Development of Hierarchical Clustering Techniques for Gridded Data from Mixed Data Sequences

Thesis submitted in fulfillment of the requirements for the award of the degree of

Doctor of Philosophy

Under the guidance of

Dr. A. Unnikrishnan

and

Dr. K. Poulose Jacob

By

Bindiya M. Varghese

COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY

Department of Computer Science
Cochin University of Science and Technology
Kochi- 682022. India
2013

Certificate

*This is to certify that the thesis entitled "**Development of Hierarchical Clustering Techniques for Gridded Data from Mixed Data Sequences**" is a bonafide record of the research work carried out by **Ms. Bindiya M. Varghese** under our supervision and guidance in the Department of Computer Science, Cochin University of Science and Technology, Kochi. The results presented in this thesis or parts of it have not been presented for the award of any other degree.*

Dr. K Poulose Jacob
Co-guide
Professor and Head
Department of Computer Science
Cochin University of Science and Technology
Kochi-682022, Kerala

Dr. A. Unnikrishnan
Supervising guide
Outstanding Scientist
Naval Physical Oceanographic Laboratory
Kochi
Kerala

June 7, 2013

Declaration

*I hereby declare that the work presented in this thesis entitled "**Development of Hierarchical Clustering Techniques for Gridded Data from Mixed Data Sequences**" is based on the original research work carried out by me under the supervision and guidance of **Dr. A. Unnikrishnan**, Outstanding Scientist, NPOL, Kochi, Kerala with **Dr. K. Poulose Jacob**, Professor and Head, Department of Computer Science, Cochin University of Science and Technology, Kerala as co-guide. The results presented in this thesis or parts of it have not been presented for the award of any other degree.*

Kochi-682022
June 7, 2013

Bindiya M Varghese
Register No: 3444

Acknowledgements

"For the LORD gives wisdom, and from his mouth come knowledge and understanding." Proverbs 2:6

I would first like to thank God who gave me the grace and privilege to pursue this research and successfully complete it in spite of many challenges faced. The journey has been quite remarkable and it is a unique stepping stone to many endeavors ahead.

I have been extremely lucky and blessed to have a research guide like ***Dr. A. Unnikrishnan****, who cared so much about my work, and who responded to my every questions and queries so promptly. He has been a research guide, a mentor, a well-meaning critic, and a kind teacher, all rolled into one amazing person. I would like to express my heartfelt gratitude to my guide, for the patient guidance, support and advice he has provided throughout my time as his student.*

I also reserve special thanks to ***Dr. K Poulose Jacob****, co-guide for my research, for his support and invaluable suggestions throughout the study. I extend my gratitude to all the members of Administration of the Department of Computer Science, especially* ***Ms. Girija****, who has helped me clearing all the administrative hurdles during the period.*

At this juncture, I must record my sincere gratitude to the ***Management of Rajagiri College of Social Sciences, Kalamassery****, for permitting me to do this research along with my duties in the college. In particular, I would like to thank* ***Dr. Joseph I. Injodey****, who initiated my doctoral studies, by signing the no-objection certificate and who has backed me up in every circumstances of my career.*

I am grateful to my colleagues, ***Dr. P. X. Joseph, Ms. Abusha Zafar, Mr. Madhu S. Nair, Dr. Baby M D, Ms. Jaya Vijayan, Ms.Vimina E. R, Ms. Prema S. Thomas, Ms. Sunu Abraham, Mr.Renjtih P. R, Mr. Shiju Thomas, Mr. Ajith***

V. Abraham, Ms. Sumitra Binu, Mr. Tony M. J *and* ***Mr. Arun Sam*** *and all my friends for their unwavering support.*

I thank ***Dr. Ganghadharan, Dr. Arun Warrior*** *and* ***Dr. Anupama G****, and all the staff of department of Oncology, Lakeshore hospital. When times were tough, you gave me the confidence and strength to keep up.*

Words are not enough to thank my family. The sacrifice you made throughout my years here are simply ineffable. I thank my ***mother*** *and my* ***mother–in-law****, for their tender, loving care and compassion. I thank my sister* ***Beena*** *for persistent push to achieve all my goals. I send my love and prayers to* ***my father, father-in-law*** *and* ***my sister*** *who were inspirational to kindle my interest in research, but am now with God showering me with blessings. Finally, words would not suffice to express my gratitude towards my husband* ***Abhilash*** *for the faith he has in me and the support he gave throughout this endeavor and in every aspect of my life. Finally this thesis is dedicated to you, my dear little ones,* ***Ayn, Antony*** *and* ***Ivan****.*

Bindiya

Table of Contents

List of Figures

List of Tables

Abstract

Knowledge discovery in databases is the non-trivial process of identifying valid, novel potentially useful and ultimately understandable patterns from data. The term Data mining refers to the process which does the exploratory analysis on the data and builds some model on the data. To infer patterns from data, data mining involves different approaches like association rule mining, classification techniques or clustering techniques. Among the many data mining techniques, clustering plays a major role, since it helps to group the related data for assessing properties and drawing conclusions. Most of the clustering algorithms act on a dataset with uniform format, since the similarity or dissimilarity between the data points is a significant factor in finding out the clusters. If a dataset consists of mixed attributes, i.e. a combination of numerical and categorical variables, a preferred approach is to convert different formats into a uniform format. The research study explores the various techniques to convert the mixed data sets to a numerical equivalent, so as to make it equipped for applying the statistical and similar algorithms. The results of clustering mixed category data after conversion to numeric data type have been demonstrated using a crime data set. The thesis also proposes an extension to the well known algorithm for handling mixed data types, to deal with data sets having only categorical data. The proposed conversion has been validated on a data set corresponding to breast cancer. Moreover, another issue with the clustering process is the visualization of output. Different geometric techniques like scatter plot, or projection plots are available, but none of the techniques display the result projecting the whole database but rather demonstrate attribute-pair wise analysis.

With a view to get a better visualization of a multi-dimensional dataset, the technique for dimensionality reduction techniques is explored and implemented. The

capability of dimensionality reduction techniques to permit clustering on the reduced dimension, degrading the quality of the clustering is established, by analyzing as many as four standard data sets. Since dimensionality reduction to 2-D results in a gridded representation of a multidimensional dataset, the study progresses on spatial clustering techniques on the resultant spatial image. Spatial data structure quad tree is used for representing the gridded image. The homogeneity of the region is considered as the factor to decompose the quadrants into sub-level quadrants. While statistical measures like mean and variance have been examined for deciding homogeneity, the thesis proposes methods based on fuzzy set theory. The fuzzy rules are used to decompose the quad tree, to avoid crisp boundaries of mean and variance. The information dense regions are extracted from the leaf nodes of the quad tree and proper clusters are identified.

The clusters identified at lower dimensions using spatial clustering are indexed back into the original data set to explain the implications of clusters. The final demonstration of the FARS data set finally underscores the efficacy of all the techniques developed in the thesis to cluster high-dimensional, multi category data and establish the properties of clusters based on the variables in each record. These clusters helps to infer useful patterns out of the dataset based on the domain of data. The thesis thus reports results of the study undertaken with the following objective:

(i) Convert a mixed attributed data set into a uniform format to make it equipped for general clustering algorithms

(ii) Find out a gridded representation of high- dimensional normalized uniform formatted dataset, after reducing the dimensionality

(iii) Spatially cluster the gridded representation and extract low level information content from dense regions using spatial data structures.

(iv) Map the cluster index back into the original dataset, without any information loss, and establish grouping of records based on some leading traits.

Chapter 1

Introduction

This chapter gives a brief introduction to knowledge discovery process and basic data mining concepts. It also explains the motivation behind the research work and defines the objective of the research study. The contributions of the research work are also detailed in this chapter and it concludes with the organization of the thesis.

1. Introduction

1.1. Knowledge discovery

Knowledge discovery in databases is *the non-trivial process of identifying valid, novel potentially useful and ultimately understandable patterns from data*. The process predicts the future trends and behaviors from accumulated large volumes of data to make proactive, knowledge driven decisions. At an abstract level, the core of knowledge discovery process is to map low-level data which is too voluminous in nature to compact, predictive model for estimating the values of future cases. The phrase knowledge discovery in databases (KDD) was first coined at the first KDD workshop in 1989 [1].The term process implies that KDD comprises many steps, which involve data preparation, search for patterns, knowledge evaluation, and refinement, all repeated in multiple iterations. The steps involved in the KDD process is diagrammatically represented below [2].

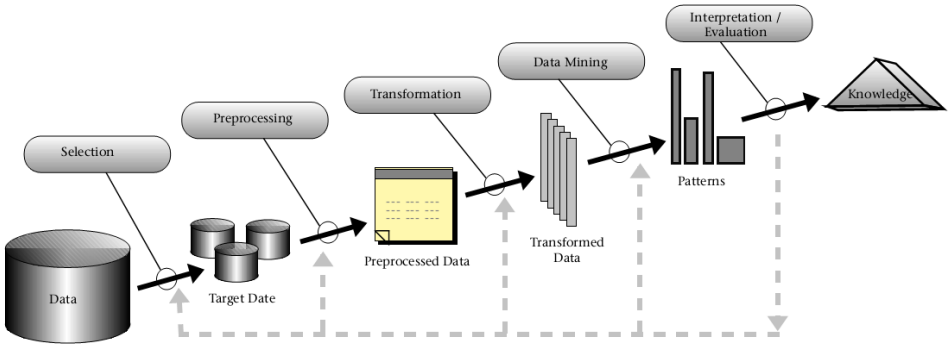


Figure 1-1: The Overview of the steps in the KDD process

The knowledge discovery process starts with the understanding of the application domain and identifying the goal of the process from the user's perspective. A target data which is relevant to the analysis task is selected on which

the discovery process is to be applied. The selected data may be considered dirty which may contain noisy values or missing values. The preprocessing step cleans the dirty data to a more consistent form. Next step in the process transforms the data into a more consolidated form appropriate for mining by performing operations like normalization, aggregation etc. Data reduction helps to find useful features which can represent the large dataset eliminating the redundant factors.

The next step , which is the actual data mining process does the exploratory analysis on the data and build some model based on known techniques of statistics, neural networks , machine learning and pattern recognition. Data mining is the step in KDD process that consists of applying data analysis and discovery algorithms on preprocessed subsamples, and transformed data. Data mining is the application of specific algorithms for extracting patterns from data. Data mining involves many different approaches to accomplish the pattern extraction tasks. The data mining process tries to fit a model to the data. The algorithm examines the data and determines a model that is closest to the characteristics of the data being examined. Data mining tasks can be classified into two categories, descriptive data mining and predictive data mining. The former describes the data in a concise and summarized manner whereas the latter constructs one or a set of models performs inference on the available set of data and attempts to predict the behavior of new data sets. The patterns of interest from the models are visually represented for better interpretation in the final step. These steps need not be mutually exclusive and may need iterations to fine tune the results.

A related field evolving from databases is data warehousing, which refers to the trend of collecting and cleaning transactional data to make them available for online analysis and decision support. Data warehousing helps set the stage for KDD in two important ways: (1) data cleaning and (2) data access. Data selection and preprocessing: the initial step produces a data store which is created by integrating

data from a number of databases. When integrating data, the problems like identifying data, missing data, data conflict and ambiguity may arise. An Extraction transformation and loading tool (ETL) is used to overcome these problems.

1.2. Data mining Approaches

1.2.1. Based on Statistical attributes

Research in Statistics has helped to produce many of the proposed data mining algorithms. Statistical concepts like determining a data distribution and calculating a mean and variance can be viewed as data mining techniques and each of these is a descriptive model for the data under consideration. Sampling is a technique which is often used in data mining. A subset of total population is examined and a generalization about the entire population is made from the subset.

1.2.2. Machine learning

Machine learning is the area of artificial intelligence that examines how to develop systems and algorithms that can learn, based on the feedback of prediction made by it, thereby correcting the domain knowledge also. When machine learning is applied to data mining tasks, a model is built to represent data. Samples from the entire database are used to train the model and generate the model parameter. Further samples are applied to the model to perform the task of data mining. There are two different types of machine learning:

(i) supervised learning where the model is trained from examples

(ii) Unsupervised learning, where the model learns on its own from the data stream.

1.2.3. Classification

Supervised classification works well, if data is known to have some predefined classes. Classification is a data mining technique used to predict group membership for data instances. Classification maps data into predefined groups or classes. Pattern recognition forms the basis of classification, where an input pattern is classified into one of the several classes based on similarity (or proximity) to the predefined classes.

1.2.4. Regression

Regression deals with the estimation of an output value based on a sequence of input values. It can be used to map a data item to a real-valued prediction variable. Regression involves in learning of the function that does this mapping. Regression tries to fit the target data to some known types of function like linear regression or logistic regression.

1.2.5. Neural network based algorithms.

An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. It is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems. An artificial neuron is a device with many inputs and one output. The neuron has two modes of operation; the training mode and the using mode. In the training mode, the neuron can be trained to fire (or not), for particular input patterns A firing rule determines how one calculates whether a neuron should fire for any input pattern. It relates to all the input patterns, not only the ones on which the node was trained. The ANN consists of three or more layers: a layer of "input" units is connected to one layer of "hidden" units; the output of a hidden layer is connected to either another hidden layer or to a layer of "output" units. The activity of the input units represents the raw

information that is fed into the network. The activity of each hidden unit is determined by the activities of the input units and the weights on the connections between the input and the hidden units. The behavior of the output units depends on the activity of the hidden units and the weights between the hidden and output units. The weights between the input and hidden units determine when each hidden unit is active, and so by modifying these weights, a hidden unit can choose what it represents. The neural network is trained by computing the weights using example cases in the supervised learning mode or the network learns on its own in the case of unsupervised learning mode. A trained neural network can thus represent a model in the case of machine learning or a regression function.

1.3. Scope of the Thesis

Clustering is a descriptive task that seeks to identify homogeneous groups of objects based on the values of their attributes. Since no classes are known prior for most of the real world data, substantial research is going on in unsupervised learning techniques, to evolve the classes. The present study specifically focuses on clustering of multidimensional mixed category data. Most of the clustering algorithms act on a dataset with uniform format, since the similarity or dissimilarity between the data points is a significant factor in finding out the clusters. If a dataset consists of mixed attributes, i.e. a combination of numerical and categorical variables, then a preprocessing step is done distinctly for different data types. A preferred approach is to convert different formats into a uniform format. The present research study explores the various techniques to convert the categorical attributes to a numerical equivalent, so as to make it equipped for applying the common clustering algorithms. The techniques developed in the thesis are then applied to the converted data types and the results are illustrated.

Moreover, another issue, with the clustering process, which merits attention, is the visualization of output. Visualization of higher dimensional data on a lower dimension is an attractive feature in evaluating properties of clusters. Different geometric techniques like scatter plot, or projection plots are available; but most of the techniques display the result suitable for pair wise analysis of attributes. On the other hand, projecting the whole database to two dimensions, based on importance of the chosen dimension, often gives impressive visualization. In-order to represent a multi-dimensional dataset into a plane or a space, dimensionality reduction techniques are explored and implemented. The clustered data set then becomes gridded in two dimensions

With the gridded representation of a multidimensional dataset, the present study further progresses on spatial clustering techniques, on the resultant spatial image (or gridded data). The Quad tree, a prevalent spatial data structure for representing two dimensional data based on regional homogeneity, is investigated for suitability in representing the gridded data in two dimensions The homogeneity of the region is considered as the factor to decompose the quadrants into sub-level quadrants. Though the homogeneity can be measured using statistical measures like mean and variance or entropy, the present work uses fuzzy rules to decompose the quad tree, to avoid crisp boundaries of mean and variance. The information dense regions can be extracted from the leaf nodes of the quad tree, using quadrant merging. The research also explores the possibility of using the information content of the regions to identify the edges of clusters from the merged quadrants.

In all the phases of research, the record level identifications of the observations in the dataset are maintained and this helps to remap the cluster index into the original dataset. The clusters identified can be indexed back to the original data set for subjective interpretation and assimilation. These clusters will help to infer useful patterns out of the dataset based on the domain of data.

1.4. Objective of the thesis

The objective of this research study is to construct a general framework which will

- Convert a mixed attributed data set into a uniform format to make it equipped for general clustering algorithms, which can be extended to completely categorical dataset,
- Find out a gridded representation of high- dimensional normalized uniform formatted dataset, after reducing the dimensionality,
- Spatially cluster the gridded representation and extract the cluster edges after merging the neighbor dense regions using spatial data structures,
- Map the cluster index back into the original dataset, without any information loss, and establish grouping of records based on some leading traits.

1.5. Structure of the thesis

The thesis is organized as follows. Chapter 2 provides the literature survey done as part of this research, which helped the author to focus on the research topic, by deriving the hints and directions from the published results, at the same time maintaining the individually of the approach. The chapters to follow describe work carried out and the contributions of the thesis. Toward this, the Chapter 3 explains the preprocessing steps applied on the mixed attributed dataset to convert it into a uniform numerical format. The chapter also suggests an extension of the algorithm to a completely categorical dataset. The algorithm is applied on a mixed attributed crime dataset and a combined hierarchical-k-means clustering is done on the numerical equivalent of the crime dataset to demonstrate the influence of the procedure. Chapter 4 explores the technique of visually representing the high-dimensional dataset obtained from the first phase using singular value decomposition

of the space corresponding to the data set and projecting the data set to chosen reduced dimensions. The chapter concludes with the experimentation with standard data sets from UCI repository like Iris, Wine, Yeast, Thyroid, and Breast Cancer. The crime dataset used in the chapter 3 is also experimented with in order to ascertain the outcome of the technique proposed. The dimensionality reduction and thereby projection to a lower dimension of the high-dimensional mixed attributed dataset, results in a spatial image, which is then used for spatial clustering. Chapter 5 details the application of fuzzy techniques for the decomposition of quad tree, merging the neighbor dense quadrants using, fuzzy rules. This chapter also deals with the final contribution of the thesis viz. the remapping of clusters into the original dataset, to infer useful patterns from the clusters of data elements. Though every chapter is provided with case studies on different domain, a complete demonstration of the framework of the technique developed as part of the thesis is done on FARS (Fatal Accident Reposting System) dataset, which has got more than 37 thousand observations defined with 16 variables of mixed type and the case study is presented as the chapter 6. Chapter 7 summarizes the conclusion and outlines the future scope of the research study.

Chapter 2

..... Literature Survey

The goal of this literature survey is to provide a extensive walk through of different techniques in clustering relevant to the area of mining of mixed data sequences and visualization techniques in data mining. The chapter is organized based on the classification of clustering algorithms and concludes with the limitations of the existing clustering algorithms and challenges in visualizing the data mining output, which in turn is leading to the present research study.

.....

2. Literature Survey

2.1. Introduction

The main focus of the research is clustering and hence many literatures were analyzed during the study. Another aspect of the research was the visualization in data mining which is a very widespread and vital part.

Clustering is a descriptive task that seeks to identify homogeneous groups of objects based on the values of their attributes [3]. In spatial data sets, clustering permits a generalization of the spatial component like explicit location and extension of spatial objects which define implicit relations of spatial neighborhood. Current spatial clustering techniques can be broadly classified into three categories, viz. partitional, hierarchical and density based and model based algorithms. The term object invariably represents the data points in the dataset.

2.2. Partition based algorithms

Given a set of objects and a clustering criterion, partitional clustering obtains a partition of objects into clusters, such that the objects in a cluster is more similar to the objects inside the cluster than to objects in other clusters. Partitional clustering algorithms attempt to decompose the dataset directly into a set of k disjoint clusters, provided k is the number of initial clusters. An iterative optimization is done to emphasize the local structure of data, which involves minimizing some measure of dissimilarity in the objects within the cluster, while maximizing the dissimilarity of different clusters. Partitional algorithms are generally iterative in nature and converge to some local optima. Given a set of data points $x_i \in \Re^l$, $i = 1,\ldots,N$, partitional clustering algorithms aim to organize them into K

clusters $\{C_1, \ldots, C_K\}$ while maximizing or minimizing a pre-specified criterion function J [4].

2.2.1. K-Mean

K-means is one of the simplest unsupervised learning algorithms used for clustering. K-means partitions n objects into k clusters in which each object belongs to the cluster with the nearest mean. This algorithm aims at minimizing an objective function, in this case a squared error function [5]. The algorithm aims to minimize the objective function

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^j - c_j \right\|^2 \quad \text{.....Equation 2-1}$$

where $\left\| x_i^j - c_j \right\|^2$ is a chosen distance measure between a data point x_i^j and the cluster centre c_j, is an indicator of the distance of the n data points from their respective cluster centres.

2.2.2. K-medoid

K-medoids algorithms are partitional algorithm which attempt to minimize squared error, the distance between points labeled to be in a cluster and a point designated as the center of that cluster [6]. A medoid can be defined as that object of a cluster, whose average dissimilarity to all the objects in the cluster is minimal i.e. it is a most centrally located point in the given data set. In contrast to the *k*-means algorithm *k*-medoids chooses data points as centers.

2.2.2.1. PAM

The Partitioning around medoid (PAM) algorithm represents a cluster by a medoid [7] . PAM is based on the search for k representative objects among the objects of the data set. These objects should represent various aspects of the structure of the data are often called centrotypes. In the PAM algorithm the

representative objects are the so-called medoid of the clusters [6]. After finding a set of k representative objects, the k clusters are constructed by assigning each object of the data set to the nearest representative object.

Initially, a random set of K items is taken to be the set of medoids. Then, at each step, all items other than the chosen medoids from the input sample set are examined one by one to see if they should be the new medoids. The algorithm chooses the new set of medoids which improves the overall quality of the clustering and replaces the old set of medoids with them.

Let K_i be the cluster represented by the medoid t_i. To swap with a non medoid t_h, the cost change of an item t_j associated with the of exchange of t_i with t_h, C_{jih} has to be computed [8]. The total impact to quality by a medoid change TC_{jih} is given by

$$TC_{jih} = \sum_{j=1}^{n} C_{jih} \quad \text{.....Equation 2-2}$$

The k-medoid methods are very robust to the existence of outliers. Also, Clusters found by K-medoid methods do not depend on the order in which the objects are examined. They are invariant with respect to translations and orthogonal transformations of data points. PAM does not scale well to large datasets because of its computational complexity. For each iteration, the cost TC_{jih} has to be computed for $k(n-k)$ pair of objects. Thus the total complexity per iteration is $k(n-k)^2$, thereby making PAM not an alternative for large databases.

2.2.2.2. CLARA

CLARA (Clustering Large Applications) improves on the time complexity of PAM [7]. CLARA relies on sampling. PAM is applied to samples drawn from the large datasets. For better approximations, CLARA draws multiple samples and gives best clustering as the result. For accuracy, the quality of a clustering is measured

based on the average dissimilarity of all objects in the entire data set, and not only of those objects in the samples. The method used in CLARA, which was first described by Kaufman and Rousseeuw (1986), is based on the selection of five (or more) random samples of objects. The size of the samples depends on the number of clusters. For a clustering into k clusters, the size of the samples is given by *(40 + 2k)* for CLARA, by applying PAM just to the samples, each iteration is of *O(k(40 + k)2 + k(n - k))*. This explains why CLARA is more efficient than PAM for large values of *n*.

2.2.2.3. CLARANS

CLARANS (Clustering Large Applications based on Randomized Search) improves on CLARA by using multiple different samples [7]. While CLARA draws a sample of nodes at the beginning of a search, CLARANS draws a sample of neighbors in each step of a search. This has the benefit of not confining a search to a localized area. In addition to the normal input to PAM, CLARANs uses two additional parameters; *numlocal* and *maxneighbor*. Numlocal indicates the number of samples to be taken. The *numlocal* also indicates the number of clustering to be made since a new clustering has to be done on every sample. *Maxneighbor* is the number of neighbors of a node to which any specific node can be compared. As *maxneighbor* increases, CLARANs resemble PAM, because all nodes are to be examined. J. Han et al shows the good choice for the parameters are *numlocal = 2* and *maxneighbor = max ((0.0125* x *k(n-k)),250)*. The disadvantage of CLARANS is that it assumes all data are in main memory. A related research proposes fuzzy CLARANS for text clustering [9]and another work suggest ECLARANS, a modified CLARANS for outlier detection [10].

2.2.3. Related works in partitional clustering

Substantial research has been going on in the field of partitional clustering and a number of studies have been published [11]. Significant research studies has brought out improvement on deciding initial k cluster centers, acceleration of k-means, merging other clustering techniques with k-means for various domain [12], [13], [14], [15] [16], [17] , [18]. Modified and extended versions of PAM is used in clustering gene expression data [19] , in mobile network planning [20] and in Image compression [21]. Modified CLARA is used in image segmentation by reducing the error due to sampling [22] and a fuzzy c-medoids based CLARA are related works to the field [23].

