Heat Exchanger System", Proceeding of the 2011 IEEE Students' Technology Symposium 14-16 January, 2011, IIT Kharagpur.