Spatial dominant CLARANS assumes the data set to contain spatial and non-spatial components [7]. The general approach is to cluster spatial components using CLARANS and then examines the non-spatial within each cluster to derive a description of that cluster. For mining spatial attributes, a tool named DBLEARN is used [24]. From a learning request, DBLEARN first extracts a set of relevant tuples via SQL queries. Then based on the generalization hierarchies of attributes, it ,iteratively generalizes the tuples. SDCLARANS is a combination of CLARANS and DBLEARN. Opposite to SDCLARANS, NSDCLARANS considers the non-spatial attributes in the first phase [24]. DBLEARN is applied to the non-spatial attributes, until the final number of generalized tuples fall below a certain threshold. For each generalized tuple obtained above, the spatial components of the tuples represented by the current generalized tuple are collected, and CLARANS is applied.

2.3. Hierarchical Based algorithms

A dataset is said to be a hierarchical cluster if there exists 2 samples, c1 and c2, which belong in the same cluster at some level k and remain clustered together at

all higher levels > k [25]. The hierarchy is represented as a tree, called a dendrogram, with individual elements at one end and a single cluster containing every element at the other. Hierarchical clustering algorithms are either top-down or bottom-up. Bottom-up algorithms called hierarchical agglomerative clustering, treat each object as a singleton cluster at the outset and then successively merge (or agglomerate) pairs of clusters until all clusters have been merged into a single cluster. Top-down or divisive clustering proceeds by splitting clusters recursively until individual objects s are reached. The agglomerative algorithms described below differ in the strategy for merging in bottom-up approach and the divisive algorithms differs in the strategy for splitting in top-down approach.

2.3.1. Agglomerative algorithms

2.3.1.1. CURE

CURE (Clustering Using Representatives) identifies clusters having non-spherical shapes and wide variances in size [26]. CURE is a bottom-up hierarchical clustering algorithm, but instead of using a centroid-based approach or an all-points approach that is based on choosing a well-formed group of points to identify the distance between clusters. CURE achieves this by representing each cluster by a certain fixed number of points that are generated by selecting well scattered points from the cluster. In fact, CURE begins by choosing a constant number, c of well scattered points from a cluster. These points are used to identify the shape and size of the cluster. The next step of the algorithm shrinks the selected points toward the centroid of the cluster using some pre- determined fraction α. These scattered points after shrinking are used as representatives of the cluster. The clusters with the closest pair of representative points are the clusters that are merged at each step of CURE's hierarchical clustering algorithm. CURE is less sensitive to outliers since shrinking the scattered points toward the mean reduces the

adverse effects due to outliers since outliers are typically further away from the mean and are thus shifted a larger distance due to the shrinking.

The kinds of clusters identified by CURE can be tuned by varying α: between 0 and 1. CURE reduces to the centroid-based algorithm if $\alpha = 1$, while for $\alpha = 0$, it becomes similar to the all-points approach. CURE's hierarchical clustering algorithm have a space complexity linear to the input size n and has a worst-case time complexity of $O(n^2 \log n)$. For lower dimensions the complexity is further reduced to $O(n^2)$. The overview of CURE algorithm can be diagrammatically represented as on Figure2-1 [26].

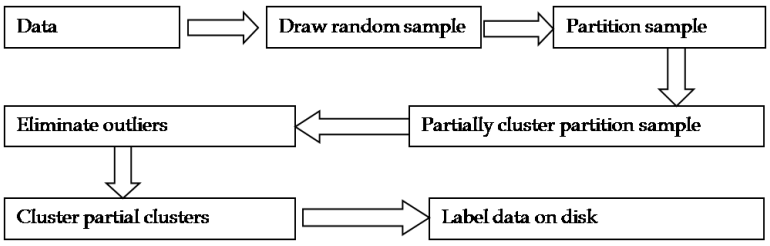


Figure 2-1: Overview of CURE

2.3.1.2. ROCK

ROCK (Robust Clustering using links) implements a new concept of links to measure the similarity/proximity between a pair of data points [27]. A pair of data points are considered neighbors if their similarity exceeds a certain threshold. The number of links between a pair of points is then the common neighbors for the points. Points belonging to a single cluster will have a large number of common neighbors. Let $sim(p_i,p_j)$ be a similarity function that is normalized and captures the closeness between the pair of points p_i and p_j. The $sim(p_i,p_j)$ assumes values between 0 and 1. Given a threshold θ between 0 and 1, a pair of points (p_i, p_j) is defined to be neighbors if $sim\ (p_i,p_j)>\theta$. *Link* (p_i,p_j), the number of common neighbors between the pair of points p_i and p_j. The criterion function is to maximize the sum of $link(p_q,p_r)$ for

data pairs p_q, p_r belonging to a single cluster and at the same time, minimize the sum of $link(p_q,p_s)$ for p_q and p_s in different clusters.

i.e. Maximize

$$\sum_{i=1}^{k} n_i * \sum_{p_q p_r \in C_i} \frac{link\,(p_q,\ p_r)}{n_i^{1+2f\,(\theta)}} \quad \text{.....Equation 2-3}$$

where cluster C_i denotes cluster *i* of size *n*. The worst case time complexity of the algorithm is $O\ (n_2 + nm_m m_a + n^2\ log\ n)$, where m_m is the maximum number of neighbors, m_a is the average number of neighbors, and *n* is the number of data points. The space complexity is $O(\ min\{n^2,\ nm_m m_a\})$

2.3.1.3. CHAMELEON

CHAMLEON measures the similarity based on a dynamic model [28]. Two clusters are merged only if the inter-connectivity and closeness between two clusters are high relative to the internal inter-connectivity of the clusters and closeness of data points within the clusters. CHAMELEON operates on a sparse graph in which nodes represent data items, and weighted edges represent similarities among the data items. This sparse graph representation of the data set allows CHAMELEON to scale to large data sets.

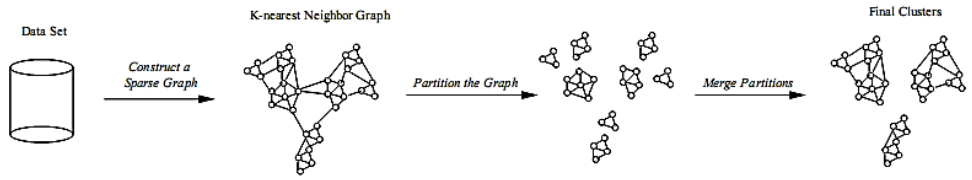


Figure 2-2: Overview of CHAMELEON algorithm

CHAMELEON finds the clusters in the data set by using a two phase algorithm. During the first phase, CHAMELEON uses a graph partitioning algorithm to cluster the data items into a large number of relatively small sub-clusters. During the second phase, it uses an agglomerative hierarchical clustering

algorithm to find the genuine clusters by repeatedly combining together these sub-clusters.

CHAMELEON's sparse graph representation of the data items is based on the k-nearest neighbour graph approach. Each vertex of the k-nearest neighbour graph represents a data item, and there exists an edge between two vertices, if data items corresponding to either of the nodes are among the k-most similar data points of the data point corresponding to the other node. CHAMELEON determines the similarity between each pair of clusters C_i and C_j by looking both at their relative inter-connectivity $RI(C_i, C_j)$ and their relative closeness $RC(C_i, C_j)$. CHAMELEON's hierarchical clustering algorithm selects to merge the pair of clusters for which both $RI(C_i, C_j)$ and $RC(C_i, C_j)$ are high; i.e., it selects to merge clusters that are well inter-connected as well as close together with respect to the internal inter-connectivity and closeness of the clusters. The relative inter-connectivity between a pair of clusters C_i and C_j is defined as the absolute inter-connectivity between C_i and C_j normalized with respect to the internal inter-connectivity of the two clusters C_i and C_j. The absolute inter-connectivity between a pair of clusters C_i and C_j is defined to be as the sum of the weight of the edges that connect vertices in C_i to vertices in C_j . This is essentially the edge-cut of the cluster, $EC_{\{Ci,Cj\}}$ containing both C_i and C_j such that the cluster is broken into C_i and C_j . The relative inter-connectivity between a pair of clusters Ci and Cj is given by

$$RI(C_i, C_j) = \frac{|EC\{C_i, C_j\}|}{\frac{|EC_{C_i}| + |EC_{C_j}|}{2}} \quad \text{.....Equation 2-4}$$

which normalizes the absolute inter-connectivity with the average internal inter-connectivity of the two clusters. The relative closeness between a pair of clusters Ci and Cj is computed as,

$$RC(C_i,C_j) = \frac{\bar{S}_{EC\{C_i,C_j\}}}{\frac{|C_i|}{|C_i|+|C_j|}\bar{S}_{EC_{C_i}} + \frac{|C_j|}{|C_i|+|C_j|}\bar{S}_{EC_{C_j}}}$$ .....Equation 2-5

where $\bar{S}_{EC_{C_i}}$ and $\bar{S}_{EC_{C_j}}$ are the average weights of the edges that belong in the min-cut bisector of clusters C_i and C_j ,respectively, and $\bar{S}_{EC\{C_i,C_j\}}$ is the average weight of the edges that connect vertices in C_i to vertices in C_j . The overall complexity of CHAMELEON's two-phase clustering algorithm is $O(nm + n \log n + m^2 \log m)$.

2.3.2. Divisive Algorithms

2.3.2.1. STING

Statistical Information Grid-based method exploits the clustering properties of index structures [29]. The spatial area is divided into rectangular cells which forms a hierarchical structure. Each cell at a high level is partitioned to form a number of cells of the next lower level. Statistical information of each cell is calculated and stored beforehand and is used to answer queries. For each cell, two types of parameters are considered; attribute-dependent and attribute-independent parameters. The attribute- independent parameter are the number of objects (points) in this cell, say n. Attribute-dependent parameters are *m*: mean of all values in this cell, *s*: standard deviation of all values of the attribute in this cell, *min*: the minimum value of the attribute in this cell, *max*: the maximum value of the attribute in this cell, *distribution:* the type of distribution that the attribute value in this cell follows.

Clustering operations are performed using a top-down method, starting with the root. The relevant cells are determined using the statistical information and only the paths from those cells down the tree are followed. Once the leaf cells are reached, the clusters are formed using a breadth-first search, by merging cells based on their proximity and whether the average density of the area is greater than some

specified threshold. The computational complexity is O(K), where K is the number of grid cells at the lowest level. Usually K << N, where N is the number of objects.

2.3.2.2. STING+

STING+ is an approach to active spatial data mining, which takes advantage of the rich research results of active database systems and the efficient algorithms in STING [29] for passive spatial data mining [30]. A region in STING+ is defined as a set of adjacent leaf level cells. Also, object density and attribute conditions in STING+ are defined in terms of leaf level cells. The density of a leaf level cell is defined as the ratio of the number of objects in this cell divided by the area of this cell. A region is said to have a certain density c if and only if the density of every leaf level cell in this region is at least c. Conditions on attribute values are defined in a similar manner. Two kinds of conditions can be specified by the user. One condition is an absolute condition, i.e., the condition is satisfied when a certain state is reached. The other type of condition is a relative condition, i.e., the condition is satisfied when a certain degree of change has been detected. Therefore, four categories of triggers are supported by STING+; Region-trigger: absolute condition on certain regions, Attribute-trigger: absolute condition on certain attributes, Region trigger: relative condition on certain regions, Attribute trigger: relative condition on certain attributes.

2.3.2.3. BIRCH

Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH), is designed for clustering large amount of multidimensional metric data points [31]. It requires only one scan of the entire database and uses only a limited memory. BIRCH uses a hierarchical data structure called a CF-tree, or Clustering-Feature-tree that captures the needed information. A clustering-feature vector CF is a triple that

stores the information maintained about a cluster. The triple CF= {*N*, $\overrightarrow{LS}$, *SS*} contains the number of data points in the cluster, *N*, and $\overrightarrow{LS}$, the linear sum of the *N* data points, i.e.

$$\overrightarrow{LS} = \sum_{i=1}^{N} \overrightarrow{X_i},$$ .....Equation 2-6

and *SS*, the square-Sum of the *N* data points i.e.

$$SS = \sum_{i=1}^{N} \overrightarrow{X_i}^2$$ .....Equation 2-7

A CF-tree is a height balanced tree with a branching factor B. each internal node contains a CF triple for each of its children. Each leaf node also represents a cluster and contains a CF entry for each sub cluster in it. A sub cluster in a leaf node must have a diameter no greater than a given threshold value T.

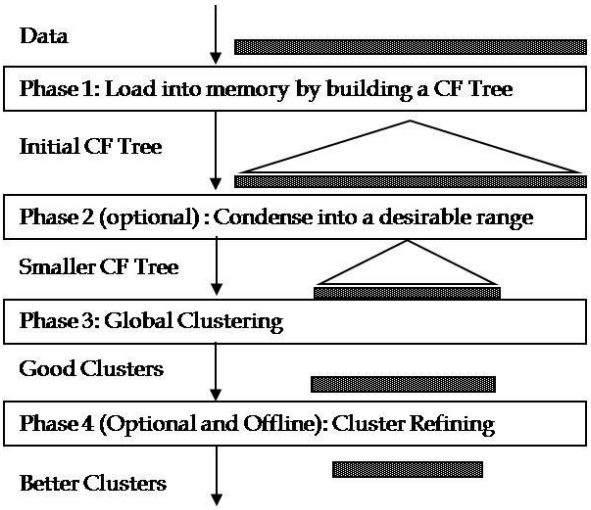


Figure 2-3: Overview of BIRCH

In the pre-clustering phase, the entire database is scanned and an initial in-memory CF-tree is built, representing dense regions of points with compact summaries or sub-clusters in the leaf nodes. Phase 2, rescans the leaf nodes entries to build a smaller CF-tree. It can be used to remove outliers and make larger clusters

from sub-clusters. Phase 3 attempts to compensate for the order-dependent input. It uses either an existing centroid based clustering algorithm, or a modification of an existing algorithm applied to the sub-clusters at the leaves as if these sub-clusters were single points. The pre-clustering algorithm is both incremental and approximate. BIRCH is linear in both space and I/O time. The choice of threshold value is vital to an efficient execution of the algorithm. The worst case complexity of BIRCH can be $O(n^2)$. Figure 1-3 gives an Overview [31].

2.3.3. Grid Based Clustering

2.3.3.1. WAVECLUSTER

WaveCluster is a clustering approach based on wavelet transforms [32]. WaveCluster is based on the representation of spatial object as a feature vector where each element of the vector corresponds to one numerical attribute. These feature vectors of the spatial data can be represented in the spatial area, which is termed feature space, where each dimension of the feature space corresponds to one of the features. For an object with n numerical attributes, the feature vector will be one point in the n-dimensional feature space. The collection of objects in the feature space composes an n-dimensional signal. The high frequency parts of the signal correspond to the regions of the feature space where there is a rapid change in the distribution of objects, that is the boundaries of clusters. The low frequency parts of the n-dimensional signal which have high amplitude correspond to the areas of the feature space where the objects are concentrated, i.e., the clusters themselves. The complexity of generating clusters is O(n) and is not impacted by Outliers. WaveCluster can find arbitrarily shaped clusters and does not need to know the desired number of clusters.

2.3.3.2. BANG

BANG structure adapts to the distribution of items so that the dense areas have larger number of smaller grids and less dense areas have a few large ones [33]. BANG organizes the value space containing the patterns. The patterns are treated as points in a k-dimensional value space and are inserted into the BANG-Structure [34]. These points are stored accordingly to their pattern values preserving the topological distribution. The BANG-structure partitions the value space and administers the points by a set of surrounding rectangular shaped blocks. These blocks are then sorted based on their density, which is the number of items in the grid divided by its area. Based on the number of clusters needed, the grids with the highest density are treated as cluster centre.

2.3.3.3. CLIQUE

CLIQUE, named for Clustering In Quest, the data mining research project at IBM Almaden and is an grid-based approach for high dimensional data sets that provides "automatic sub-space clustering of high dimensional data" [35]. CLIQUE identifies dense clusters in subspaces of maximum dimensionality. It generates cluster descriptions in the form of DNF expressions that are minimized for ease of comprehension. It produces identical results irrespective of the order in which input records are presented and does not presume any specific mathematical form for data distribution. The initial phase of the algorithm partitions the data space S into non-overlapping rectangular units, where $S = S = A_1 \times A_2 \times \ldots \times A_d$ a d-dimensional numerical space.. The units are obtained by partitioning every dimension into ξ intervals of equal length, which is an input parameter. Each unit u is the intersection of one interval from each attribute. The *selectivity* of a unit is defined to be the fraction of total data points contained in the unit. A unit u is dense, if *selectivity(u)* is greater than τ, where the density threshold τ is

another input parameter. Similarly all the units in all subspaces of the original d-dimensional space are defined. A cluster is a maximal set of connected dense units in k-dimensions. Region in k dimensions is an axis-parallel rectangular k-dimensional set. A region can be expressed as a DNF expression on intervals of the domains A_i. A *region* R contained in a cluster C is said to be *maximal* if no proper superset of R is contained in C.A minimal description of a cluster is a non-redundant covering of the cluster with maximal regions.

The running time the algorithm is the exponential in the highest dimensionality of any dense unit. The algorithm makes k passes over the database. Thus the time complexity is $O(c^k + m\ k)$ for a constant c where k is the highest dimensionality of any dense unit and m is the number of input points. This algorithm can be improved by pruning the set of dense units to those that lie in "interesting" subspaces using a method called MDL-based pruning or minimal description length. Subspaces with large coverage of dense units are selected and the remainder is pruned.

2.3.3.4. MOSAIC

MOSAIC greedily merges neighboring clusters maximizing a given fitness function [36]. MOSAIC uses Gabriel graphs to determine which clusters are density- based neighboring and approximates non-convex shapes as the unions of small clusters that have been computed using a representative-based clustering algorithm. The Gabriel graph of a set of points S in the Euclidean plane expresses one notion of proximity or nearness of those points. MOSAIC constructs the Gabriel graph for a given set of representatives, and then uses the Gabriel graph to construct a Boolean merge-candidate relation that describes which of the initial clusters are neighboring. This merge candidate relation is then updated incrementally when clusters are merged.

2.3.4. Related works in hierarchical clustering

Many hybrid algorithms are developed by merging partitional and hierarchical techniques to improve upon the limitations of both the techniques [37]. Survey and evaluation of various hierarchical clustering algorithms are mentioned in [38], [39], [40]. Other significant studies related to the field are [41], [42], [43], [44], [45], [46], [47].

2.4. Density Based Algorithms

Clustering algorithms which form clusters based on the density of data points are termed as density based clustering algorithms. The major advantage of such algorithms is they could find clusters of arbitrary shape. Density-based clustering algorithms are independent of prior knowledge of number of cluster. Such algorithms may be useful in situations where the number of cluster should be determined easily before the start of the algorithm [48].

2.4.1. DBSCAN

Density based Spatial Clustering of Applications with noise (DBSCAN) creates clusters with a minimum size and density [49]. Density is defined as the number of points within a certain distance of each other. The algorithm uses two parameters, *Eps* and *MinPts* to control the density of the cluster. *Minpts*, indicates the minimum number of points in any cluster.

The *Eps-neighborhood* of a point is defined by $N_{Eps}(p) = \{ q \in D \mid dist(p, q) \leq Eps \}$. The distance function *dist(p, q)* determines the shape of the neighborhood. Algorithm DBSAN does not require the desired number of cluster as initial input. Two kinds of points in a cluster are specified in the algorithm, i.e. *core points*; points inside of the cluster and *border points*; points on the border of the cluster. An *Eps-neighborhood* of a border point contains significantly less points than an *Eps-*

neighborhood of a core point. For every point *p* in a cluster *C* there is a point *q* in *C* so that *p* is inside of the Eps-neighborhood of *q* and $N_{Eps}(q)$ contains at least *MinPts* points. A point p is directly density-reachable from a point q wrt. Eps, MinPts if

$$p \in N_{Eps}(q), \left|N_{Eps}(q)\right| \geq Minpts \ (Core\ point\ condition)$$

Directly density-reachable is symmetric for a pair of core points and it is not symmetric if one core point and one border point are involved. A point p is density-reachable from a point q wrt. *Eps* and *MinPts* if there is a chain of points $p1, ..., pn, p_1 = q, p_n = p$ such that p_{i+1} is directly density-reachable from p_i. A point p is density-connected to a point q wrt. *Eps* and *MinPts* if there is a point *o* such that both, *p* and *q* are density-reachable from *o* wrt. *Eps* and *MinPts*. A cluster C wrt. *Eps* and *MinPts* is a non-empty subset of *D*, where *D* is a database of points, satisfies the following conditions:

$\forall p, q$: *if* $p \in C$ and *q* is density-reachable from p wrt. *Eps* and *MinPts*, then $q \in C$. *(Maximality)*

$\forall p, q \in C$: *p* is density-connected to q wrt. EPS and MinPts. *(Connectivity)*

The noise is defined as the set of points in the database *D* not belonging to any cluster *Ci* , i.e. noise = $\{p \in D \mid \forall i: p \notin Ci\}$.

DBSCAN starts with an arbitrary point *p* and retrieves all points density-reachable from *p* wrt. *Eps* and *MinPts*. If *p* is a core point, this procedure yields a cluster wrt. *Eps* and *MinPts*. If *p* is a border point, no points are density-reachable from *p* and DBSCAN visits the next point of the database. Merge two clusters, if two clusters of different density are "close" to each other. The algorithm may need to be called recursively with a higher value for *MinPts* if "close" clusters need to be merged because they are within the same *Eps* threshold. The expected time complexity of DBSCAN is $O(n \log n)$.

2.4.2. GDBSCAN

GDBSCAN - can cluster point objects as well as spatially extended objects according to both, their spatial and their non-spatial attributes [50]. GDBSCAN generalizes DBSCAN in two important ways. Any notion of a neighborhood of an object can be used, if the definition of the neighborhood is based on a binary predicate which is symmetric and reflexive. Instead of simply counting the objects in the neighborhood of an object, other measures can be used for example, considering the non-spatial attributes such as the average income of a city, to define the "cardinality" of that neighborhood.

To find a density-connected set, GDBSCAN starts with an arbitrary object p and retrieves all objects density-reachable from p with respect to *NPred;* neighbourhood of the object and *MinWeight*; minimum weighted cardinality. If p is a core object, this procedure yields a density-connected set with respect to *NPred* and *MinWeight* . If p is not a core object, no objects are density-reachable from p and p is assigned to Noise, where Noise is defined as the set of objects in the database *D* not belonging to any density-connected set *Ci*. This procedure is iteratively applied to each object p which has not yet been classified.

2.4.3. OPTICS

OPTICS (Ordering Points to Identify the Clustering Structure) creates an augmented ordering of the database representing its density-based clustering structure [51]. Let DB be a database containing n points. The OPTICS algorithm generates an ordering of the points $o:\{1..n\} \rightarrow$DATABASE and corresponding reach ability-values $r:\{1..n\} \rightarrow R \geq 0$. OPTICS does not assign cluster memberships. Instead, the algorithm store the order in which the objects are processed and the information which would be used by an extended DBSCAN algorithm to assign cluster memberships. This information consists of only two values for each object: the core-

distance and a reachability distance. The core-distance of an object p is simply the smallest distance ε' between p and an object in its ε-neighborhood such that p would be a core object with respect to ε' if this neighbor is contained in $N_\varepsilon(p)$. Otherwise, the core-distance is *UNDEFINED*. The reachability-distance of an object p with respect to another object o is the smallest distance such that p is directly density-reachable from o if o is a core object. Depending on the size of the database, the cluster-ordering can be represented graphically for small data sets or can be represented using appropriate visualization technique for large data sets.

2.4.4. DBCLASD

Distribution Based Clustering of Large Spatial Databases (DBCLASD) is another locality-based clustering algorithm, but unlike DBSCAN, the algorithm assumes that the points inside each cluster are uniformly distributed [52]. Three parameters are defined in the algorithm; $NN_S(q)$, $NN_{Dist}(q)$, and $NN_{DistSet}(S)$. Let q be a query point and S be a set of points. Then the nearest neighbor of q in S, denoted by $NN_S(q)$, is a point p in $S - \{q\}$ which has the minimum distance to q. The distance from q to its nearest neighbor in S is called the nearest neighbor distance of q, $NN_{Dist}(q)$ for short. Let S be a set of points and e_i be the elements of S. The nearest neighbor distance set of S, denoted by $NN_{DistSet}(S)$, or distance set for short, is the multi-set of all values. The probability distribution of the nearest neighbor distances of a cluster is analyzed based on the assumption that the points inside of a cluster are uniformly distributed, i.e. the points of a cluster are distributed as a homogeneous Poisson point process restricted to a certain part of the data space. A grid-based representation is used to approximate the clusters as part of the probability calculation. DBCLASD is an incremental algorithm. Points are processed based on the points previously seen, without regard for the points yet to come which makes the clusters produced by DBCLASD dependent on input order. The major

advantage of DBCLASD is that it requires no outside input which makes it attractive for larger data sets and sets with larger numbers of attributes.

2.4.5. Related works in density based algorithms

Density based clustering is employed in various domain with either extension or modification in [53] , [54], [55], [56], [57], [58].

2.5. Model based algorithms

Model-based clustering can give a probability distribution over the clusters, in that it not a hard clustering. Expectation Maximization Clustering [59] can be described as generalization of K-Means algorithm. A model connecting the observations to the cluster memberships and parameters is built with the probability

$$\boldsymbol{P(x_k) = \sum_{m=1..m} P(x_k|z_k = m)P(z_k = m)} \text{ ... ...Equation 2-8}$$

$$\boldsymbol{= \sum_{m=1..m} f_m\ (x_k\ \lambda_m)\pi_m} \text{.....Equation 2-9}$$

f_m ( ) is a set of known probability density function parameterized on λ_m and π_m are the weights for the function f_m ( ) . The Expectation Maximization (EM) algorithm finds the values of the parameters λ_m and π_m, by maximizing the likelihood, usually the log of the likelihood of the observations. The E-Step creates an approximate distribution of the missing data, say $u(z)= p(z|x, \theta)$. Let $Q(\theta)$ = *the log likelihood under* $\theta = \{ \lambda_m , \pi_m \}$ averaged by $u(z) = \int \log p(x,z|\theta)\, u(z)\, dz$, the M-Step maximizes $Q(\theta)$ over θ. θ_{new} will the new maximizer. The E-step and M-step are repeated until convergence.

2.6. Other Clustering techniques

2.6.1. Hybrid clustering techniques

2.6.1.1. DENCLUE

DENCLUE (Density basted Clustering) is a generalization of partitioning, locality-based and hierarchical or grid-based clustering approaches [60]. The influence of each data point can be modeled formally using a mathematical function called influence function. This influence function is applied to each data point. The algorithm models the overall point density analytically using the sum of the influence functions of the points. An example influence function can be a Gaussian function

$$f_{Gauss}(x,y) = e^{-\frac{d(z,y)^2}{2\sigma^2}} \quad \text{.....Equation 2-10}$$

The density function which results from a Gaussian influence function is

$$f_{Guass}^{D}(x) = \sum_{i=1}^{N} e^{-\frac{d(z,y)^2}{2\sigma^2}} \quad \text{.....Equation 2-11}$$

Clusters can then be determined mathematically by identifying density attractors. Density attractors are local maxima of the overall density function. These can be either center-defined clusters, similar to k-means clusters, or multi-center-defined clusters, that is a series of center-defined clusters linked by a particular path which identify clusters of arbitrary shape. Clusters of arbitrary shape can also be defined mathematically. The mathematical model requires two parameters, α and ξ. α is a parameter which describes a threshold for the influence of a data point in the data space and $\xi$is a parameter which sets a threshold for determining whether a density-attractor is significant.

The three major advantages for this method of higher-dimensional clustering claimed by the authors are that the algorithm provides a firm mathematical base for finding arbitrary shaped clusters in high-dimensional datasets. Also, result show

good clustering properties in data sets with large amounts of noise and significantly faster than existing algorithms.

2.6.1.2. SParClus

SpaRClus (Spatial Relationship Pattern-Based Hierarchical Clustering) to cluster image data is based on an algorithm, SpIBag (Spatial Item Bag Mining), which discovers frequent spatial patterns in images [61]. SpIBag is invariant on semi-affine transformations. Semi-affine transformation is a way to express or detect shape preserving images. SpaRClus uses internally SpIBag algorithm to mine frequent patterns, and generates a hierarchical structure of image clusters based on their representative frequent patterns.

2.6.1.3. C2P

C2P, Clustering based on Closest Pairs, exploits spatial access methods for the determination of closest pairs [62]. C2P consists of two main phases. The first phase efficiently determines a number of sub-clusters. The first phase of C2P has as input n points and produces m sub-clusters, and it is iterative. The first phase of C2P has the objective of efficiently producing a number of sub-clusters which capture the shape of final clusters. Therefore, it represents clusters with their center points. The second phase uses a different cluster representation scheme to produce the final clustering. The second phase performs the final clustering by using the sub-clusters of the first phase and a different cluster representation scheme. The second phase merges two clusters at each step in order to better control the clustering procedure. The second phase is a specialization of the first, i.e., the latter can be modified in: a) finding different points to represent the cluster instead the center point, b) finding at each iteration only the closest pair of clusters that will be merged, instead of finding for each cluster the one closest to it. The time complexity of $C^2P$for large datasets is O(nlog n), thus it scales well to large inputs.

2.6.1.4. DBRS+

Density-Based Spatial Clustering in the Presence of Obstacles and Facilitators (DBRS+) aims to cluster spatial data in the presence of both obstacles and facilitators [63]. The authors claim that without preprocessing, DBRS+ processes constraints during clustering. It can also find clusters with arbitrary shapes and varying densities. DBRS is a density-based clustering method with three parameters, *Eps*, *MinPts*, and *MinPur*. DBRS repeatedly picks an unclassified point at random and examines its neighborhood, i.e., all points within a radius *Eps* of the chosen point. The purity of the neighborhood is defined as the percentage of the neighbor points with the same non-spatial property as the central point. If the neighborhood is sparsely populated ($\leq$*MinPts*) or the purity of the points in the neighborhood is too low ($\leq$*MinPur*) and disjoint with all known clusters, the point is classified as noise. Otherwise, if any point in the neighborhood is part of a known cluster, this neighborhood is joined to that cluster, i.e., all points in the neighborhood are classified as being part of the known cluster. If neither of these two possibilities applies, a new cluster is begun with this neighborhood. The time complexity of DBRS is $O(n \log n)$ if an R-tree or SR-tree is used to store and retrieve all points in a neighborhood.

2.6.2. Incremental methods

Certain application clusters an incoming document stream and is typical clustering algorithms are not a good choice for these kinds of problems. Charikar et al defines incremental clustering as "for an update sequence of n points in M, maintain a collection of k clusters such that as each one is presented, either it is assigned to one of the current k clusters or it starts off a new cluster while two existing clusters are merged into one" [64].

2.6.3. Ensemble methods

A cluster ensemble consists of different partitions. Such partitions can be obtained from multiple applications of any single algorithm with different initializations, or from the application of different algorithms to the same dataset. Consider a set $S = \{x_1, x_2, ..., x_n\}$ of n points. A clustering ensemble is a collection of m clustering solutions: $C = \{C1, C2, ..., Cm\}$. Each clustering solution C_L for $L = 1,...,m$, is a partition of the set S, i.e. $C_L = \{C_1, C_2 .. C_L^{K_l}\}$ where $U_K C_L^K = S$, where [65]. Given a collection of clustering solutions C and the desired number of clusters k, the objective is to combine the different clustering solutions and compute a new partition of S into k disjoint clusters.

2.6.4. Soft computing methods

Zadeh defined soft computing as "soft computing is not a homogeneous body of concepts and techniques. Rather, it is a partnership of distinct methods that in one way or another conform to its guiding principle." [66]. Recently various soft computing methodologies have been applied to handle the different challenges posed by data mining. The main constituents of soft computing include fuzzy logic, neural networks, genetic algorithms, and rough sets. Each of them contributes a distinct methodology for addressing problems in its domain [67].

2.6.4.1. Fuzzy sets

Fuzzy sets are sets whose elements have degrees of membership. Fuzzy set can be considered as an extension of classic sets. In a classical set, an element may belong to a set or not belong to a set, but cannot partially belong to sets. In fuzzy set theory elements are permitted a gradual assessment of the membership in a set; which is described with the aid of a membership function valued in the real unit

interval [0, 1]. The fuzzy set defined on a universe of discourse helps to represent uncertainties is in definition.

2.6.4.2. Neural network

An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information [68]. It is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems. For classification, using neural networks, a model is constructed. To classify a tuple, the attribute values are input into a directed graph at the corresponding source nodes. The output value generated indicates the probability that the corresponding input tuple belongs to that class.

The neural network approach to clustering tends to represent each cluster as an exemplar [48]. An exemplar acts as a prototype of the cluster. New objects can be distributed to the cluster with most similar exemplar based on some distance measure. Kohnen's Self-organizing Map (SOM) is one among the most popular neural network approach for cluster analysis [69] . Self-organizing Map is an unsupervised learning to produce a low-dimensional (typically two-dimensional), discretized representation of the input space of the training samples, and called a map. If x1 and x2 are two input vectors and t1 and t2 are the locations of the corresponding winning output nodes, then t1 and t2 should be close if x1 and x2 are similar. A network that performs this kind of mapping is called a feature map. As in the case of K-means algorithm, the SOM starts with few nodes and adds more nodes if the given data points are far away from the cluster weights, leading to an existing node.

2.6.4.3. Genetic Algorithms

Genetic Algorithms (GAs) [70] are adaptive heuristic search algorithm based on the evolutionary ideas of natural selection and genetics. As such they represent an intelligent exploitation of a random search used to solve optimization problems. Although randomized, GAs are by no means random, instead they exploit historical information to direct the search into the region of better performance within the search space. A typical genetic algorithm requires a genetic representation of the solution domain and a fitness function to evaluate the solution domain. Once the genetic representation and the fitness function are defined, a GA proceeds to initialize a population of solutions and then to improve it through repetitive application of the mutation, crossover, inversion and selection operators. Initially many individual solutions are randomly generated to form an initial population. During each successive generation, a proportion of the existing population is selected through a fitness-based process. The next step is to generate a second generation population of solutions from those selected through genetic operators: crossover and/or mutation. This generational process is repeated until a termination condition has been reached.

2.6.4.4. Rough Sets

A rough set, first described by a Polish computer scientist Zdzisław I. Pawlak, is a formal approximation of a crisp set in terms of a pair of sets which give the lower and the upper approximation of the original set. The lower approximation consists of all objects which surely belong to the set and the upper approximation contains all objects which possible belong to the set. The difference between the upper and the lower approximation constitutes the boundary region of the rough set. Lower and Upper approximations are two basic operations used in rough set theory.

2.7. Related studies on Information visualization

A large number of information visualization techniques have been developed over the last decade to support the exploration of large data sets. Visualization can produce qualitative overview, of large and complex data sets, can summarize data, and can assist in identifying regions of interest, and appropriate parameters for more focused quantitative analysis [71]. The visualization techniques available are either basic or limited to particular domains of interest. They are used in an analytical way and require affirmation by other data mining techniques to be certain about what is revealed.

Visualization techniques are classified on the basis of the purpose of visualization, the underlying data and the dimension of display. The work of Tufte [72], [73] provides a broad review of some of the better approaches and examples. Various geometrical techniques like scatter plot matrix [74], parallel coordinates [75], survey plots [76], icon based techniques like chernoff faces [77] , stick figures [78] and pixel oriented techniques [79] are popular visualization methods. Various visualization techniques are compared and reviewed in [80], [81], [82]. The figure 2-4 shows a periodic table of information visualization referred from www.visual-literacy.org

The major challenges faced by the current methods are the usability and scalability of algorithms. Another fact which is to be pointed out is knowledge visualization is different from fact visualization. Often the current techniques facilitate the representation of data in a visual manner, but impede from depicting the knowledge inferred as a result of the data mining process. The challenges and issues of this area are discussed in [83], [84]. A periodic table of visualization techniques is presented in figure 2-4. (Courtesy: http://www.visual-literacy.org/)

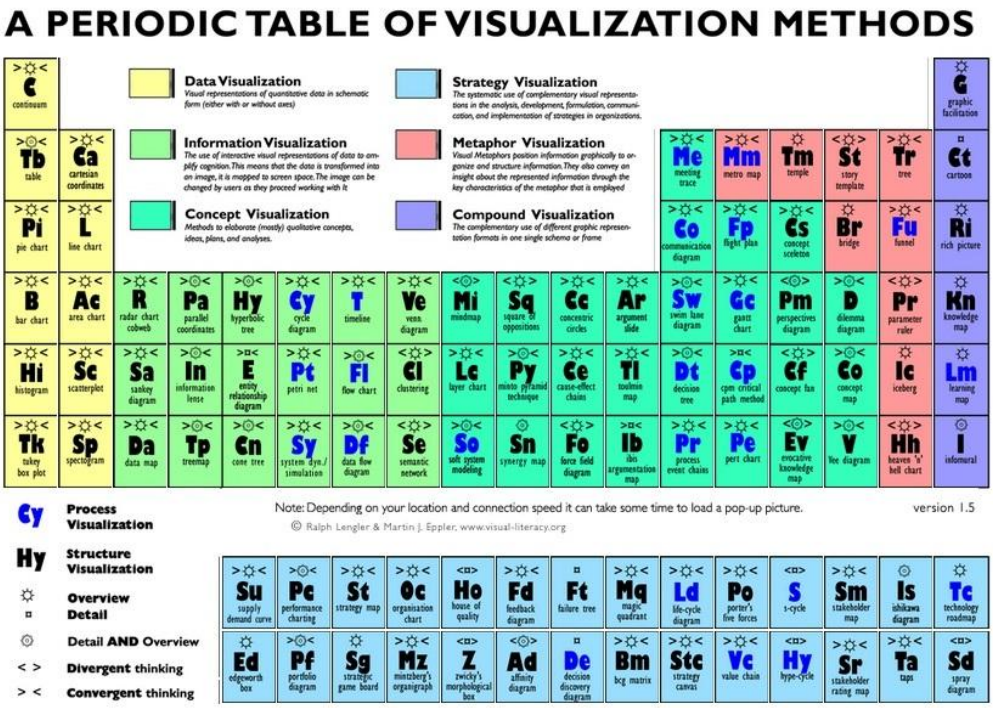


Figure 2-4: Periodic Table of Information Visualization

2.8. Summary of the literature review

2.8.1. Clustering

A good cluster should exhibit the property of external isolation and internal cohesion [85]. External isolation requires that entities in one cluster should be separated from entities in another cluster by a fair distance. Internal cohesion requires that entities within the same cluster should be similar to each other, at least within the local metric. Sneath and Sokal have described a number of properties of a cluster, the most important of which are density, variance, dimension, shape and separation [86], [87].

Density is a property of a cluster that defines it as a relatively thick swarm of data points in a space when compared to other areas of the space that may have

comparatively few or no points. Variance is the degree of dispersion of the points in this space from the center of the cluster. Clusters can be said to be "tight" when all data points are near the centroid or they may be "loose" when the data points are dispersed from the center. Dimension is a property closely related to that of variance. If a cluster can be identified, it is then possible to measure its "radius." Shape is simply the arrangement of points in the space. While the typical conception of the shape of clusters is that they are hyper spheres or ellipsoids, many different kinds of shapes, such as elongated clusters, are possible. Separation is the degree to which clusters overlap or lie apart in the space. Taken together, these terms can be used to describe any type of clusters within a space.

2.8.2. Evaluation of Clusters

Validation of the results from clustering methods is one of the most important issues in cluster analysis. The cophenetic correlation was first proposed by Sokal and Robill and is the major validation measure advocated by the numerical taxonomists. This measure is appropriate only when a hierarchical agglomerative method of clustering is used. The cophenetic correlation is used to determine how well the tree or dendrogram resulting from a hierarchical method actually represents the pattern of similarities or dissimilarities among the entities. The magnitude of this value should be very close to 1 for a high-quality solution. Another validation measure is a multivariate analysis of variance (MANOVA) of the variables in order to test for the significance of the clusters.

Silhouette plot is another method of validation of clusters. The silhouette is a measure of how much closer points in the cluster are to one another compared to points outside the cluster. Let any clustering algorithm has grouped a dataset into k clusters. For each data item *i*, let *a(i)* be the average dissimilarity of *i* with all other data item within the same clusters. The measure of *a(i)* can be interpreted as how

well the item *i* is assigned to the cluster it belongs to. The smaller the value, the better will be the assignment. The average dissimilarity of *i* with every other items in other clusters are also measured. Let *b(i)* denote the lowest average dissimilarity to any other cluster. Now silhouette value *s(i)* can be defined as

$$\boldsymbol{s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}} \quad \text{.....Equation 2-12}$$

i.e.
$$s(i) = \begin{cases} 1 - \frac{a(i)}{b(i)}, if\ a(i) < b(i) \\ 0, if\ a(i) = b(i) \\ \frac{b(i)}{a(i)} - 1, if\ b(i) > a(i) \end{cases}$$

From the above definition, it is clear that $-1 \leq s(i) \leq 1$. Values near 1 mean that the observation is well placed in its cluster; values near 0 mean that it's likely that an observation might really belong in some other cluster. Within each cluster, the value for this measure is displayed from smallest to largest. If the silhouette plot shows values close to one for each observation, the fit was good; if there are many observations closer to zero, it's an indication that the fit was not good. The silhouette plot is very useful in locating groups in a cluster analysis that may not be doing a good job; in turn this information can be used to help select the proper number of clusters. A sample plot is given below in Figure 2-5.

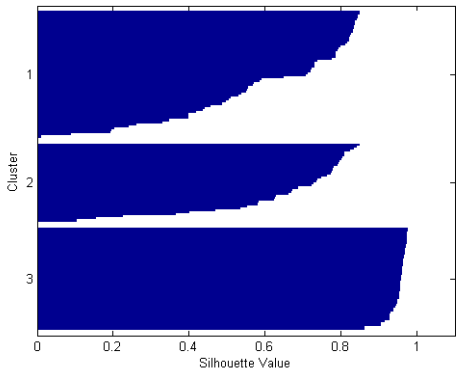


Figure 2-5: Silhouette plot

A problem common to all clustering techniques is the difficulty of deciding the number of clusters present in the data. No algorithm can be chosen as perfect clustering process as they suffer from limitations like scaling, prior knowledge of domain, dimensionality or types of attributes. A problem common to all clustering techniques is the difficulty of deciding the number of clusters present in the data. Certain hierarchical clustering techniques give rise to a property called chaining, which refers to the tendency of the method to cluster together at a relatively low level objects linked by chains of intermediates. This is a limitation of hierarchical clustering techniques. There is no provision for reallocation of entities which may have been poorly classified at an early stage in the analysis [88]. The density based methods such as the fitting of mixtures of multivariate normal distributions also suffer from the problem of sub-optimal solutions, since there may be more than one solution to the maximum likelihood equations. Other density-seeking techniques are dependent on parameters arbitrarily chosen by the investigator.

The survey on clustering brought out the fact that no clustering algorithm can be explicitly assigned for mixed category data set. Also none of the classical clustering algorithm works well on uniform dataset. For a mixed attributed dataset, the variables have to be converted into a numerical format either by some encoding scheme or using similarity measures. It is required to explore specific techniques to deal with mixed attributed data.

2.8.3. Information Visualization

The review of literatures regarding information visualization showed that most of the existing methods are distinct in some respect from each other and have also gained research attention in the published literature. Many visualization tools like scatter plot matrix envisage data in either two or three dimensions and are flawed for representing higher dimensional relationships. Parallel co-ordinates can represent

any number of dimensions in a single display so higher dimensional features in the data set can be identified in a single display. The user must learn to interpret and manipulate the display. If the number of data instances is large the display can become overcrowded and features of the data set are obscured. Pixel oriented techniques are able to represent a large number of instances in a single display by using a single pixel for each data instance. All the variations of the technique are novel and require the user to learn to interpret the display. They are difficult to interpret and ambiguous in some cases. Icon based approaches representing each instance takes up considerable space on the display. A higher dimensional display of data requires a sophisticated hardware and software to implement.

Chapter 3

Preprocessing the Mixed Attributed Dataset

.....

The chapter explores the various techniques to preprocess a mixed attributed dataset so as to prepare for applying the common clustering algorithms. An algorithm is also suggested to convert a fully categorical dataset into a numerical equivalent, as an extension to an existing algorithm.

.....

3. Preprocessing the Mixed Attributed Dataset

3.1. Introduction to the chapter

The survey of literature on clustering algorithms indicates that the output of any clustering process solely depends on the quality of input data. Real world data can be incomplete, noisy, or inconsistent. Tasks like data cleaning, data integration, transformation, reduction and discretization are applied to the raw data to make it normalized and ~~a~~menable for further processing

3.2. Dataset

The input to a machine learning scheme is a set of instances, which are to be classified or associated or clustered. These instances, alternately coined as datasets are characterized by the values of a set of predetermined attributes. A dataset is a collection of data, usually presented in tabular form. The term data set also refers to a file that contains one or more records. The record is the basic unit of information, where each column represents a particular variable and each row corresponds to a given member of the data set in question. The record lists the values for each of the variables and each value is known as a datum. In simplest terms, a record is a fixed number of bytes containing data. Often, a record collects related information that we treat as a unit, such as one item in a database or personnel data about one member of a department. The term field or attribute refers to a specific portion of a record used for a particular category of data, such as an employee's name or department.

The data set may comprise data for one or more members, corresponding to the number of rows. Non-tabular data sets can take the form of marked up strings of characters, such as an XML file. Data sets can hold information such as medical records or insurance records that are used by a program running on the system. Data

sets are also used to store information needed by applications or the operating system itself, such as source programs, macro libraries, or system variables or parameters [89].

Datasets consist of all of the information gathered during a requirement analysis. Learning how to interpret the results is a key component to the data mining process. A dataset suitable for clustering is a collection of points, which are objects belonging to some space. In most general sense, a space is just a universal set of points, from which the points in the dataset are drawn. Data sets can be cataloged, which permits the data set to be referred to by name without specifying where it is stored.

3.2.1. Structured and Semi-Structured data

With the growing use of computers, a great amount of data is being generated and digitally stored by systems like multimedia based environments or Internet infrastructures. To prepare adequate data mining methods, these data has to be analyzed for its basic data types and characteristics. The first step in the analysis is systemization of data with respect to their computer representation and use. The source data can be classified as structured, semi-structured or unstructured. Most databases contain structured data, consisting of well-defined fields with numeric or alphanumeric values, while scientific data may contain all the types of data. Structured data is often referred as traditional data, while semi-structured data is termed as non-traditional or multimedia data.

The structured data is a collection of cases with potential measurements called features are specified and are uniformly measured over many cases. The structured data is represented in a tabular form or in a relational form where columns represent the features and rows represent entities. The value of an attribute for a particular instance is a measurement of the quantity to which the attribute

refers. Data can be quantitative or descriptive. A description of the attributes is given below [48].

3.2.1.1. Numeric attribute

Quantitative data is a numerical measurement and are called numeric attributes. Numeric attributes measure numbers either real or integer valued. Typical examples come from employee records in an organization, data collected from experiments like atmospheric, seismic exploration and share market data. Numerical data will be within some bounds and any strong deviation from the boundary is referred to as an outlier.

3.2.1.2. Categorical attribute

Descriptive data is a categorical measurement expressed by means of a natural language description and is called categorical data. A categorical variable is a generalization of the binary variable in that it can take on more than two states. A categorical attribute can be a nominal, ordinal, interval or ratio type based on the levels of measurement.

3.2.1.3. Nominal attributes

Nominal values are values that are distinct symbols identifying unique entities. There is no sense to add or multiply or even compare the nominal values. A nominal attribute can only be tested for equality or inequality like *overcast = "sunny"*. The attributes are compared on the basis of dissimilarity. The dissimilarity between two variables i and j can be computed based on the ratio of mismatches;

$$\boldsymbol{d(i,j) = \frac{p-m}{p}} \quad \text{.....Equation 3-1}$$

i.e., the number of variables for which i and j are in the same state where m is the number of matches and p is the total number of variables in the set.

3.2.1.4. Interval attributes

Interval scaled variables are continuous measurements of a roughly linear scale. Interval values are ordinal values in which it is possible to determine a distance between the ordered categories. Temperature scale is an example for interval variables, where the difference between 46 degree Celsius and 48 degree Celsius can be clearly expressed. To standardize the measurement of the variable *f*, we can compute the mean absolute deviation:

$$s_f = \frac{1}{n}(|x_{1f} - m_f| + |x_{2f} - m_f| + \cdots + |x_{nf} - m_f|) \quad \text{.....Equation 3-2}$$

where $x_{1f} \ldots . x_{nf}$ are the *n* measurements of *f* and m_f is the mean value of *f*, i.e. $m_f = \frac{1}{n}(x_{1f} + x_{2f} + \cdots + x_{nf})$. The standardized measurement z-score is calculated by

$$z_{if} = \frac{x_{if} - M_{if}}{s_f} \quad \text{.....Equation 3-3}$$

After standardization, or without standardization in certain applications, the dissimilarity between the objects described by interval-scaled variables is typically computed based on the distance between each pair of objects. Any distance measure like Euclidean or Manhattan can be used.

3.2.1.5. Ordinal attributes

A discrete ordinal variable resembles a categorical variable, except that the M states of the ordinal value are ordered in a meaningful sequence. Ordinal attributes are rationally listed in some order. It is possible to rank the order of the values. Even though there is a notion of ordering, there is no notion of distance. For example *hot* > *mild*> *cool* makes sense and by convention, it means that mild lies between hot and cool. If f is a variable from a set of ordinal variables describing n objects, the dissimilarity computation with respect to *f* involves the following steps

The value of *f* for the i^{th} object is x_{if} and has *f* has M_f order states, representing the ranking *1,….*M_f Replace each x_{if} by its corresponding rank, $r_{if} \in \{1, \ldots . M_f\}$. to map the range of each variable into [0.0,0.1] the rank r_{if} of the i^{th} object in the f^{th} variable is replaced by

$$z_{if} = \frac{r_{if}-1}{M_f-1} \quad \text{.....Equation 3-4}$$

3.2.1.6. Ratio attributes

Ratio attributes are measurement scheme which inherently defines a zero point. A ratio scaled variable makes a positive measurement on a non-linear scale, such as an exponential scale following the formula Ae^{Bt} or Ae^{-Bt} where *A* and *B* are positive constants and *t* represents time. To compute the dissimilarity between ratio-scaled variables, logarithmic transformation of variable *f* having x_{if} for object i by using the formula $y_{if} = \log(x_{if})$. The y_{if} can be treated as interval valued as described above. For example when measuring distance between two objects, distance between the object to itself is zero. Ratio quantities are treated as real numbers and any mathematical operations can be defined on them.

3.2.1.7. Computing dissimilarity of variables of mixed types

Suppose the data set contains *p* variables of mixed type. The dissimilarity *d(i,j)* between objects *i* and *j* is defined as

$$d(i,j) = \frac{\sum_{f=1}^{p} \delta_{i,j}^{(f)} d_{i,j}^{(f)}}{\sum_{f=1}^{p} \delta_{i,j}^{(f)}} \text{.....Equation 3-5}$$

where the indicator $\delta_{i,j}^{(f)} = \begin{cases} 0, x_{if} = 0/missing \\ 1, otherwise \end{cases}$. The contribution of variable *f* to the dissimilarity between *i* and *j,* is dependent on its type.

If *f* is interval-based; $d_{ij}^{(f)} = \frac{|x_{if} - x_{jf}|}{max_h x_{hf} - min_h x_{hf}}$, where h runs over all non-missing objects for variable f,

If *f* is binary or categorical, $d_{ij}^{(f)} = 0$, if $x_{if} = x_{if}$,otherwise $d_{ij}^{(f)} = 1$.

If *f* is ordinal: compute the ranks r_{if} and $z_{if} = \frac{r_{if}-1}{M_f-1}$, and treat as z_{if} as interval-scaled.

If *f* is ratio-scaled: perform logarithmic transformation and treat the transformed data as interval-scaled; or treat *f* as continuous ordinal data, compute r_{if} and z_{if}, and then treat z_{if} as interval-scaled.

3.3. Normalization of numeric data set

In the overall knowledge discovery process, data preprocessing plays a vital role. As far as mining algorithms are concerned, they do not distinguish between the types or magnitude of variables. Though the algorithms do not differentiate the scale of variables, the results will be strongly influenced by it. Normalizing the data will remove this bias when dealing with parameters of different units and scales. If data is considered to be a vector, normalizing is the process by which the vector takes up a unit norm. If data are though as random variables, normalization transforms data to a normal distribution. The well known normalization techniques are described below [48].

3.3.1. Min-Max normalization

Min-Max normalization performs a linear transformation on the original data. Min-Max normalization maps a value *v* to *v'* in the range *[new_min$_A$, mew_max$_A$]* by computing

$$v' = \frac{v - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A$$ .....Equation 3-6

where min_A and max_A are the minimum and maximum value of the attribute respectively.

3.3.2. Z-score normalization

In z-score normalization, the attribute values are normalized based on the mean and the standard deviation of values. A value v is normalized to v' by computing

$$v' = \frac{v - \mu}{\sigma}$$ .....Equation 3-7

where μ and σ are the mean and the standard deviation of the attribute values respectively.

3.4. Transforming a mixed dataset into a numeric dataset

Many of the data mining algorithms work effectively either on pure numeric data or on pure categorical data, most of them perform poorly on mixed categorical and numeric data types. But the real world data consists of both categorical and numerical data. In order to apply a data mining algorithm, either we need to convert the entire database to categorical or numerical in type. The methods used to compute the dissimilarity as discussed in Sec 3.2 designates the attribute to specific type, before computing the dissimilarity. However, practical data records suffer from the problem of multiples types of attributes and wide variation in the connotation of each type. Accordingly algorithm that addresses the attributes as a whole for computing the dissimilarity based on the co-occurrence matrices has been proposed in [90].

3.4.1. Processing of Categorical attributes in Mixed-attributed dataset

Categorical data set is stored in a n * p matrix, where n is the number of observations and p the number of categorical variables. The sample space consists of all possible combinations generated by p variables. The sample space is discrete and has no natural origin. Since a large number of data mining techniques like k-means algorithms focus on numeric data, conversion of a categorical or nominal attribute to a numeric attribute is a basic requirement in data mining. Usual methodology adopted is to convert categorical values into a scale, for example values like "low", "medium" and "high" to numeric scale. But categorical attributes like geographical area or weapons used cannot be ordered naturally. A possible solution for converting the categorical attribute to a numeric value is to explore the relationship among the attributes present in the training dataset. This study adopts the idea finding similarity among the categorical attributes based on the notion of co-occurrence.

The assumption of co-occurrence is based on the fact that that if two values always shows up in one object together, then there will be a strong similarity between them. The values with stronger similarity can be assigned closer numeric values. The first step of the pre-processing of attributes is normalizing the numeric attributes of the dataset into a range [0 1]. This is done to avoid the dominance of attributes with larger values over the lesser valued attributes. Next step is to select a categorical attribute with most number of values, as a base attribute. The values appearing in the base attribute are selected as base items. This method ensures the mapping of a non-base item to multiple base items. A co-occurrence matrix M is constructed with n columns and n rows, where n is the number of total categorical items; m_{ij} represents the co-occurrence between item i and item j ; m_{ii} represents the appearance of item i. The similarity matrix D can be constructed by adopting the formula

$$D_{xy} = \frac{|m(X,Y)|}{|m(X)| + |m(Y)| - |m(X,Y)|} \quad \text{..... Equation 3-8}$$

where *X* represents the event that the item *x* appears in the set of objects; Y represents the event that item *y* appears in the set of objects; *m(X)* is the set of objects containing the item *x*; *m(X, Y)* is the set of objects containing both *x* and *y*. Finally, the matrix D represents the similarity between the categorical items; higher the value, higher the similarity.

The second phase of the algorithm proceeds with finding a numeric attribute in the same instance, which minimizes the within group variance to base attribute. The within group variance can be found out by applying the following formula

$$SS_W = \sum_j \sum_i \left(X_{ij} - \bar{X}_J\right)^2 \quad \text{..... Equation 3-9}$$

where $\bar{X}_J$ is the mean of mapping numeric attribute of j^{th} base item and X_{ij} is the i^{th} value in mapping numeric attribute of j^{th} base item. Every base item is quantified by assigning mean of the mapping value in the selected numeric attribute. All non-base items of the categorical type can be quantified by applying the following formula

$$F(x) = \sum_{i=1}^{d} a_i * v_i \quad \text{..... Equation 3-10}$$

where *d* is the number of base item; a_i is the similarity between item *x* and i^{th} base item taken from D_{xy} (Equation 3.8) ; v_i is the quantified value of i^{th} base item. Thus all the attributes in the dataset is assigned a numeric value.

3.4.2. Extension of algorithm to entirely categorical dataset

The second phase of the above algorithm is not practical, when it comes to an entirely categorical dataset, as there is no numerical attribute to map each base item to a numerical value, unlike so assumed in [90]. A modification to the second phase of the above algorithm is proposed to extend to an entirely categorical dataset. Since there is no numerical attribute, we add a temporary variable F to the dataset. An object *i* in the temporary attribute F, will represent the frequency of base item *i* in the dataset. i.e *freq(i)=number of occurrences of base item i in the entire instances.*. The

frequency indicates the strength of presence of each base item and a higher value indicates a base item with high number of occurrences. For example consider the dataset given in the Table 3-1.

Attr A	Attr B	Attr C	Attr D
A1	B1	C1	D1
A1	B1	C2	D2
A1	B2	C1	D2
A1	B2	C2	D2
A2	B3	C3	D1
A2	B4	C3	D2
A2	B5	C3	D1

Table 3-1: Sample Dataset

Here the Attribute B is chosen as the base attribute, since Attribute B has the maximum number of items. The base items will be B1, B2, B3, B4, and B5. A new attribute F is temporarily added to the dataset with the frequency of base items. Computing the frequency of each base item, we get F(B1)= 2,F(B2)=2,F(B3)=1,F(B4)=1,F(B5)=1.

The new dataset is given below.

Attr A	Attr B	Attr C	Attr D	F
A1	B1	C1	D1	2
A1	B1	C2	D2	2
A1	B2	C1	D2	2
A1	B2	C2	D2	2
A2	B3	C3	D1	1
A2	B4	C3	D2	1

A2	B5	C3	D1	1

Table 3-2: Sample Data appended with the temporary attribute *freq (i)*

Each base item can now be quantified by assigning the mean of the values corresponding to the attribute F. All non- base items can be assigned a numerical equivalent using the equation 3-7.

In order to demonstrate the efficacy of converting mixed attribute values to numeric attributes and clustering the same, a crime data set consisting of more than 600 records is examined. The mixed attribute data is converted in totally numeric attributed data using the algorithm suggested in [90]. Agglomerative and the k mean clustering algorithm are applied in combination. Agglomerative clustering produces the dendrogram. The k clustering centers are selected from the dendrogram to facilitate the k-means clustering. The experimentation thus makes use of the advantages of both agglomerative and k-means clustering algorithms. The idea is demonstrated in Section 4.4.5 for data corresponding to breast cancer.

3.5. Experimentation with Crime dataset

Crime is defined as "an act or the commission of an act that is forbidden, or the omission of a duty that is commanded by a public law and that makes the offender liable to punishment by that law" (Webster Dictionary). Crime detection is an area of vital importance in police department. Various kinds of narrative reports and criminal records are kept by the department of law enforcement and the analysis of this huge database opens up the possibilities of identifying common behavior patterns hidden in it. Crime rate are rapidly changing and improved analysis enables discerning hidden pattern of crime, if any, without any explicit prior knowledge of these pattern. Detecting crime from data analysis can be difficult because the police records exist in various formats and the quality of analysis greatly depends on the background knowledge of the analyst. Police department

has large collection of information recorded by the officers at the time of specific incidents.

Data mining is a powerful technique with great potential to help criminal investigators focus on the most important information in their crime data. In this background, the study is planned as per the given objectives 1) the extraction of crime pattern by analysis of available spatial data and attribute statistics, 2) prediction of a crime based on the spatial distribution of existing data and anticipation of crime rate, 3) detection of a crime, 4) to discern trends to identify analytical solution for police which can routinely be used to associate between types of incident, location, time and descriptive details of the incident.

3.5.1. Investigation process

Indian Penal Code (IPC) is the main criminal code of India. It is a comprehensive code, intended to cover all substantive aspects of criminal law. Every person shall be liable punishment under this Code and not otherwise for every act or omission contrary to the provisions thereof, of which, he shall be guilty within India (IPC Section 2). Each state and union territory of India has its own separate police force. Article 246 of the Constitution of India designates the police as a state subject, which means that the state governments frame the rules and regulations that govern each police force. These rules and regulations are contained in the police manuals of each state force. A First Information Report (FIR) is a written document prepared by police, when they receive information about the commission of a cognizable offence. When a crime has happened, investigation officers start their investigation by making a first investigation report (FIR). An FIR is an important document because it sets the process of criminal justice in motion. It is only after the FIR is registered in the police station that the police take up investigation of the case.

Some terminologies are described below which is used in criminal justice. Suspect refers to the person or persons that are believed to have committed the crime. The suspect may be identified or unidentified. The suspect is not a criminal until proved guilty. The victim is the person who is the target of the crime. Most of the time the victim is identifiable and in most cases is the person reporting the crime. Additionally, the crime may have some witnesses. Crime includes homicides, robbery, cheating, or any act violating the law of land. The police department use electronic systems for crime reporting that have replaced the traditional paper-based crime reports. These crime reports have the different kinds of information categories like type of crime, date/time, location and information about the suspect (identified or unidentified), victim and the witness. Additionally, there is the description of the crime and Modus Operandi (MO) that is usually in the text form.

3.5.2. Clustering process

Clustering is a method to group data into classes with identical characteristics in which the similarity of intra-class is maximized or minimized. Cluster analysis can provide a foundation for predictive modeling, since crimes vary in nature widely. An investigator may use this technique to identify suspects who conduct crimes in similar ways or discriminate among groups belonging to different gangs. Also, the clusters can be used to evolve new patterns and hence may lead to some direction for the unsolved crimes. Some of these clusters will useful for identifying a crimes committed by one or same group of suspects. Some crime may recurrently happen in a specific geographical area. The cluster analysis brings out the possible correlations between the crime type and crime location, criminal age and type of crime, motivation and the area of incident and so on.

3.5.3. Hierarchical Agglomerative Clustering

Hierarchical clustering is a traditional clustering method that generates a tree structure, or a dendrogram, in the clustering process, and there are top-down and bottom-up clustering approaches. Given a set of *N* items to be clustered, and an *N***N* distance (or similarity) matrix, the basic process of hierarchical clustering introduced in 1967 by S.C. Johnson is given as following:

1. Start by assigning each item to a cluster, so that for *N* items, we have *N* clusters, each containing just one item.
2. Find the closest pair of clusters and merge them into a single cluster
3. Compute distances between the new cluster and each of the old clusters.
4. Repeat steps 2 and 3 until all items are clustered into a single cluster of size *N*

To compute the step 3, any of the following methods can be adopted:

1. Single-linkage: shortest distance from any member of one cluster to any member of the other cluster.
2. Complete-linkage clustering: the greatest distance from any member of one cluster to any member of the other cluster.
3. Average-linkage clustering: the average distance from any member of one cluster to any member of the other cluster.
4. Centroid method the centroid (mean value for each variable) of each cluster is calculated and the distance between centroids is used. Clusters whose centroids are closest together are merged.
5. Ward's method: all possible pairs of clusters are combined and the sum of the squared distances within each cluster is calculated. This is then summed over all clusters. The combination that gives the lowest sum of squares is chosen.

3.5.3.1. Limitations of Hierarchical clustering algorithm

The primary disadvantage is that there is no explicit partition into clusters. Also hierarchical clustering is only effective at splitting data small amounts of data. It is difficult to make sense of the data just by looking at the tree, due to the size of the data selecting the optimum number of clusters.

3.5.3.2. Dendrogram

Dendrogram illustrates the clusters which can be joined at each stage of the analysis and the distance between clusters at the time of joining. If there is a large jump in the distance between clusters from one stage to another then this suggests that at one stage clusters that are relatively close together can be joined whereas, at the following stage, the clusters that can be joined which are relatively far apart. The optimum of clusters may be the number present just before that large jump in distance.

3.5.4. K-Means Clustering

K-Means is a crispy partitioning algorithm which takes numeric data as input and outputs clusters with no elements intersecting each other [91]. K-Means starts with selecting either first k elements or specifically chosen k objects as the cluster centroids. The algorithm proceeds by iteratively assigning each object to the closest cluster based on distance to the mean of the cluster using any distance measure. The cluster's mean is then recomputed and the process begins again. The iteration continues until all the objects are assigned to some clusters.

1. The algorithm arbitrarily selects k points as the initial cluster centers
2. Each point in the dataset is assigned to the closed cluster, based upon the Euclidean distance between each point and each cluster center.
3. Each cluster center is recomputed as the average of the points in that cluster.

4. Steps 2 and 3 repeat until the clusters converge. Convergence may be defined differently depending upon the implementation, but it normally means that either no observations change clusters when steps 2 and 3 are repeated or that the changes do not make a material difference in the definition of the clusters

3.5.4.1. Distance Measure

To compute the distance between two data objects, different distance measures are available. Three of the prevalent measures are given below.

Euclidian Distance: $D(x,y) = \sqrt{\sum(x_i - y_i)^2}$.....Equation 3-11

Manhattan/City blocks measure: $D(x,y) = \sum|x_i - y_i|$.....Equation 3-12

Pearson Correlation Coefficient: $D(x,y) = \frac{\sum(x-\bar{x})\sum(y-\bar{y})}{\sqrt{\sum(x-\bar{x})^2}\sqrt{\sum(y-\bar{y})^2}}$.....Equation 3-13

3.5.4.2. Limitations in K-means algorithm

Though K-means is one among the most popular clustering algorithm, it suffers from certain limitations. The user has to choose the value of K, the number of clusters as there are usually no clues as to what number of clusters might be appropriate. Different values of K can bring different outcomes and hence it is difficult to compare the quality of the clusters. Empty clusters can be obtained if no points are allocated to a cluster during the assignment step.

3.5.5. Crime Dataset used for study

Non spatial data / Attribute information from police records can be broadly classified as details of Crime, Evidence(s), Victim(s), Suspect(s), and Witness(s).The Crime data includes a unique crime-id, crime name, the geographical area and location of the incident, date and time of the crime , the weapons used, if any and the type of crime. Different types of crime are Murder, theft, auto-theft, Fraud,

money laundering, cyber-crime etc. The crime dataset is generated after studying a set of available crime records under the attributes crime id, victim area, victim gender, victim age, Crime location, weapons used, motivation for the crime and the identified criminal name. All the attributes except victim age are categorical in nature. The steps in the study for evolving some patterns from the crime dataset in planned into 3 phases.

3.5.5.1. Finding numeric equivalent for categorical attribute

In phase one, the categorical attributes are converted to numerical equivalent based on the algorithm explained in the section 3.3. The dataset consisted of 8 categorical attributes out of which the crime id was exempted from further clustering process. Min-max normalization is applied to the only numerical attribute in the dataset, victim's age, to fall within a range of [0, 1]. The dataset, Crime area is chosen as the base attribute and the values in crime area as base items, since this attribute has got the maximum values. A co-occurrence matrix *M* and subsequently *D*, the similarity matrix is constructed from the original dataset by equation 3.5.The attribute victim age was selected as the numeric attribute with minimum group variance to the base items. Successively, by equation 3.7, entire categorical attributes were converted to its numerical equivalents.

3.5.5.2. Application of Hierarchical Agglomerative Clustering

In order to recuperate the limitations of both k-means and the hierarchical agglomerative clustering algorithm, it is decided to combine the good aspects of these two algorithms. The proposed method is given below

1. Apply hierarchical agglomerative clustering on the crime dataset
2. Find the number of clusters obtained from single linkage output. The cluster height cut off is selected based on the length of the longest branches of dendrogram. The height of each node in the dendrogram is proportional to the

value of the intergroup dissimilarity between its two sub-trees, assuming the nodes representing individual observations are all plotted at zero height.

3. The number of optimal clusters obtained from hierarchical clustering is taken as the optimal value for k in K-Means algorithm.

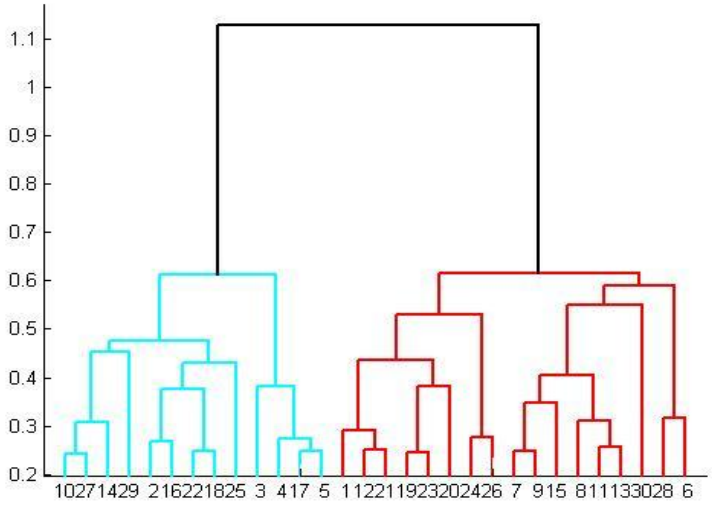


Figure 3-1: Dendrogram of single linkage Clustering

From the above diagram, it can be seen that the average number of clusters from single linkage agglomerative algorithms is 2; hence the value for K as the input parameter for K-Means is set as 2.

3.5.5.3. Evaluation of Dendrogram

Cophenetic coefficient [92] is calculated to measure correlation between the input similarities and the output dendrogram. A value which is closer to 1 indicates a good clustering. The equation for cophenetic coefficient is given below

$$C = \frac{\sum_{i<j}(Y_{ij}-y)(Z_{ij}-z)}{\sqrt{\sum_{i<j}(Y_{ij}-y)^2 \sum_{i<j}(Z_{ij}-z)^2}} \quad \text{.....Equation 3-14}$$

where Z is the output of linkage function and Y is the distance matrix of the observations. The cophenetic coefficient for the dendrogram given in Fig 3.1 is c=0.92 which indicates significant clustering. The cophenetic coefficient value of the dendrogram in Figure 3.1 is 0.8768; a value closer to 1, which shows a good quality agglomerative cluster formation

3.5.5.4. K-means clustering on the crime dataset

K-means clustering is applied on the crime dataset with the parameter k as 2. The scatter plot of crime dataset after K-mean clustering with K = 2 is given below in the Figure 3-2. (For the legibility of the figure, certain attributes are omitted). The following conclusions are drawn from Figure 3.2, 3.3 (a) and (b)

1. The main two clusters broadly indicate crime locations
2. In most of the victim areas the same type weapon has been used to inflict the injury
3. The crime is wide spread. However the victims are not from the same location; they are from all over.
4. In one cluster spread over a certain number of locations, the female victims are more than male. (Figure 3.3)

In both the clusters, motivation seems to be uniformly spread.

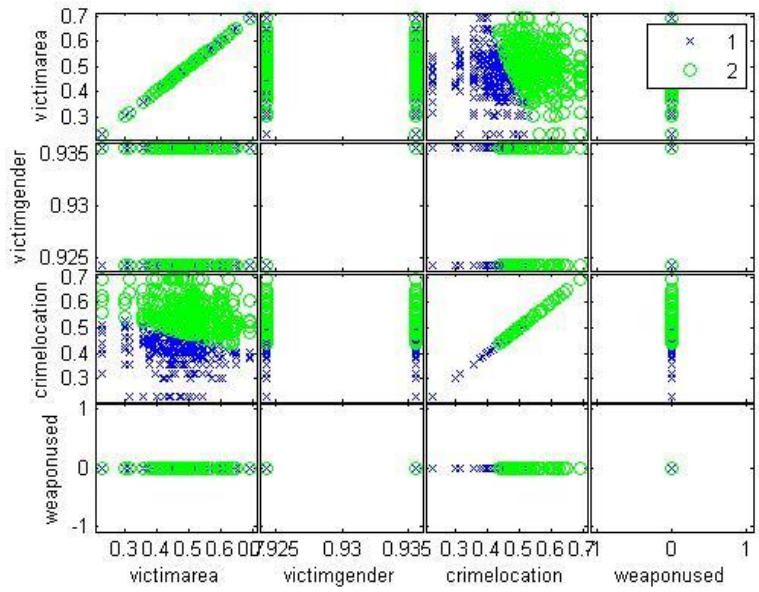


Figure 3-2: scatter matrix plot; clustering, k=2

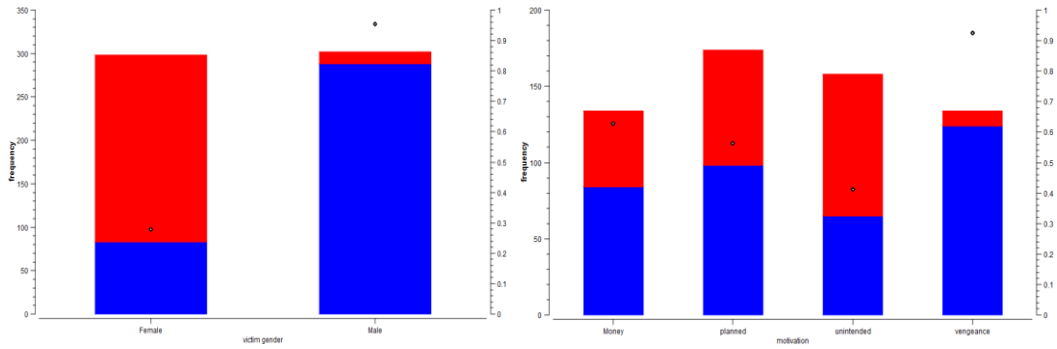


Figure 3-3: (a) Gender Distribution of victims over clusters, (b) Motivation distribution over clusters

3.5.5.5. Evaluation of Clusters

To validate the cluster significance at this stage, silhouette plot is used. The silhouette value for each point is a measure of how similar that point is to points in

its own cluster compared to points in other clusters, and ranges from -1 to +1. It is defined as

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

i.e.

$$s(i) = \begin{cases} 1 - \dfrac{a(i)}{b(i)}, if\ a(i) < b(i) \\ 0, if\ a(i) = b(i) \\ \dfrac{b(i)}{a(i)} - 1, if\ b(i) > a(i) \end{cases}$$

where for each data item *i*, *a(i)* be the average dissimilarity of *i* with all other data item within the same clusters, *b(i)* denote the lowest average dissimilarity to any other cluster. Values near 1 mean that the observation is well placed in its cluster; values near 0 mean that it's likely that an observation might really belong in some other cluster.

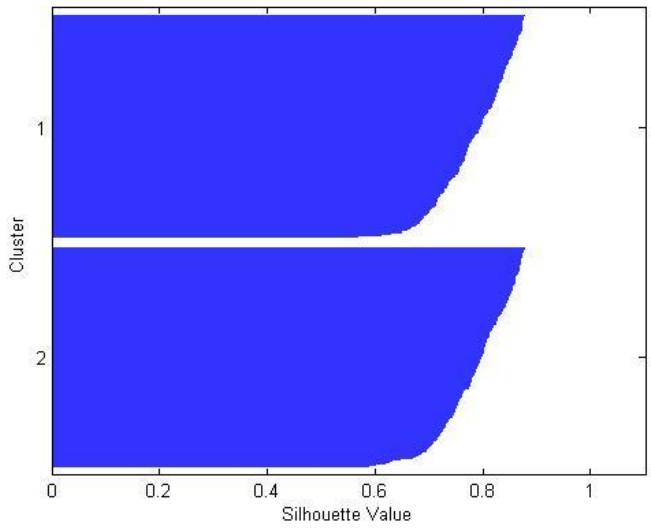


Figure 3-4: Silhouette Plot of Clusters

From figure 3-5, it is evident that the data are assigned to well-formed clusters and the value of *s(i)*=.7848 which is more closer to 1.

3.6. Chapter Summary

The present chapter examined the different types of representation of mixed data attribute (i) using dissimilarity based on category specific properties and (ii) using co-occurrence. Noting that the method based on co-occurrence require at least one numerical attribute, the method has been extended to data having all category type. The efficacy of the method of clustering mixed data type has been demonstrated using a crime data set generated after accessing crime data. The dendrogram has been generated using the agglomerative clustering, which give generated the value of K required for K-means clustering. The scatter matrix plot and distribution of attributes over clusters have been generated and the conclusions drawn from the plots have been discussed. The cophenetic coefficient and the silhouette plot confirm the correctness of the clustering approaches on crime data set. The method of representing the all category data has been used in a study on Breast Cancer reported in Chapter 4 later. The availability of mixed data in the numerical format motivated the author to look into the use of gridded representation, which will be discussed in the Chapter to follow.

Chapter 4

Gridded representation of High Dimensional Dataset

.....

The chapter explores the various techniques to visually represent a multidimensional dataset as a spatial image. The chapter explores the existing visualizing algorithms and various dimensionality reduction techniques at the initial segment of the chapter and a framework of visualizing the reduced dataset is explained further. The alternate approaches proposed in the thesis are also discussed in the chapter.

.....

4. Gridded Representation of High Dimensional Dataset

4.1. Introduction to the chapter

Visualization implies representing data in a visual form. Data visualization is the mapping of data into a Cartesian space [93]. The visualization also gives the user the opportunity to observe the data in an appealing manner and helps to gain a better insight into the data. Data visualization is graphical presentation of a dataset with the aim of providing viewers a quality understanding of the information contents in natural and direct way. A good spatial representation for high dimensional data will be very handy for envisaging it.

4.2. Visualization techniques

The most popular visualization techniques are classified as geometric, icon-based, pixel-oriented, hierarchical, graph-based, and hybrid class [94]. Geometric projection techniques map the attributes to a Cartesian plane like scatter plot, or to an arbitrary space such as parallel coordinates. In geometric methods for visualizing high dimensional data, various techniques like scatter plot, bar charts, survey plots, and parallel coordinates are available.

4.2.1. Scatter plot matrix

A collection of two-dimensional scatter plots is the natural way of extending the scatter plot to higher dimensions. A matrix of scatter plots is an array of scatter plots displaying all possible pair wise combinations of dimensions or coordinates. For n-dimensional data this yields $n(n-1)/2$ scatter plots with shared scales, although most often n^2 scatter plots are displayed. If one has 10 dimensional data, a 10 X 10 array of scatter plots is used to provide a visualization of each dimension versus every other dimension.

Everitt considers that there are two reasons why scatter plots can prove unsatisfactory [95] .When the number of variables exceeds about 10 the number of plots to be examined is very large and is as likely to lead to confusion regarding the structures in the data. Also, the structures existing in the p-dimensional space are not necessarily reflected in the joint multivariate distributions of the variables that are represented in the scatter plots. Despite these potential problems, scatter plot approach are the most commonly used of all the visualization techniques. Scatter plot matrix of iris data set along the four variables, sepal length, petal length, sepal width and petal width is given in the figure 4-1.

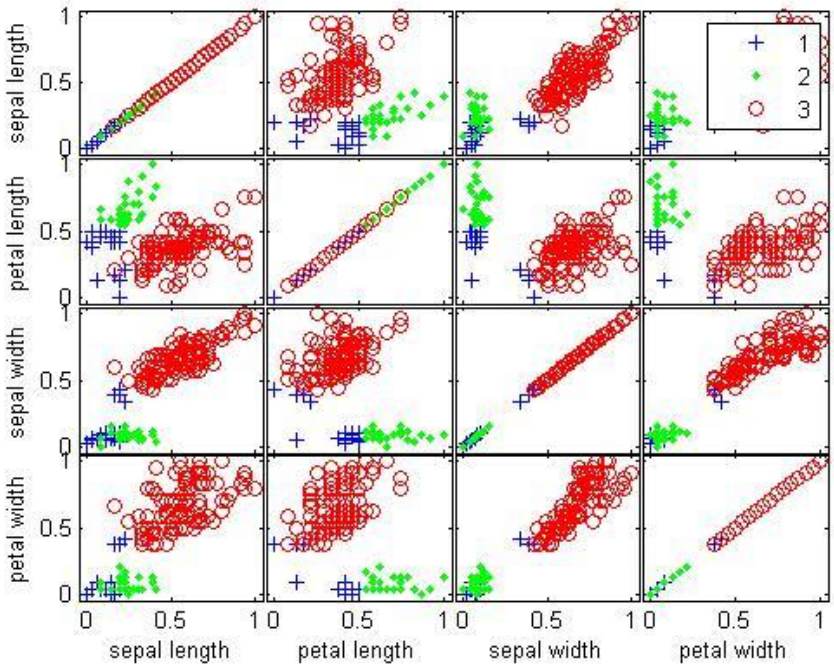


Figure 4-1: Scatter plot matrix of Iris Data

4.2.2. Survey plots

A survey plot is a simple multi-attribute visualization technique that can help to spot correlations between any two variables, especially when the data is sorted

according to a particular dimension [76]. Each horizontal splice in a plot corresponds to a particular data instance. The data on a specific attribute is shown in a single column, where the length of the line corresponds to the value. When data includes a discrete or continuous class, data instances are colored correspondingly. The survey plot method is highly similar to permutation matrix. This visualization of n-dimensional data allows seeing the correlation between two variables, especially when data is sorted in a particular dimension. When color is used with classes, it can be evident that which dimension is best at classifying the data. The survey plot of iris data sorted along sepal length and then the petal length is given in the figure 4-2.

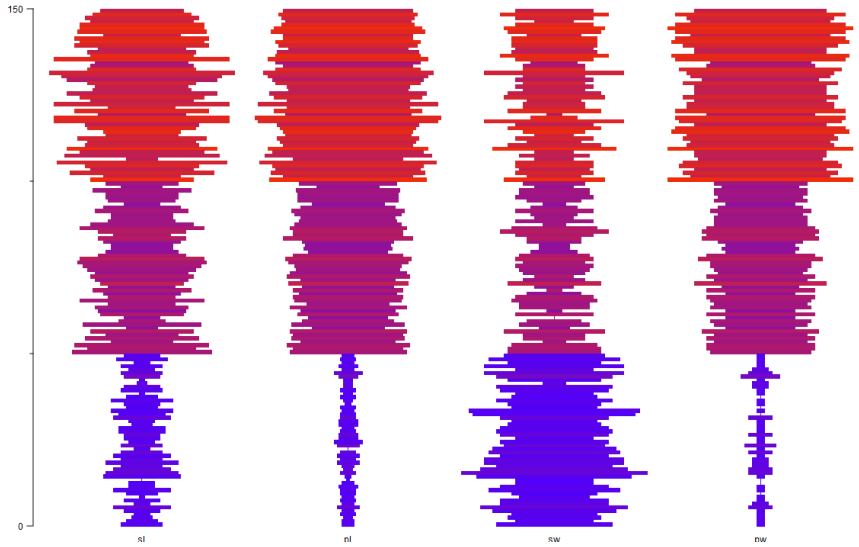


Figure 4-2: Survey plot of Iris Data

4.2.3. Parallel coordinates

Parallel coordinate displays are a tool for visualizing multivariate data. The parallel coordinate plot device is based on the observation that problems associated with Cartesian plotting arise because of the orthogonal constraint. This technique uses the idea of mapping a multi dimensional point on to a number of axes, all of which are in parallel. Each coordinate is mapped to one of the axes and as many axes

as required can be lined up side to side. A line, forming a single polygonal line for each instance represented, then connects the individual coordinate mappings. Thus there is no theoretical limit to the number of dimensions that can be represented. When implemented as software the screen display area imposes a practical limit.

The technique has applications in air traffic control, robotics, computer vision and computational geometry [75] and is been used in the software VisDB [79]. Although it can represent an unlimited number of dimensions, it seems likely that when many points are represented using the parallel coordinate approach, overlap of the polygonal lines will make it difficult to identify characteristics in the data.

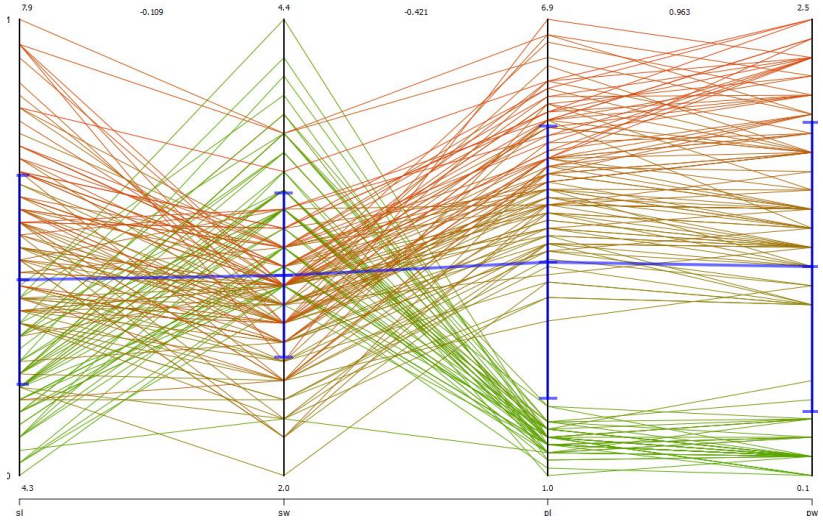


Figure 4-3: Parallel coordinates representation of iris Data.

4.2.4. Pixel- oriented techniques

The idea of the pixel oriented techniques is to use each individual pixel in the screen display to represent an attribute value for some instance in a data set. A color is assigned to the pixel based upon the attribute value. Recursive pattern technique organizes the pixels in small groups and arranges the groups to form some global pattern [96]. A two-step approach is adopted with a first - order pattern formed by

grouping the pixels and a second-order pattern formed by the global arrangement. The result of the second order structure is taken as the basic building element for a third level structure. This process may be iterated up to an arbitrary level, forming a general recursive scheme. The patterns for all recursion levels are identical in a simple case. In more complex cases, the inherent structure will be reflected by the pattern of visualization.

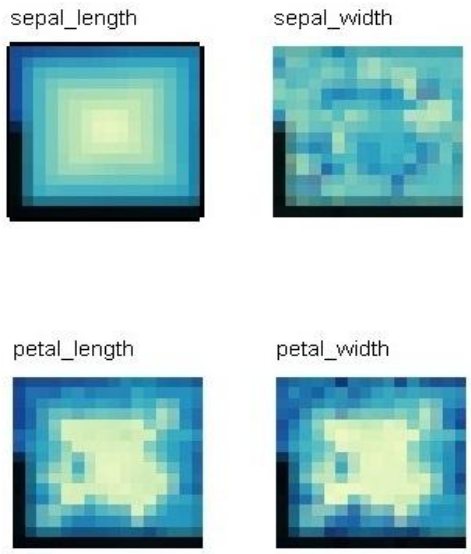


Figure 4-4: Pixel oriented display if Iris Data.

The techniques may be divided into query-independent techniques which directly visualize the data and query-dependent techniques which visualize the data in the context of a specific query [97]. The basic idea of pixel-oriented techniques is to map each data value to a colored pixel and present the data values belonging to one variable in separate windows. Examples for the class of query-independent techniques are the screen-filling curve and recursive pattern techniques. The screen-filling curve techniques are based on the well-known Morton and Peano-Hilbert curve algorithms. The basic idea of space filling curves is to provide a continuous curve which passes through every point of a regular spatial region.

The basic idea of recursive pattern technique is to retain the clustering properties of the Peano-Hilbert and Morton techniques but allow the user to influence the arrangement of pixels that it becomes semantically meaningful The query-independent visualization techniques visualize the variable values by directly mapping them to color. Examples for the class of query-dependent techniques are the snake-spiral and snake axes techniques, which visualize the distances with respect to a database query and arrange the most relevant data items in the center of the display.

4.2.5. Icon- based Visualization

A stylized face is used to represent an instance with the shape and alignment of the features on the face representing the values of the attributes. A large number of faces is then used to represent a data set with one face for each instance. The idea of using faces to represent multidimensional data was introduced by Herman Chernoff [98]. The faces are considered to display data in a convenient form, help find clusters, and identify outliers and to indicate changes over time, all within certain limitations. Data having a maximum of 18 dimensions may be visualized and each dimension is represented by one of the 18 facial features (Courtesy [77]).

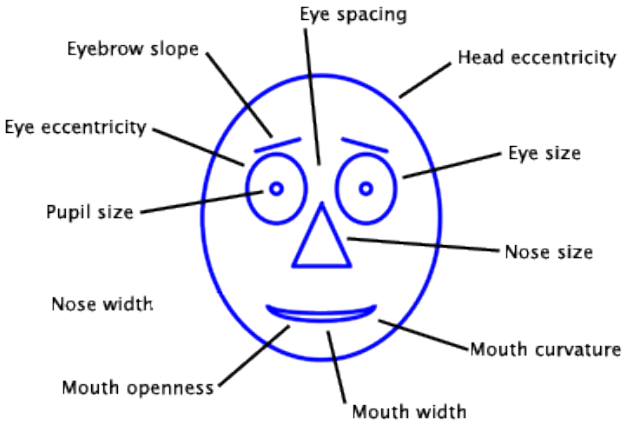


Figure 4-5: Chernoff face

Most of the visualization techniques explored here uses data sets of numeric nature. Much of the data in commercial databases is categorical in nature, and there is a significant need to investigate the manifestation of these algorithms against mixed attributed data.

4.3. Curse of dimensionality

Dimensionality of a problem in information visualization refers to the number of attributes, or more generally as variables, that presents in the data to be visualized [99]. A dataset maybe a set of points drawn from possibly different proportions, and may be a mixture of unknown multivariate and non-parametric distributions. Significant numbers of points may be noisy and there may be missing values as well. The dataset may belong to non-identical attribute spaces which are mixtures of nominal and continuous variables. High dimensional datasets are inherently sparse. In dimensionality reduction, data encoding or transformations are applied to get a reduced representation of the original data. Most of the dimensionality reduction techniques are lossy, in the sense that, only an approximation of original data can be reconstructed.

The dimensionality reduction basically reduce the number of variables to few variables; category variables or categorizations of huge dimensional spaces into understandable fewer partitioned spaces, category spaces, with appropriate discounting of unusual dimensions, variables, categories, and spaces and trends not attributable to robust decision rules from the data. Dimensionality reduction selects some dimension of a dataset in order to create a visualization from which relevant information can be extracted.

4.3.1. Dimensionality Reduction Techniques

Specifically, for a given database D, find an orthonormal set of vectors V, so that when the database D is projected onto the subspace represented by V, the total amount of variance of the projected database D_v [100]. This transformation approximately preserves the distance between the pair of points. A formal definition is given below:

Given a set of data points $\{x_1, x_2,\ldots x_n\}$, the low-dimensional representation is

$$xi \in Rd \rightarrow yi \in Dp \ (p << d) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{Equation 4-1}$$

which maximally preserves the information contained in the original dataset. Dimensionality reduction helps to improve generalized performance, learning speed and the interpretability of learned models.

Dimensionality reduction techniques can be basically classified as linear or non-linear techniques [100]. Linear techniques assume that the data lie on or near a linear subspace of the high-dimensional space. Nonlinear techniques for dimensionality reduction do not rely on the linearity assumption as a result of which more complex embedding of the data in the high-dimensional space can be identified. Linear techniques find the best linear subspace underlying the data that can capture the maximum variance of the projected data. Most common Linear Dimensionality Reduction techniques are

1. Principal Component Analysis (PCA)
2. Singular valued decomposition (SVD)

4.3.1.1. Principal Component analysis (PCA)

Principal Component analysis searches for k-dimensional orthogonal vectors that can be used to represent the data, where $k \leq n$. PCA projects the initial dataset

into a smaller one. Before explaining the procedure of PCA, following concepts are briefly explored.

4.3.1.1.1. Covariance Matrix

Co-variance is a measure to find out how much the dimensions vary from the mean with respect to each other. Covariance is always measured between two dimensions. Covariance between vectors *X* and *Y* is given by equation 5-3.

$$Cov(X,Y) = \frac{1}{n}\sum X_i - \bar{X})(Y_i - \bar{Y})^T$$ .....Equation 4-2

where $\bar{X}$ and $\bar{Y}$ are mean of dimension X and dimension Y respectively and *n* is the total number of observations in each dimension. Covariance matrix is symmetrical

4.3.1.1.2. Eigen values

An Eigen vector is a nonzero vector that satisfies the equation 5-5.

$$A\vec{v} = \lambda\vec{v}$$.....Equation 4-3

where A is a square matrix, λ is a scalar, $\vec{\boldsymbol{v}}$is the eigenvector. λ is called the Eigen value [101] . Eigen values and eigenvectors are also known as, respectively, characteristic roots and characteristic vectors, or latent roots and latent vectors. The significance of an eigenvector is that, when the original matrix is acted on the eigenvector, its changes its magnitude but its direction remains unchanged. A matrix acts on an eigenvector by multiplying its magnitude by a factor, which is positive if its direction is unchanged and negative if its direction is reversed. This factor is the Eigen value associated with that eigenvector. An Eigen space is the set of all eigenvectors with the same Eigen value, together with the zero vectors.

4.3.1.1.3. Principal Component Analysis (PCA)

Let the data set is *X*, be an $m \times n$ matrix, where *m* is the number of variables and *n* is the number of samples (Shlens, 2009). The goal of PCA can be summarized

as finding some orthonormal matrix P in $Y = PX$ such that $C_Y = \mathcal{E}(YY^T)$ is a diagonal covariance matrix.

The rows of P are the principal components of X. Replacing Y with PX,

$$C_Y = \mathcal{E}(PX)(PX)^T)$$

$$= \mathcal{E}(PXX^TP^T)$$

$$= P\mathcal{E}(XX^T)P^T$$

$$C_Y = PC_XP^T \quad \text{.....Equation 4-4}$$

where C_X is the covariance matrix of X. The i,j^{th} element of $\mathbf{C_x}$ is the expectation of outer product between the vector of the i^{th} variable with the vector of the j^{th} variable.

The properties of Cx can be extracted as (i) C_X is a square symmetric $m \times m$ matrix, (ii) the diagonal terms of C_X are the variance of corresponding variables, (iii) the off-diagonal terms of C_X are the covariance between variables. Therefore, C_X can be rewritten as EDE^T where E is the column matrix of eigenvectors and D is a diagonal matrix.

Let P be a matrix with each row p_i as the eigenvector of $\frac{1}{n}XX^T$. i.e. $P= E^T$

Equation 5-6, $C_Y = PC_XP^T$ can be rewritten as

$$C_Y = P(EDE^T)P^T \quad \text{.....Equation 4-5}$$

Since $P=E^T$, $E= P^{-1}$ since E is orthonormal

Equation 5-7 can be written as $C_Y = P(P^TDP)P^T$

$$= (PP^T)D(PP^T)$$

$$C_Y = D \quad \text{.....Equation 4-6}$$

It is clear from the equation 5-8, that the choice of P diagonalizes the C_Y. Also, it can be inferred from C_Y that the P is the eigen vector matrix of C_x principal

components of X are the eigenvectors of $C_X = E(XX^T)$ and i^{th} diagonal value of C_Y is the variance of X along p_i.

The basic procedure of PCA is given as Algorithm 4-1.

Algorithm 4-1: Principal Component Analysis

.....

1. *Normalize the dataset using z-score normalization to Normdataset = X (of dimension n)*
2. *Calculate the covariance matrix of X*
3. *Calculate the eigenvectors and eigen values of the covariance matrix*
4. *The eigenvector with the highest eigen values will be the principal components of the data set.*
5. *Order the eigenvector based on the m (<< n) significance of eigen values.*
6. *Ignore the less significant eigenvalues.*
7. *Form a feature vector matrix given by*

   Feature vector matrix, $F = (eig_1, eig_2, eig_3, \dots, eig_m)$; $m << n$
8. *Reduce the dataset by computing*

   Final dataset = $F * X^T$ (The dimensionality of the reduced data set will be equal to m)

.....

4.3.1.2. Singular Valued Decomposition (SVD)

Principal Component Analysis is based on eigen value decomposition of the covariance matrix C into

$$C = PDP^T$$

where P is orthogonal and D is a diagonal matrix given by

$$D = Diag(\lambda_1, \lambda_2, \lambda_3 \dots \lambda_n).$$

The columns of P are eigenvectors $Cx_i = \lambda x_i$ for the eigenvalues $\lambda_1 \geq \lambda_2 \geq \lambda_3 \dots \geq \lambda_n$. In Singular valued decomposition, the idea of eigenvalue decomposition

can be generalized to an arbitrary matrix A which can be neither symmetric nor a square matrix (Ientilucci, 2003). SVD is based on a theorem from linear algebra which says that a rectangular matrix A can be broken down into the product of three matrices; an orthogonal matrix U, a diagonal matrix Σ, and the transpose of an orthogonal matrix V.

$$A_{m*n} = U_{m*r}\Sigma_{r*n}V_{n*n}{}^{T} \quad \text{.....Equation 4-7}$$

where U and V are orthogonal coordinate transformations and Σ is a rectangular-diagonal matrix of singular values. The columns of matrix U are orthonormal, ~~h~~ and are called the left singular vectors. The diagonal values of Σ viz. (σ_1, σ_2 ,..., σ_n) are called the singular values. The i^{th} singular value shows the amount of variation along the i^{th} dimension.

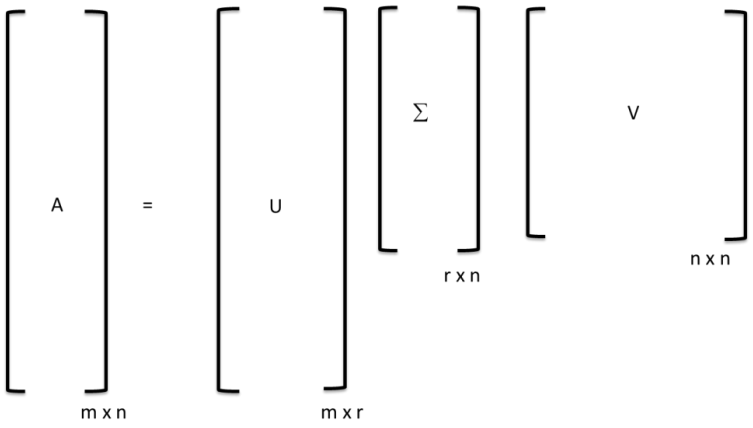


Figure 4-6: Illustration of SVD

Singular valued decomposition is one of the best strategies to reduce the dimensionality by discarding irrelevant information by orthogonal projection into subspace of latent dimensions. The columns of V form a set of orthonormal vectors, which can be regarded as the input basis vectors for Σ. The columns of U form a set of orthonormal vectors, which can be viewed as the output basis vectors for Σ. The

singular values can be thought of as scalar gain controls by which each corresponding input is multiplied to give a corresponding output. The computational complexity of finding U, Σ and V in SVD when applied to a dataset of size $m * n$ (usually $m>>n$) is $4m^2n + 8mn^2 + 9n^3$.

The section below discusses how the SVD can be effectively applied to represent multidimensional data set in two or three dimensions.

4.4. Gridded representation of dataset using SVD

Scaling of data in order to make it 2-Dimensional or 3-Dimensional without losing much information is the greatest challenge in visualizing the data. Also, much of the dataset contains attributes which are non-spatial. When a non-spatial data is visualized into a plane or surface, the spatial component is attached to the non-spatial attributes based on their co-occurrence. The locality factor signifies the similarity of the observations.

The algorithm explained in 4-1 reduces a multi-dimensional numeric dataset into k-dimensional dataset using the singular valued decomposition. As a preprocessing step, the variables of dataset is normalized to a range of [0, 1] so that, no attribute with a high-valued domain dominate the low-ranged domain attributes. SVD is applied on the normalized dataset which yields three matrices U, Σ, and V. The first k columns are taken as the principal components. The dot product of the dataset with each column gives the reduced representation of the dataset. If a 2-Dimensional representation of dataset is needed, then the first two columns will be taken as the principal components. The dot product of the dataset with the first vector will be plotted against the dot product of the dataset with the second vector resulting in the gridded representation of the multi-dimensional data.

Algorithm 4-1: Dimensionality reduction of dataset using SVD

.....

Input: X: dataset with m observations and n dimensions, k: reduced number of dimensions

Output: X_{reduce} *with k dimensions*

Steps:

1. *Normalize the columns of X into an interval [0 1] using the min-max normalization*

$$v' = \frac{v - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A$$

2. *Apply SVD on* X^T*;*

$$[U\ \Sigma\ V] = SVD\ (X^T)$$

3. *Select first k columns of U as principal components. i.e.,*

4. $$U_{reduce} = \begin{bmatrix} e_{11} & e_{12} & \dots e_{1k} \\ e_{21} & e_{22} & \dots e_{2k} \\ e_{n1} & e_{n2} & \dots e_{nk} \end{bmatrix}_{n*k}$$

5. $$X_{reduce_{m*k}} = X_{m*n} * U_{reduce_{n*k}}$$

.....

The case studies of the 2 D or 3D gridded representation with the SVD performed on 6 data sets are discussed in the section to follow. The comprehensiveness of the data representation using two/three singular values is evident from the predominant nature of the two singular values.

4.5. Experimentation with Multivariate Datasets

The data sets which are chosen for the experiments are of different sizes in records and in number of attributes. Datasets consisting of all numerical variables, all categorical variables and of mixed variables are selected. Out of the six datasets, 5 datasets are taken from the UCI machine repository archive. The crime data set is explained in Chapter 3.

4.5.1. Iris dataset (UCI Repository)

The Iris flower data set or Fisher's Iris data set is a multivariate data set introduced by Sir Ronald Fisher (1936) as an example of discriminant analysis [102]. The data set consists of 50 samples from each of three species of Iris; Iris setosa, Iris virginica and Iris versicolor. Four features were measured from each sample: the length and the width of the sepals and petals, in centimeters. The attributes sepal length, sepal width, petal width and petal length are all real and has 150 instances with no missing values.

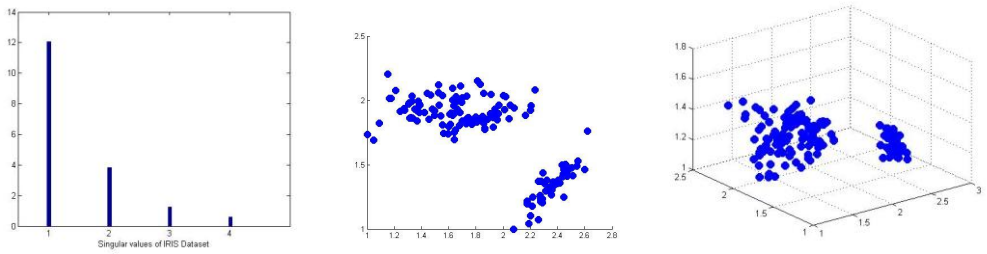


Figure 4-7: (a) Singular values of Iris Data {12.05, 3.82, 1.22, .6, 0,0 ….} (b) 2-D plot of Iris against {12.05, 3.8 } (c) 3D plot of Iris against { 12.05, 3.8, 1.22}. It is interesting to note that the two singular values viz. 12.05 and 3.82, practically absorb most of the representational properties of the reduced data set.

4.5.2. Yeast dataset (UCI Repository)

The yeast dataset is a multivariate dataset primarily used for predicting the Cellular Localization Sites of Proteins. The dataset has 1484 instances with 8 variables excepting the attribute sequence name. The description of the attributes in yeast data are given below in the table 4-1.

Table 4-1: Description of variables in Yeast data

Sl.no	***Attributes with description***
1.	*Sequence Name: Accession number for the SWISS-PROT database*
2.	*mcg: McGeoch's method for signal sequence recognition.*
3.	*gvh: von Heijne's method for signal sequence recognition.*

4. *alm: Score of the ALOM membrane spanning region prediction program.*
5. *mit: Score of discriminant analysis of the amino acid content of the N-terminal region (20 residues long) of mitochondrial and non-mitochondrial proteins.*
6. *erl: Presence of "HDEL" substring*
7. *pox: Peroxisomal targeting signal in the C-terminus.*
8. *vac: Score of discriminant analysis of the amino acid content of vacuolar and extracellular proteins.*
9. *nuc: Score of discriminant analysis of nuclear localization signals of nuclear and non-nuclear proteins.*

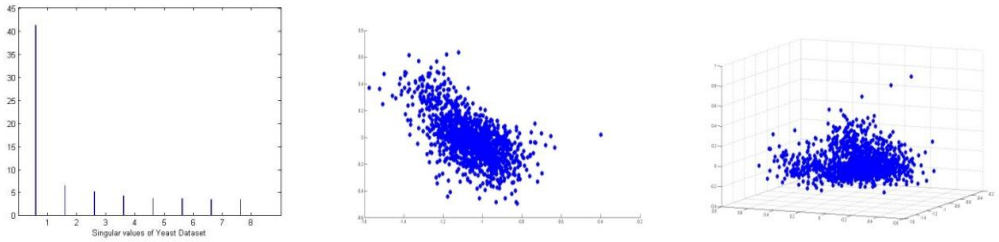


Figure 4-8: (a) Singular values of Yeast; most significant {41.09, 6.5, 5.2, 4.2, 3.7,3.6, 3.5, 3.4, 0,0,… } (b) 2-D plot of yeast dataset against {41.09,6.5} (c) 3-D plot of Yeast against {41.09,6.5,5.2}. The relative lower significance of the lower singular values are worth noting.

4.5.3. Wine Dataset (UCI Repository)

Wine data are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the three types of wines. The dataset contains 178 observations. The attributes are given in the table 4-2.

Table 4-2: Description of variables of wine data

Sl.no	***Attribute***
1.	*Alcohol*
2.	*Malic acid*

3. *Ash*
4. *Alcalinity of ash*
5. *Magnesium*
6. *Total phenols*
7. *Flavanoids*
8. *Nonflavanoid phenols*
9. *Proanthocyanins*
10. *Color intensity*
11. *Hue*
12. *OD280/OD315 of diluted wines*
13. *Proline*

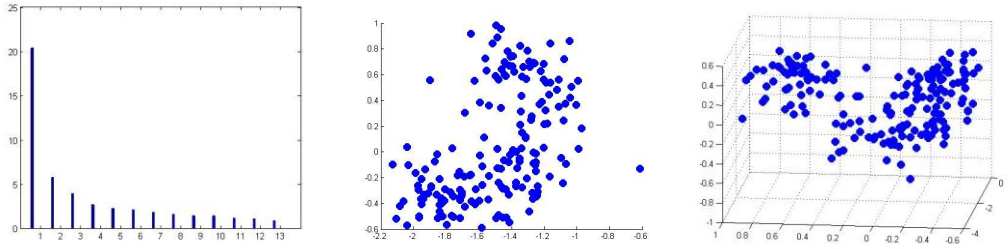


Figure 4-9: (a) singular values of Wine; most significant {20.35, 5.8, 3.94, 2.7, 2.3, 2.12, 1.8, 1.6, 1.47, 1.46, 1.21, 1.11, .88, 0, 0 ….. } (b) 2-D plot of wine dataset against {20.35, 5.8} (c) 3-D plot of wine against {20.35, 5.8, 3.94}

4.5.4. Thyroid Dataset (UCI Repository)

The Thyroid dataset is a mixed dataset contains thyroid disease records supplied by the Garavan Institute and J. Ross Quinlan and is obtained from UCI machine learning repository. The dataset contains 9172 observations with missing values. The missing values in each column were filled by the most frequent/average value of the column. The details of variables of the data set are given in the table below.

Table 4-3: Description of variables in Thyroid Data

Sl.No	Attribute Name	Possible Values
1.	*age*	*Continuous.*
2.	*sex*	*M, F.*
3.	*on thyroxine*	*f, t.*
4.	*query on thyroxine*	*f, t.*
5.	*on antithyroid medication*	*f, t.*
6.	*sick*	*f, t.*
7.	*pregnant*	*f, t.*
8.	*thyroid surgery*	*f, t.*
9.	*I131 treatment*	*f, t.*
10.	*query hypothyroid*	*f, t.*
11.	*query hyperthyroid*	*f, t.*
12.	*lithium*	*f, t.*
13.	*goitre*	*f, t.*
14.	*tumor*	*f, t.*
15.	*hypopituitary*	*f, t.*
16.	*psych*	*f, t.*
17.	*TSH measured*	*f, t.*
18.	*TSH*	*Continuous.*
19.	*T3 measured*	*f, t.*
20.	*T3*	*Continuous.*
21.	*TT4 measured*	*f, t.*
22.	*TT4*	*Continuous.*
23.	*T4U measured*	*f, t.*
24.	*T4U*	*Continuous.*
25.	*FTI measured*	*f, t.*

26.	*FTI*	*Continuous.*
27.	*TBG measured*	*f, t.*
28.	*TB*	*Continuous.*
29.	*referral source*	*WEST, STMW, SVHC, SVI, SVHD, other.*

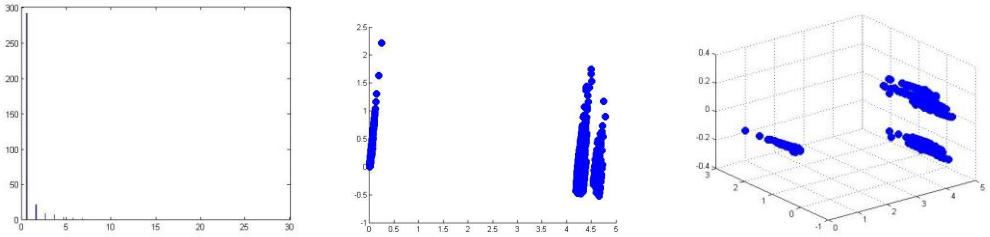


Figure 4-10: (a) singular values of thyroid dataset {291.29, 21.63, 9.46, 7.48, 4.34, 3.24, 3.02, 0,0,…}. (b) 2-D plot against most significant {291.29, 21.63}. (c) 3-D plot against {291.29, 21.63, 9.46}.

4.5.5. Breast Cancer Dataset (UCI Repository)

Breast cancer dataset is a multivariate dataset consisting of 286 instances. The instances are described by 9 attributes, which all are categorical in nature. The details of attributes are given below.

Table 4-4: Description of variables of breast cancer data

Sl.No	***Attribute***	***Possible Values***
1.	*Age*	*10-19, 20-29, 30-39, 40-49, 50-59, 60-69, 70-79, 80-89, 90-99.*
2.	*Menopause*	*lt40, ge40, premeno.*
3.	*tumor-size*	*0-4, 5-9, 10-14, 15-19, 20-24, 25-29, 30-34, 35-39, 40-44, 45-49, 50-54, 55-59.*
4.	*inv-nodes*	*0-2, 3-5, 6-8, 9-11, 12-14, 15-17, 18-20, 21-23, 24-26, 27-29, 30-32, 33-35, 36-39.*
5.	*node-caps*	*yes, no.*

6.	*deg-malig*	*1, 2, 3.*
7.	*Breast*	*left, right.*
8.	*breast-quad*	*left-up, left-low, right-up, right-low, central.*
9.	*Irradiat*	*yes, no.*

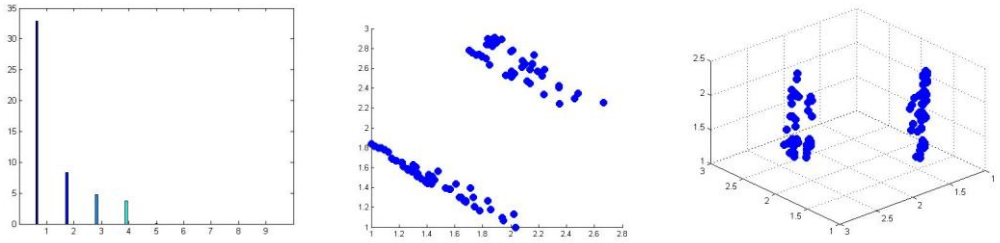


Figure 4-11: (a) singular value plot for Breast Cancer data {32.9, 8.20, 4.76, 3.73, 0.02, 0,0…}. (b) 2-D plot against most significant {32.9, 8.2}. (c) 3-D plot against most significant {32.9, 8.20, 4.76}

The preprocessing of the categorical variables is done based on the extension mentioned in Section 3.3.2. The first phase of the algorithm identified the base attribute, the attribute which has got the highest number of elements. The elements in the base attribute are taken as the base items and the frequency of each base item is found out. A surrogate numerical attribute is created with the corresponding normalized frequency of each base item and is treated as the numerical variable with minimum group variance.

4.5.6. Crime Dataset

The 2-dimenional and 3-Dimensional view of the mixed Crime Dataset described in chapter 2 are given below.

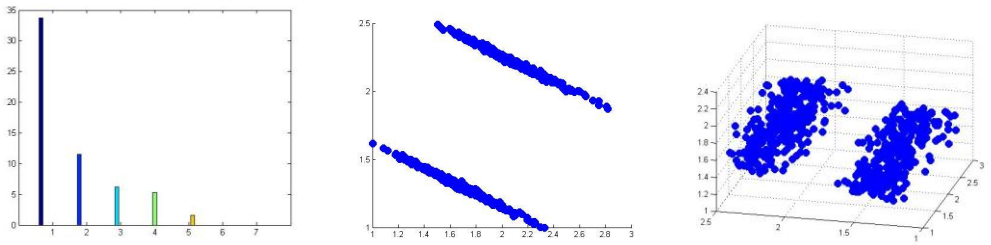


Figure 4-12: (a) singular values plot of Crime dataset { 33.76, 11.8, 6.22, 5.24, 1.56, 0,0,…}. (b) 2-D plot against of Crime Dataset against most significant {33.76, 11.8}. (c) 3-D plot against of Crime dataset against most significant {33.76, 11.8, 6.22}

The crime Dataset contains 600 observations defined by crime id, victim area, victim gender, victim age, Crime location, weapons used, motivation for the crime and the identified criminal name. All the attributes except victim age are categorical in nature.

4.6. Chapter summary

It is demonstrated that the representation with respect to two prominent singular values does retain most of the pertinent information in the data set. The singular values directly drive the visualization as against the different pair wise choices in the case of scatter plot. Apart from the ease of visualization, the representation at lower dimensions also results in utilization of spatial techniques in clustering.

Chapter 5

Spatial Clustering using Quad trees

.....

Having represented multidimensional data into 2D or 3D gridded representation, inferring meaningful clusters out of the spatially representation is the main topic covered under the purview of this chapter. The techniques developed here extensively utilizes the notion of spatial data structures like Quad trees, which are revisited and redefined using the concepts of Fuzzy set theory .

.....

5. Spatial Clustering using Quad trees

5.1. Introduction to the Chapter

Spatial data means data related to space [103]. The space of interest can be the two-dimensional abstraction of the surface of the earth or a man-made space like the layout of a VLSI design, a volume containing a model of the human brain, or another 3d-space representing the arrangement of chains of protein molecules. The data consists of geometric information and can be either discrete or continuous. The explicit location and extension of spatial objects define implicit relations of spatial neighborhood (such as topological, distance and direction relations) which are used by spatial data mining algorithms. Therefore, spatial data mining algorithms are required for spatial characterization and spatial trend analysis. Spatial data mining or knowledge discovery in spatial databases differs from regular data mining in parallel with the differences between non-spatial data and spatial data. The attributes of a spatial object stored in a database may be affected by the attributes of the spatial neighbors of that object. In addition, spatial location and implicit information about the location of an object may be exactly the information that can be extracted through spatial data mining [2].

5.2. Spatial Data

Spatial data consists of data that have a spatial component. Spatial objects can be made up of points, lines, regions, rectangles, surfaces, volumes, and even data of higher dimension which includes time. The spatial component is implemented with a specific location attribute such as address or implicitly done by partitioning the database based on location. Geographic Information systems (GIS), biomedical applications including medical imaging, agricultural science etc. produces large volume of spatial data.

5.3. Spatial Data structures

Special data structures are designed to store or index spatial data and are specifically designed to incorporate unique features of it. To perform spatial queries which may involve proximity measures based on relative locations of spatial objects, spatial objects which are close, is clustered on disk. The geographic space under consideration may be partitioned into cells based on proximity and these cells would be then related to storage locations.

5.3.1. Minimum Bounding rectangle (MBR)

Minimum bounding rectangle is as the smallest rectangle that contains every point in the region [104]. The boundaries of the rectangle are aligned with the major and minor axes of the bounded region.

5.3.2. Quad trees

The term quad tree is used to describe a class of hierarchical data structure whose common property is that they are based on the principle of recursive decomposition of space [105]. They can be differentiated on the following bases: (1) the type of data that they are used to represent, (2) the principle guiding the decomposition process, and (3) the resolution. Currently, they are used for points, rectangles, regions, curves, surfaces, and volumes. The decomposition may be into equal parts on each level (termed a regular decomposition), or unequal parts as may be governed by the situation in hand. The resolution of the decomposition (i.e., the number of times that the decomposition process is applied) may be fixed beforehand or it may be governed by properties of the input data. Diagrammatic representation (example adopted from Wikipedia) is given below in figure 5-1. From the stand point of data structure, the quad tree is a complete tree.

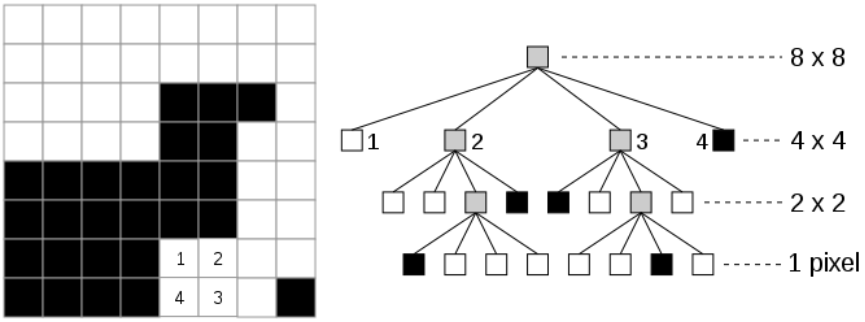


Figure 5-1: Image representation using Quad trees

5.3.3. R-Tree

The basic idea of R-tree is to group nearby objects and represent them with their minimum bounding rectangle in the next higher level of the tree. In other words, R-tree is used to index spatial data represented as MBRs [106] . Each layer in the tree identifies smaller rectangles. Cell may overlap in R-tree.

R-Tree is a height balanced tree similar to B-Tree with index records in its leaf nodes containing pointers to data objects. Data in R-trees is organized in pages that can have a variable number of entries. Each entry within a non-leaf node stores two pieces of data: a way of identifying a child node, and the bounding box of all entries within this child node. Leaf nodes store the data required for each child, often a point or bounding box representing the child and an external identifier for the child. For point data, the leaf entries can be just the points themselves. For polygon data the common setup is to store only the MBR of the polygon along with a unique identifier in the tree.

A spatial database consists of collection of tuples representing spatial objects, and each tuple has a unique identifier which can be used to retrieve it. Leaf nodes in an R-Tree contain index record entries of the form *(I, tuple-identifier),* where tuple-identifier refers to a tuple in the database and I is an n-dimensional rectangle which is

the bounding box of the spatial object indexed $I= (I_0, I_1, \ldots I_{n-1})$. n is the number of dimensions and I_i is a closed bounded interval $[a, b]$ describing the extent of the object along dimension *i*. Non-Leaf nodes contain entries of the form *(I, child-pointer)* where child-pointer is the address of a lower node in the R-Tree and I covers all rectangles in the lower node's entries.

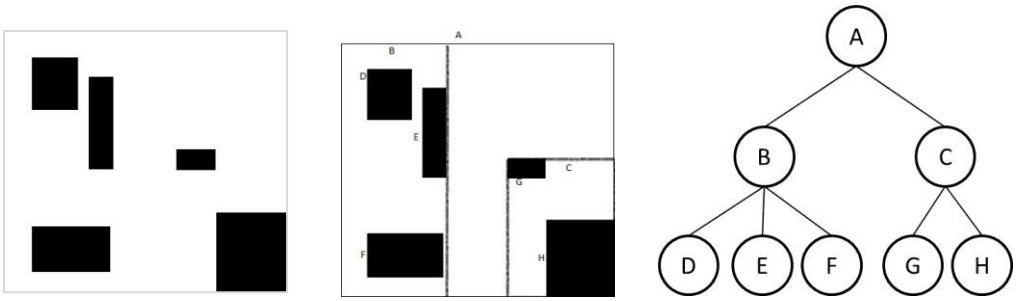


Figure 5-2: R-Tree Example

5.3.4. k-D Tree

A k-D tree which is a variation of binary search tree is designed to index multi attribute data. Each level in the tree is used to index on of the attributes. k-D Tree is a space-partitioning data structure for organizing points in a k-dimensional space and is implemented as a binary tree in which every node is a k-dimensional point [107]. If a file is stored as a node in the tree, then each record in the file is stored as a node in the tree. A node contains k keys which comprise the record and two pointers which are either null or may point to another k-D tree. Also a discriminator which is an integer with value *[0, k-1]* is associated with each node. All nodes at some given level have the same discriminator. Each node in the tree represents a division of the space into two subsets based on the division point used. , also the division alternates between the two axes.

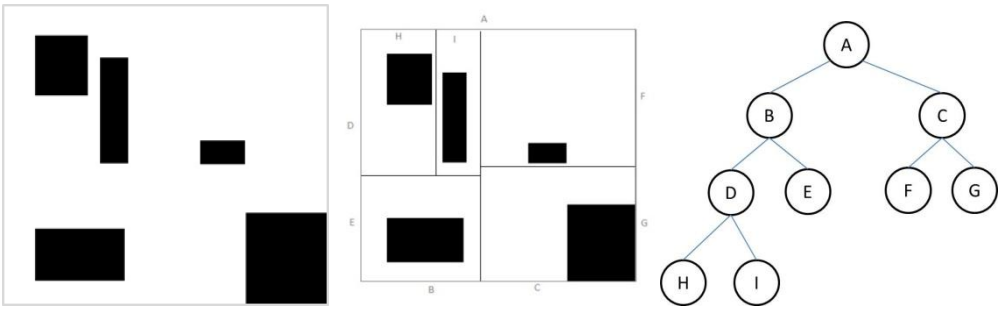


Figure 5-3: k-D tree example; each node stores k keys.

5.4. Quad tree decomposition

5.4.1. Process of decomposition

Quad tree decomposition is an analysis technique that involves subdividing an image into blocks that are more homogeneous. For the ease of computation, the image is square in shape with its length n is considered to be of the power of 2. The top-down approach will consist of successive non overlapping subdivisions of the original image blocks with dimensions $2^n \times 2^n$ by a factor of four. If a block is homogeneous, given some criterion it is not subdivided. If it is non-homogeneous, it is subdivided into non-overlapping four sub-blocks with dimensions $2^{n-1} \times 2^{n-1}$. Quad tree encoding will segregate the image into equal 4 quadrants sub-blocks. All these sub-division of the non-homogeneous blocks will continue until the smallest block reaches a minimum pre-established block size. The pixel values of the image frame is normalized to a range of [0 1].

5.4.2. Homogeneity

The word homogeneity according to Merriam –Webster's Dictionary is the quality or state of being homogeneous, where homogeneous is defined as "of the same or a similar kind or nature". Homogeneity is as well defined as "the state of having identical cumulative distribution function or values". Different metrics are

used to find out the homogeneity of a region. A region is said to be homogeneous if, if all the pixels in the region are within a specific dynamic range. For example, in a black and white image, a region is homogeneous if it entirely white or entirely black or minute enough to be identified as black or white.

5.4.2.1. Homogeneity based on statistical metrics

The mean and the variance are two statistical measures which can be effectively used to find the homogeneity of the region. Each Quad tree region is analyzed for its mean and variance in pixel values. Without the loss of generality, assume the size of input image array is $n \times n$, where n is power of two. An image is represented mathematically by a spatial brightness function $f(x, y)$ where (x, y) denotes the spatial coordinate of a point in the image. Also assume the size of input image array of the region under consideration is $w \times w$, where w is power of two. The mean value can be calculated by the following formula

$$\mu_w = \frac{1}{w^2}\sum_0^w \sum_0^w f(x,y) \quad \text{.....Equation 5-1}$$

The standard deviation of the region is given by the following formula

$$\sigma_w = \frac{1}{w^2}\sqrt{(f(x,y) - \mu_w)^2} \quad \text{.....Equation 5-2}$$

5.4.2.2. Homogeneity based on entropy

Entropy is a statistical measure of randomness that can be used to characterize the texture of the input image. Entropy is calculated based on distribution of similar pixels in the image frame buffer D.

$$Entropy(D) = -\sum_{i=1}^{m} p_i \log(p_i) \quad \text{.....Equation 5-3}$$

where p_i is the probability of pixel of color C determined by the number of pixels with color C, C_i, by the total number of pixels in D. Lower entropy implies higher homogeneity and vice versa. Since the entropy for non homogenous region is always

> 0 and dependant on the data, with no indication of extent of homogeneity, the clustering based on this property was not attempted in the present work.

5.4.3. Quad tree Decomposition based on Fuzzy rules

A fuzzy set is a set containing elements that have varying degrees of membership in the set. Elements of a set are mapped to a universe of membership values using a function theoretic form. This function maps elements of a fuzzy set A to a real numbered value on the interval 0 to 1. An element x, is a member of fuzzy set A, then $\mu_A(x) \in [0\ 1]$, where μ is the membership function. A fuzzy system is characterized by a set of semantic statements based on expert knowledge. The expert knowledge is usually in the form of if-then rules.

A fuzzy set A in X is characterized by a membership function which is implemented by fuzzy conditional statements. If the antecedent is true to some degree of membership, then the consequent is also true to that same degree.

If antecedent then consequent.

The *If* part involves evaluating the antecedent, by fuzzifying the input. The *then* part requires application of that result to the consequent, known as inference. Fuzzy inference is the process of devising a mapping from a given input to an output using fuzzy rules.

Two types of fuzzy inference models are generally used; Mamdani and Takagi-Sugeno-Kang, method of fuzzy inference [108], [109], [110]. Mamdani's fuzzy inference method is the most commonly seen fuzzy methodology. The fuzzy implication modeled by Mamdani's minimum operator and is defined by the max operator.

5.4.3.1. Fuzzy rules for quad tree decomposition

The homogeneity criterion for decomposing Quad tree described below is based on fuzzy inference. The quad tree region of a grey-scale image can be considered homogeneous if, region satisfies the following intensity.

a. *Pure black*
b. *Pure white*
c. *Highly Black*
d. *Highly white*

If the region is in any other intensity range other than the above listed, say, intermediately gray, the region should be further decomposed to a level until which it satisfies any of the above intensity. For finding out the intensity level, we make use of mean μ and standard deviation σ of the pixels of the region to be considered. The rules are formulated as

R1. If μ *is low and* σ *is low then region is highly black; info :=1*
R2. If μ *is medium and* σ *is low then the region is medium gray; decompose*
R3. If μ *is high and* σ *is low then the region is highly white; info:=0*
R4. If μ *is low and* σ *is high then region is scattered towards black; decompose*
R5. If μ *is medium and* σ *is high then the region is mixed; decompose*
R6. If μ *is high and* σ *is high scattered towards white; decompose*

The consequence part is a decision whether to decompose the quadrant into further level or not. We use fuzzy logic to reason the antecedent part that translates the crisp mean and standard deviation values to a decision of decompose or not. The membership function of μ to be low is specified as

$$\mu_{low}(\mu) = \frac{1}{(\frac{\mu}{0.07})^8+1} \quad \text{.....Equation 5-4}$$

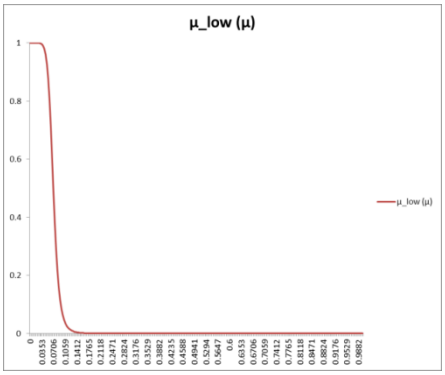


Figure 5-4: Plot of μ_low (μ)

The membership function of μ to be medium is specified as

$$\boldsymbol{\mu}_{medium}(\mu) = \boldsymbol{e}^{-\frac{(\mu-0.5)^2}{0.05}} \quad \text{.....Equation 5-5}$$

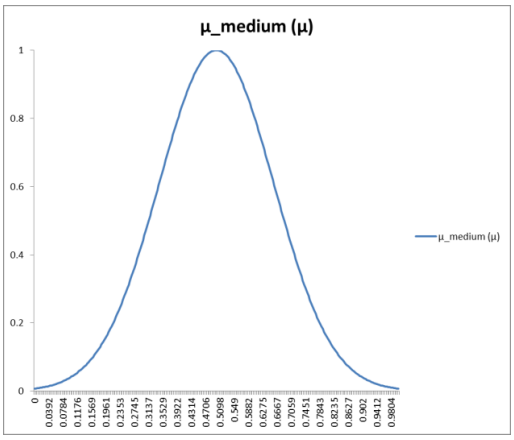


Figure 5-5: Plot of μ _medium (μ)

The membership function of μ to be high is specified as

$$\mu_{high}(\mu) = 1 - \frac{1}{1+(\frac{\mu}{0.9})^{55}} \quad \text{.....Equation 5-6}$$

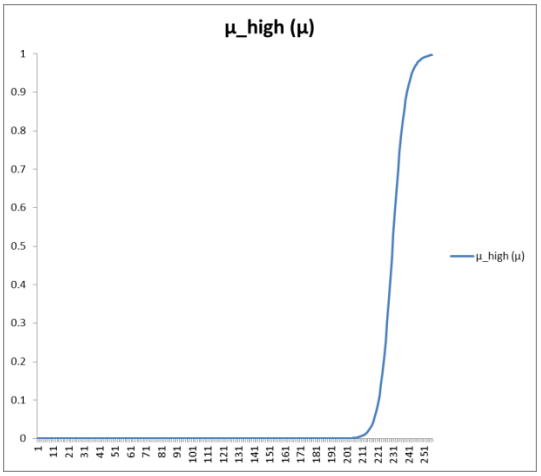


Figure 5-6: Plot of μ_high (μ)

The membership functions of standard deviation σ to be low is specified as

$$\mu_{low}(\sigma) = \frac{1}{1+(\frac{\sigma}{0.1})^9} \quad \text{.....Equation 5-7}$$

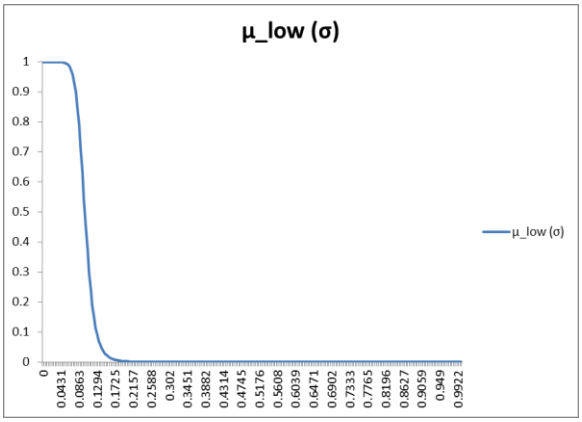


Figure 5-7: Plot of μ_low (σ)

The membership functions of standard deviation σ to be high is specified as

$$\mu_{high}(\sigma) = 1 - \frac{1}{1+(\frac{\sigma}{0.09})^{11}} \quad \text{.....Equation 5-8}$$

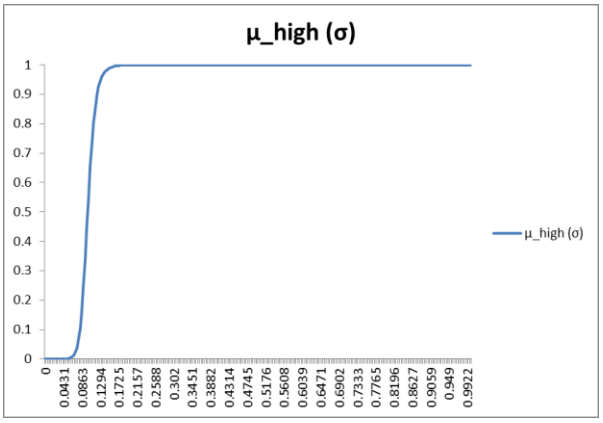


Figure 5-8: Plot of μ_high (σ)

The rules R1: As per the union rule of fuzzy inference system, the decision to decompose or not, can be stated as *max [R1, R2, R3, R4, R5, R6]* where each rule is given by

R1: $\mu_{low}(\mu)$.$\mu_{low}(\sigma)$

R2: $\mu_{medium}(\mu)$. $\mu_{low}(\sigma)$

R3: $\mu_{high}(\mu)$. $\mu_{low}(\sigma)$

R4: $\mu_{low}(\mu)$.$\mu_{high}(\sigma)$

R5: $\mu m_{edium}(\mu)$. $\mu_{high}(\sigma)$

R6: $\mu_{high}(\mu)$. $\mu_{high}(\sigma)$

If rule R1 is fired, the region is highly dense and its info bit is indicated as 1 and if rule R3 is fired, the region is devoid of information and its info bit is set to 0. The final output of the first phase of the algorithm is a Quad tree decomposed to a final level of leaf nodes whose info bit is either set to 0 or 1.In order to avoid pixel level subdivision, a threshold level for the size of block is predetermined. When the sub quadrant size reaches the threshold level, the algorithm probes the mean of the

region and if it is less than 0.5 (normalized to the interval [0 1]), it takes the region as informative and sets its info bit to 1.

5.5. Merging neighbor dense quadrants

The neighbor finding in a quad tree is one among the area, in which considerable research has been is going on. Most of the neighbor finding algorithm label each node, based on some strategy and use this label for finding the neighborhood quadrants. This research study proposes a new method to find neighbor quadrants based on their corner coordinates. The basic idea behind this algorithm is that, irrespective of the size of the quadrant, two quadrants will be neighbors, if any of the corner points are nearer. Here, the distance between two corner points is 1, if they are closer. But to obtain more realistic clusters, a threshold value T is set as the minimum distance for two clusters to be neighbors is average of the least 5 minimum distances between quadrants. The distance will never become 0; since no points are shared by the quadrants. This algorithm will find neighbors to all directions such as north, south, east, west, north east, northwest, south east, and south west. The complexity of the algorithm increases with number of quadrants. The algorithm to merge two neighbor dense quadrants is given in Algorithm 5-1.

Algorithm 5-1: Merge_neighbor

……

Input:

Q_1 : left top corner coordinates (q1ltx,q1lty), right top corner coordinates (q1rtx,q1rty), left bottom corner coordinates (q1lbx,q1lby), right bottom corner coordinates (q1rbx,q1rby)

Q_2 : left top corner coordinates (q2ltx,q2lty), right top corner coordinates (q2rtx,q2rty), left bottom corner coordinates (q2lbx,q2lby), right bottom corner coordinates (q2rbx,q2rby).

Output :

Q_n labeled with Cluster_index; Total number of clusters=New_cluster_index;

Steps:

Cluster_index(Q_n)=0;

New_cluster_index=0;

While every pair of quadrants not compared

If is-neigh(Q_1,Q_2)

If cluster_index(Q_1) !=0 or cluster_index(Q_2)!=0

Cluster_index(Q_2)=Cluster_index(Q_1)

Else

New_cluster_index=New_cluster_index+1;

Cluster_index(Q_1)=Cluster_index(Q_2)=New_cluster_index;

End

End

Function: is_neigh (Q1,Q2)

Threshold ;T=average(least 5 quadrant distances)

If distance_between(q1(left top corner),q2(all corners)!= T and

distance_between(q1(left right corner),q2(all corners)!=T and

distance_between(q1(bottom left corner),q2(all corners)!=T and

distance_between(q1(bottom right corner),q2(all corners)!=T

is_neigh=0;

else

is_neigh=1;

end

return is_neigh

……..…….

5.6. Extraction of cluster boundary using Information content

Image information content is an intrinsic property of an image. It does not depend on any subjective interaction with it and is determined solely by the available

image data. Kolmogorov complexity is a modern notion of randomness dealing with the quantity of information in individual objects; that is, point wise randomness rather than average randomness as produced by a random source [111]. The Kolmogorov complexity of an object is a form of absolute information of the individual object. The original theory deals with binary strings which are called objects. Emanuel Diamant consider image as an object, because any object can be described by a finite string of signs [112]. Kolmogorov complexity is used as a paradigm for image information extraction and discovery [113]. Low-level Image information is represented by a feature vector, which contains a number of units associated with local spatially restricted interactions between neighboring pixels.

The structure that can be perceived at low level is an edge structure. An edge can be viewed as a discontinuity in spatial homogeneity. To extract the low level information content, two components are computed; a topological component and intensity information denoted by $I_{top}(x, y)$ and $I_{int}(x, y)$. The term intensity change is applicable to different properties of local discontinuity. Here the change in pixels' gray scale values are taken into consideration. A measure of local information content $I_{loc}(x,y)$ can be measured as a product of two constituting components:

$$I_{loc}(x, y) = I_{top}(x, y).\ I_{int}(x, y) \quad \text{.....Equation 5-9}$$

The local information content is measured for every pixels in the image array and (x,y) represents the image coordinates. To compute the $I_{loc}(x, y)$ we assume a local spatial organization for a pixel and its nearest neighbor interaction, a sliding window of 3 x 3 sizes centered to the central pixel of the arrangement. $I_{int}(x, y)$, the intensity information can be estimated as the mean absolute difference between the central pixel gray level, g_c and the gray levels of its 8 neighbors, g_n . Only n results which are greater than zero are taken into account. Then the intensity component can be computed as:

$$int(x,y) = \frac{1}{n}\sum_{n}^{1}|g_c - g_n| \quad \text{.....Equation 5-10}$$

To account for topological component, we define a term called "status". The term status distinguishes between two discernible states: pixels that are at a lower intensity level than its surrounding neighbors and pixels that are at a higher or equal intensity levels than its surrounding neighbors. Mathematically it can be represented as:

$$stat = 8\,gc - \sum_{i-1}^{8} g\,i \quad \text{.....Equation 5-11}$$

where g_c is the gray level of central pixel value and $\Sigma\, g_i$ is the sum of gray level values of its eight neighbors. Encode the result into zero- if the result is negative and one- otherwise. Status is evaluated for every pixel in the image array and is mapped into a image sized status map. The topological component can be estimated as:

$$I_{top}(x,y) = p(1-p) \quad \text{.....Equation 5-12}$$

where p is the probability that the central pixel share the same status state with its neighbors. The support for each I_{top} value is defined as a 3x3 matrix; p the probability can be replaced with, m where m is the number of neighbors that share the same status with the central pixel. The above procedure will find the edges of cluster from merged dense quadrants.

5.7. Re-mapping the identified clusters and outlier detection

All the data elements of the original dataset are represented as a point in the spatial image. By spatially representing the data points, only the dimensionality is compromised, but the number of observations in the original dataset is kept all through the process. All the data points which exist in the area of merged dense quadrants will have cluster index value associated with it and can be easily appended with cluster-index as an attribute.

5.7.1. Outliers

Outliers are those values which act differently from other set of values. Outlier can be a measurement error or an instance which may reflect a distinctive behavior. Outlier detection is equally important as cluster identification. The proposed framework easily identifies the outliers by assigning them to quadrants to which no cluster index values are appended or a cluster index value which is been associated with a quadrant with smaller area. The quadrants which are not dense may contain scattered data points and can be treated as outliers.

5.7.2. Comparison with K mean clustering in reduced dimensions.

Though K-means clustering could have been attempted in lower gridded space generated in section 4. 3, the requirement to decide on the cluster centers will still be major issue. Also the clustering works on the data points struggling to cluster even the outliers. Often the dendrogram or visual plots could aid the selection of cluster enters. The proposed method of segmenting the 2D space, on the other hand, achieves clustering by pruning out scattered data elements as part of the quad tree partitioning. Further segmentation thus collects together dense groups of the data set.

5.8. Experimentation with different datasets

5.8.1. Iris dataset

Iris dataset is a classic data set with 50 samples of iris flowers measured in four attributes, sepal length and width, petal length and width. The result of the third phase of the framework is given below in the figure 5-9. The results indicates that clearly two clusters can be identified from visual representation.

Cluster analysis on this data set only contains two clusters with rather obvious separation. One of the clusters contains *Iris setosa*, while the other cluster

contains both *Iris virginica* and *Iris versicolor* and is not separable without the species information [114]. The evaluation of the clusters using silhouette plot given by the variable s= .8136 also showed a good cluster formation.

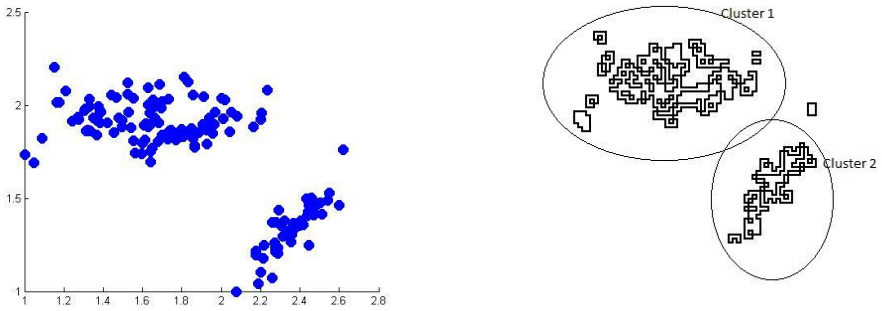


Figure 5-9: (a) 2-D plot of iris data, (b) segmented clusters from 2-D plot

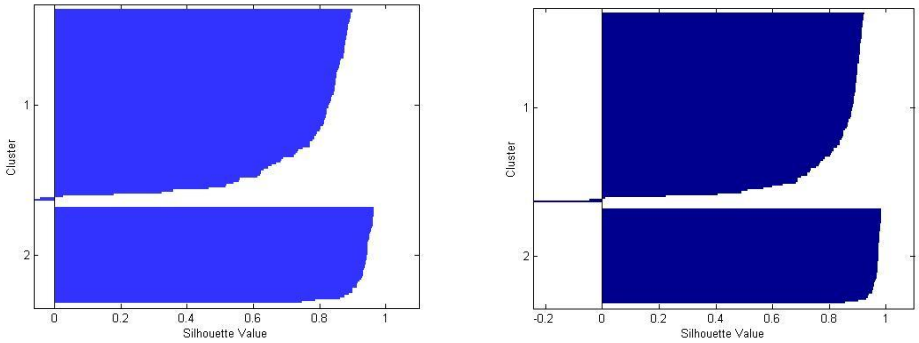


Figure 5-10: (a) Silhouette plot of Iris clusters on lower dimensions of data, (b) Silhouette plot of Iris clusters on original data

5.8.2. Yeast Dataset

Yeast dataset contained 1484 instance of gene expression data defined by 8 predictive localized site of protein. The remapping of cluster indices after the fourth phase of the framework grouped the data in to 2 clusters; one main cluster containing 99% of the data and another cluster with feeble number of values. The silhouette plot given in the figure of 2 clusters also indicated an average clustering (s= 0.5376).

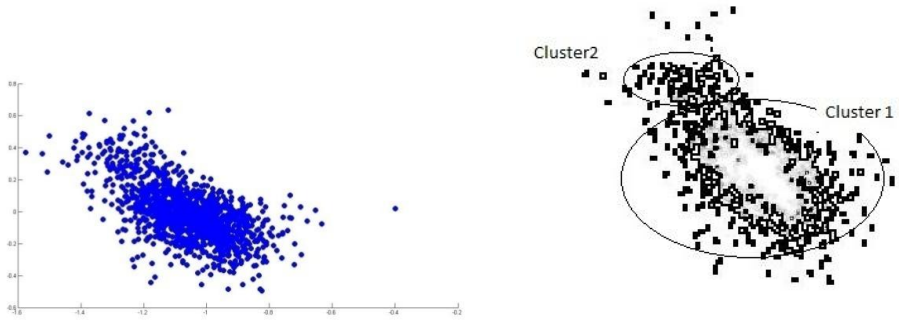


Figure 5-11: (a) 2-D plot of Yeast Dataset. (b) Segmented clusters from 2-D plot

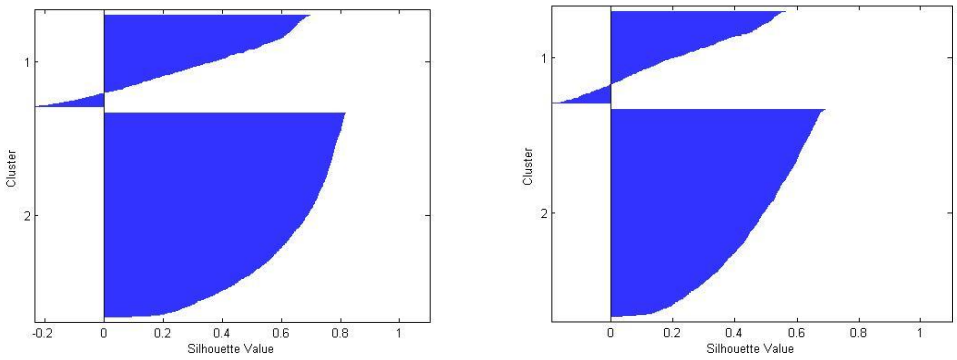


Figure 5-12: (a) Silhouette plot of clusters on reduced yeast data. (b) Silhouette plot on original yeast data.

5.8.3. Wine Dataset

Wine dataset consists of 178 observations of 13 constituents of wine grown in Italy. Three clusters were identified from the data and are plotted below. The silhouette value s= .6747 for 3 clusters shows a good grouping of elements.

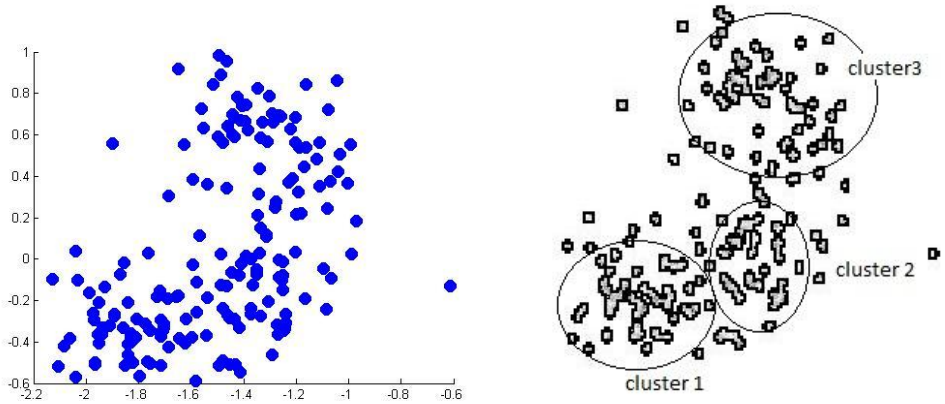


Figure 5-13: (a) 2-D plot of reduced Wine Dataset.(b) segmented clusters from 2D plot of Wine data

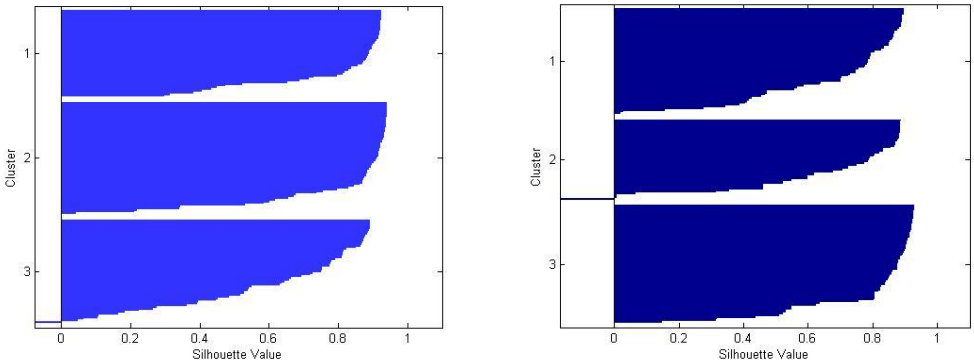


Figure 5: (a) Silhouette plot on reduced wine data. (b) Silhouette plot on original wine data

5.8.4. Thyroid Dataset

The thyroid dataset consists of 7200 observations with 21 attributes. Spatial clustering on the gridded representation of mixed thyroid dataset gives two distinctive clusters well apart from each other. The silhouette value s=.752 also indicates two well differentiated clusters.

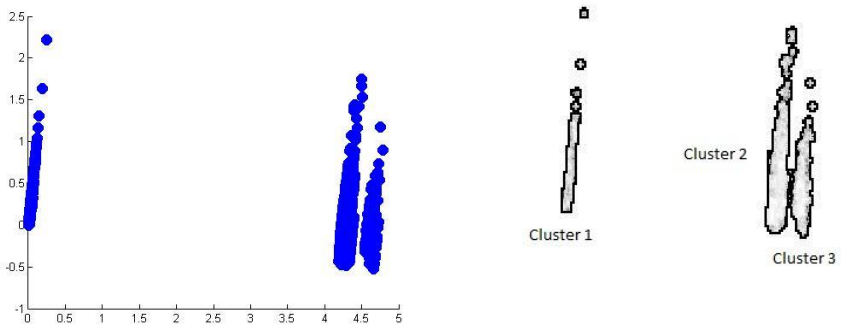


Figure 5-14: (a) 2-D plot of reduced Thyroid Dataset. (b) Segmented clusters from 2-D plot of thyroid dataset.

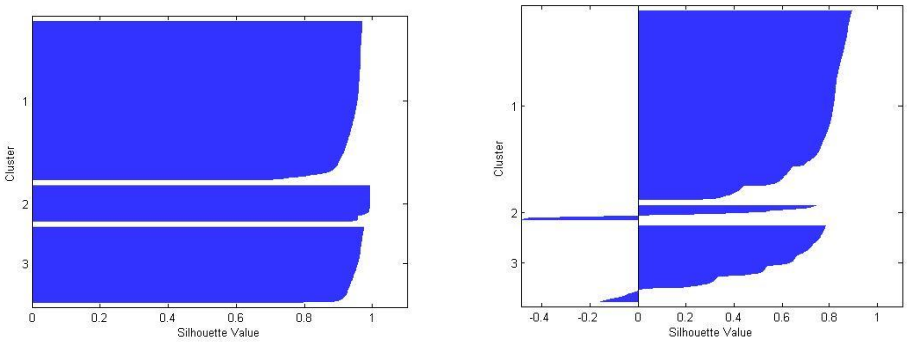


Figure 5-15: (a) Silhouette plot of reduced Thyroid data (b) Silhouette plot of original Thyroid data.

5.8.5. Breast Cancer

Breast cancer dataset consisted of 286 observations with 9 attributes. This dataset consisted of variables categorical in nature and was converted to a numerical equivalent using the extension proposed in the section 3.3.2. Spatial clustering was applied on the gridded representation from the preprocessed attributes and dense quadrants were merged based on the algorithm explained in the section 5.5. A few cluster indices were identified as outliers, but three good grouping can be found out. The silhouette value s=.6528 indicates a positive clustering and is plotted below.

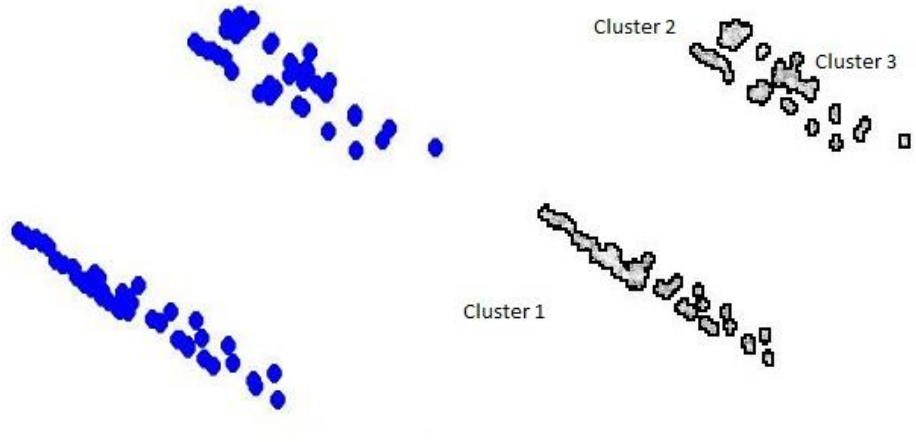


Figure 5-16: (a) 2-D plot of reduced Breast Cancer data. (2) Segmented clusters from 2-d plot of breast cancer data

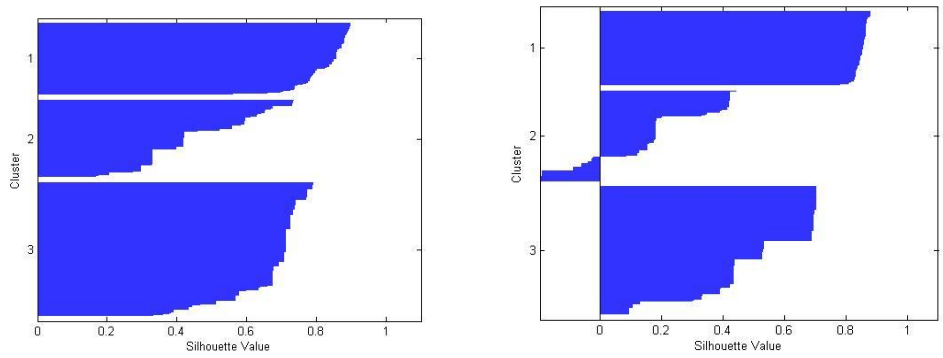


Figure 5-17: (a) Silhouette plot of reduced breast cancer data. (b) Silhouette plot of original breast cancer data.

Following inferences were formed out of the clusters of breast cancer dataset.

(i) Breast cancers diagnosed at the age range of 20-29 are not recurring event. There is an implication of a recurring cancer in those diagnosed at the age of 50-69.

(ii) Most of the recurring cancer diagnosed has a node size of 25-39.

(iii) The number of inverse nodes found in most of recurring cancers ranges from 3 to 8.

(iv) In most of the recurring cancers, an involvement of lymph node capsule is found.

(v) Out of the given cases, all the recurring cancers were irradiated.

5.8.6. Crime Dataset

Crime Dataset is a mixed attributed dataset retaining 600 observations defined by 7 attributes. The clustered image given in the figure 5.19 indicates two distinctive clusters which agree with the results of case study in section 3.5.6. The silhouette plot of crime dataset over two clusters is said in the same section.

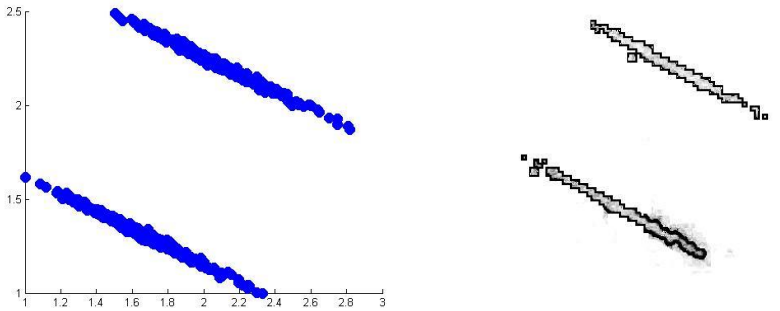


Figure 5-18: (a) 2-D plot of reduced Crime Data. (b) Segmented clusters from 2-D plot of Crime dataset.

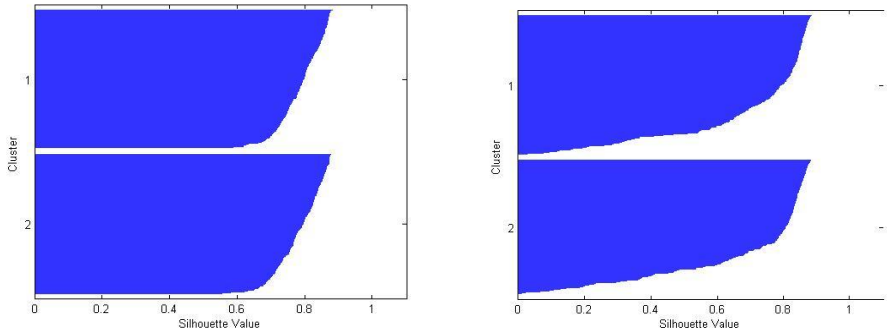


Figure 5-19: (a) silhouette plot clusters of reduced crime dataset (b) silhouette plot clusters of original crime dataset

5.9. Chapter Summary

The silhouette plot in all the 6 cases signifies that reduction in the dimension to lower dimensions has not affected the clustering properties. It is thus established that transformations to lower dimensions based on singular values does not result in any degradation of clustering properties. On the other hand, the gridded representation at lower dimensions paved the way for looking into spatial techniques of clustering. It is also interesting to note that the spatial clustering also helped to generate compact clusters removing some outliers.

Chapter 6

Analysis of the framework with FARS Dataset

.....

Having represented each phase of the structure with the aid of different data sets at each stage, to support the efficacy of the proposed framework as a whole, a study on a mixed attributed and high dimensional data set is demonstrated in this chapter.

.....

6. Analysis of the framework with FARS Dataset

6.1. Introduction to the chapter

The different phases of the framework developed and reported in the initial chapters are experimented and analyzed at each stage. The preprocessing of mixed attributed data into uniform numerical format is tested using the crime dataset in Section 3.4. The gridded representation of the mixed high dimensional data in lower dimensions using SVD is investigated in 6 data sets and the results are projected in Section 4.4. Spatial clustering using quad tree based on fuzzy rule decomposition is applied on the 2-dimensional spatial image of the datasets examined in section and is demonstrated in Section 5.8. To back up the effectiveness of the framework as a whole, an extensive study held on a mixed attributed high dimensional dataset FARS (Fatal Accident Reporting System) is described in this chapter.

6.2. FARS dataset

FARS, formally referred to as the Fatal Accident Reporting System, is a collection of files documenting all qualifying fatal crashes since 1975 that occurred within the 50 States, the District of Columbia, and Puerto Rico, monitored by the US department of transportation [115]. The dataset includes all fatal Accidents on public roads reported to the National Highway Transportation Safety Administration (NHTSA). To be included in this census of crashes, a crash had to involve a motor vehicle traveling on a traffic way customarily open to the public, and must result in the death of a person (occupant of a vehicle or a non-motorist) within 30 days of the crash. FARS dataset consists of three principal files; Accident, Vehicle, and Person. This research is applied and analyzed on Accident section of FARS dataset to infer any hidden correlation among variables which may be significant to the cause of accident.

FARS Accident variables can be classified into three categories; geographical and time related variables like latitude and longitude of the crash site, variables related to the crash site and accident and post-accident data like reporting details. The second category of variables specify details like the atmospheric conditions, road and traffic specifications, specifics about the details of human being involved etc. For example a crash is alcohol-involved if a driver, pedestrian, or pedal cyclist involved in the crash has (1) police-reported alcohol involvement, or (2) a positive alcohol test result. Atmospheric conditions like fog or rain or wet surface may have acted a significant role in the cause of accident. Any traffic flow control or alignment of the road or road regulations like speed limit etc. is also included in the second classification of variables. Since the research is more focused on the crash details, we have omitted the first and third categories of variables from further study.

6.2.1. FARS Variables

Sl.No	Attribute Name	Description	Details of Attribute
1	Weather	Atmospheric Conditions	1- No Adverse Atmospheric Conditions 2 - Rain (Mist) 3 - Sleet (Hail) 4 - Snow 5 - Fog 6 - Rain and Fog 7 - Sleet and Fog 8 - Other: Smog, Smoke, Blowing Sand or Dust 9 - Unknown
2	C_M_ZONE	Construction/Maintenance Zone	0 - None 1- Construction 2 - Maintenance 3 - Utility 4 - Work Zone, Type Unknown
3	DRUNK_DR	Drunken Driver	The number of drunk drivers involved in the fatal crash
4	SP_LIMIT	Speed Limit	Speed limit at the crash site.
5	Hit-and-Run	Hit and Run Case	0 - No Hit and run 1 - Hit Motor Vehicle in

			Transport 2 - Hit Pedestrian or Non-motorist 3 - Hit Parked Vehicle 4 - Occupant Is Struck by or Fell From Own Hit-and-Run Vehicle 5 – Hit-and Run, Other Involved Person Left Scene
6	LGT_COND	Light Condition at the crash site	1 - Daylight 2 - Dark 3 - Dark but Lighted 4 - Dawn 5 - Dusk 9 – Unknown
7	NO_LANES	Number of lanes at the crash site	1 - One lane 2 - Two lanes 3 - Three lanes 4 - Four lanes 5 - Five lanes 6 - Six lanes 7 - Seven or more lanes 9 – Unknown
8	REL_JUNC	Related to a Junction	00 - None 01 - Non-Junction 02 - Intersection 03 - Intersection-Related 04 - Driveway, Alley Access, etc. 05 - Entrance/Exit Ramp-Related 06 - Rail Grade Crossing 07 - In Crossover 08 - Driveway-Access-Related 09 - Unknown - Non-Interchange 10 - Intersection 11 - Intersection-Related 12 - Driveway Access 13 - Entrance/Exit Ramp-Related 14 - In Crossover 15 - Other Location in IC 19 - Unknown, Interchange Area 99 – Unknown
9	TRAFFIC_CONTROL	Related to Traffic way Controls	0 - No Controls 1 - Device Not Functioning 2 - Device Functioning improperly

			3 - Device Functioning improperly 9 - unknown
10	ALIGNMNT	Roadway Alignment	1 - Straight 2 - Curved 9 – Unknown
11	PROFILE	Roadway Profile	1 - Level 2 - Grade 3 - Hill crest 4 - Sag 9 – Unknown
12	SUR_COND	Road Surface Condition	1 - Dry 2 - Wet 3 - Snow or Slush 4 - Ice 5 - Sand, Dirt, Oil 8 - Other 9 – Unknown
13	SCH_BUS	School Bus Related	0- No 1- Yes
14	SP_JUR	Special jurisdiction	0 - No Special Jurisdiction 1 - National Park Service 2 - Military 3 - Indian Reservation 4 - College/University Campus 5 - Other Federal Properties 8 - Other 9 – Unknown
15	TRAF_FLO	Traffic way Flow	1 - Not Physically Divided 2 - Divided Highway 3 - Divided Highway 4 - One-Way Traffic way 5 - Divided Highway 9 – Unknown
16	PEDS	Pedestrians involved	Number of pedestrians involved

Table 6-1: List of Attributes in Accident Dataset chosen for the study.

6.3. Preprocessing FARS

The FARS accident dataset contains 55 variables including all the above mentioned categories. Out of 55 variables, those variables which state the physical details of crash are chosen for the study and include 16 variables consisting of 12

categorical and 4 numerical attributes. All others except NO_LANES, DRUNK_DR, FATALS, and PEDS are categorical in nature. More than ten thousand observations are available in the FARS Dataset. In order to simplify the clustering process, sampling is done on the dataset. A random sampling was done on the population and the records for the study were selected. Thus the sample consisted of 20 percent of the original number of observations and is around more than 7400 observations.

The numerical coding listed in the table 6-1 against each variable is not used in the study and it is retained as short form of categorical explanation itself. No missing values were observed in the Dataset.

6.3.1. Preprocessing mixed attributes using co-occurrence

The first phase of the framework is to preprocess the data into uniform format using the notion of co-occurrence explained in the chapter 3. The numerical attributes are normalized using the min-max normalization into a range of [0 1] in order to avoid any bias on large values over smaller domains. TRAF_FLO is chosen as the base attribute since; it has got the maximum number of items. The similarity matrix is constructed based on the equation 3-8, $D_{xy} = \frac{|m(X,Y)|}{|m(X)|+|m(Y)|-|m(X,Y)|}$, whereby D represents the similarity between the categorical items; higher the value, higher the similarity. The numerical attribute with the minimum group variance is NO_LANES. Every base item in the base attribute is quantified by assigning mean of the mapping value in the selected numeric attribute. All non-base items can be quantified by applying the following formula $F(x) = \sum_{i=1}^{d} a_i * v_i$ as given by Equation 3-10.

6.4. Gridded representation of FARS

The output of the first phase of the framework is a numerical dataset equivalent to the original dataset constructed on the basis of the notion of co-occurrence. . In order to visualize this high-dimensional numerical dataset, we adopt the strategy specified in chapter 4. The dimensionality of the FARS is reduced to 3 dimensional dataset using the algorithm explained in algorithm 4-2. The numerical dataset is split into U,∑, V matrix by applying singular value decomposition on the transposed dataset.

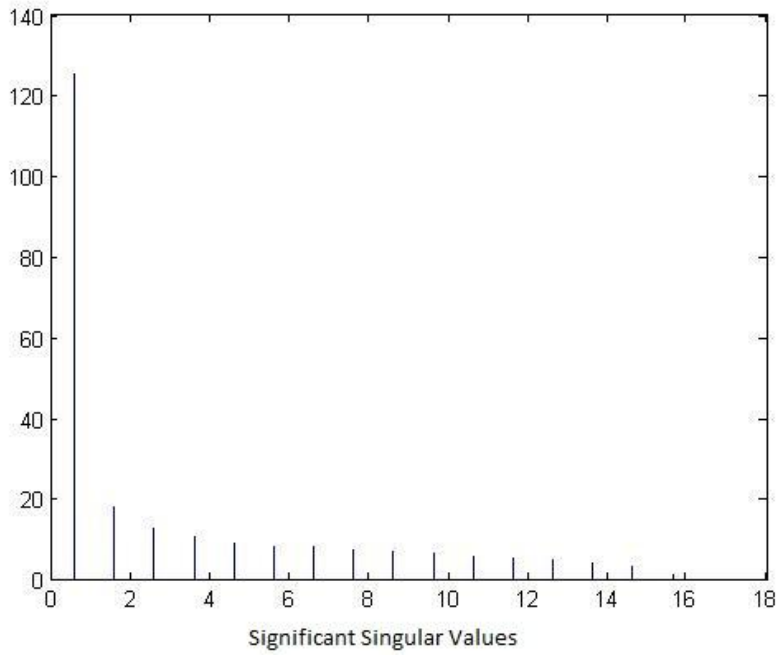


Figure 6-1: Singular value plot of FARS Data {125.23, 18.23, 12.81, 10.95, 9.24, 8.6, 8.3, 7.6, 7.19, 6.69, 6.15, 5.62, 5.27, 4.33, 3.48, 1.57, 0, 0,…}

U_{reduce} is constructed using the first 3 columns of U against the most significant singular values {68.52, 9.33, 6.84}. The dot product of original dataset with U_{reduce} gives D_{reduce}, which is the reduced dataset with 3 dimensions. To plot the new values is a positive quadrant; the values in D_{reduce} are shifted to positive axes. The

2-Dimensional representation of the FARS can be constructed by plotting the first column against the second column of $D_{reduce.}$ The 3-dimensional representation can be obtained using the third column of D_{reduce} and plotted against z axis. The 2-dimensional and 3-dimensional representation of sampled FARS dataset is given below in the figure 6-1 and figure 6-2 respectively.

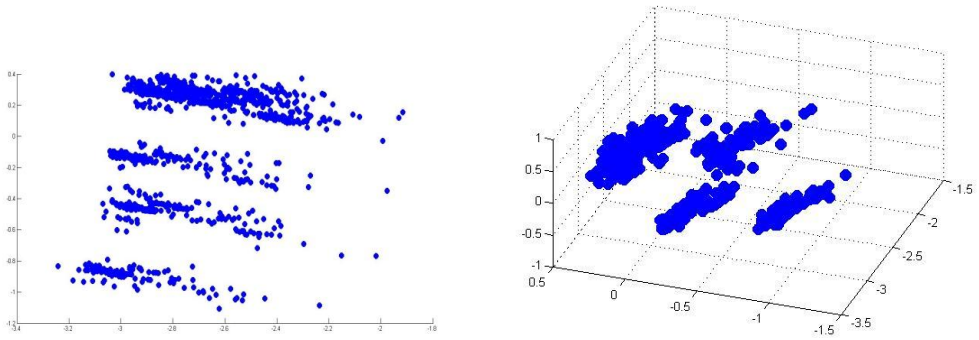


Figure 6-2: (a) 2-dimensional representation of FARS Dataset against singular values {68.52, 9.33} (b) 3-Dimensional Representation of FARS Dataset against {68.52, 9.33, 6.84}

6.5. Spatial clustering on FARS using Quad tree based on Fuzzy rules

To identify clusters from the gridded representation of FARS dataset, spatial data structure quad tree is used. The decomposition of quad tree based on fuzzy rules is explained in section 5.4.3. The quadrants were recursively decomposed up to until they are homogenous or up to a maximum level of [4 x4] pixel array, which is a preset threshold. Clusters were formed by merging the neighbouring information dense quadrants. Omitting certain cluster indices as outliers, the study could lead to 2 major clusters from the FARS dataset. The merged quadrants are shown in the figure 6-3.

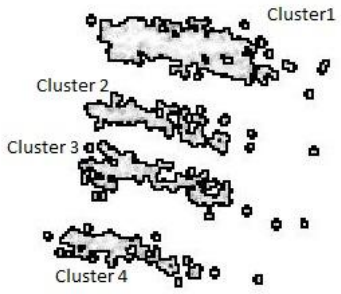


Figure 6-3: Spatial clustering FARS Dataset

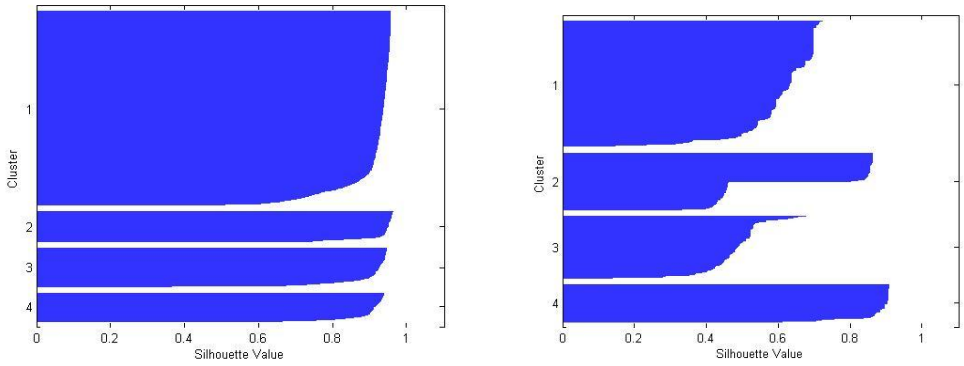


Figure 6-4: (a) Silhouette plot against reduced FARS Data(s=0.9087) (b) Silhouette plot against original FARS Data (s=0.622)

6.6. Re-mapping the cluster indices into the sample

As a result of the spatial clustering on the 2-d gridded representation of the FARS data, the study could isolate 4 distinctive clusters omitting insignificant outliers. 66 percent of the observations belong to cluster 1, 13 percent to cluster 2, 10 percent to cluster 3 and almost 11 percent to cluster 4. The distribution of records over clusters is given in the figure 6-5.

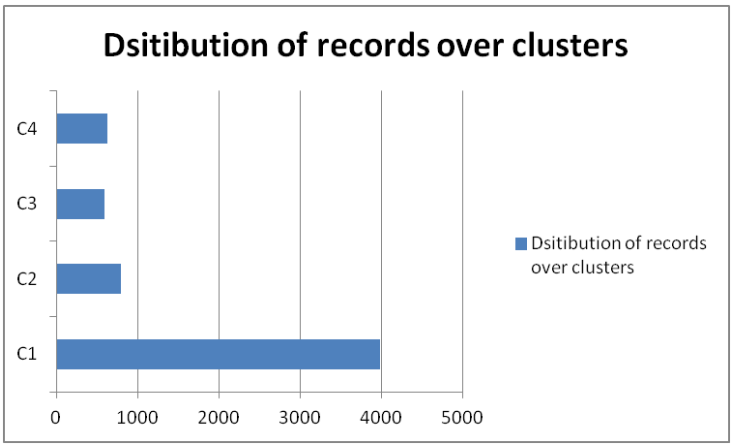


Figure 6-5: Distribution of records over clusters.

6.7. Conclusions from the study

The attribute wise distribution over clusters is projected and plotted as Figure 6-6 (a) to (j). Each clusters are colour coded as below: Cluster 1 : blue, cluster 2: green, cluster 3: red, cluster 4: brown. . From the figures, following inferences are made out of the grouping of records.

(i) there were four clusters formed viz. 66 percent of the observations belong to cluster 1, 13 percent to cluster 2, 10 percent to cluster 3 and almost 11 percent to cluster 4.

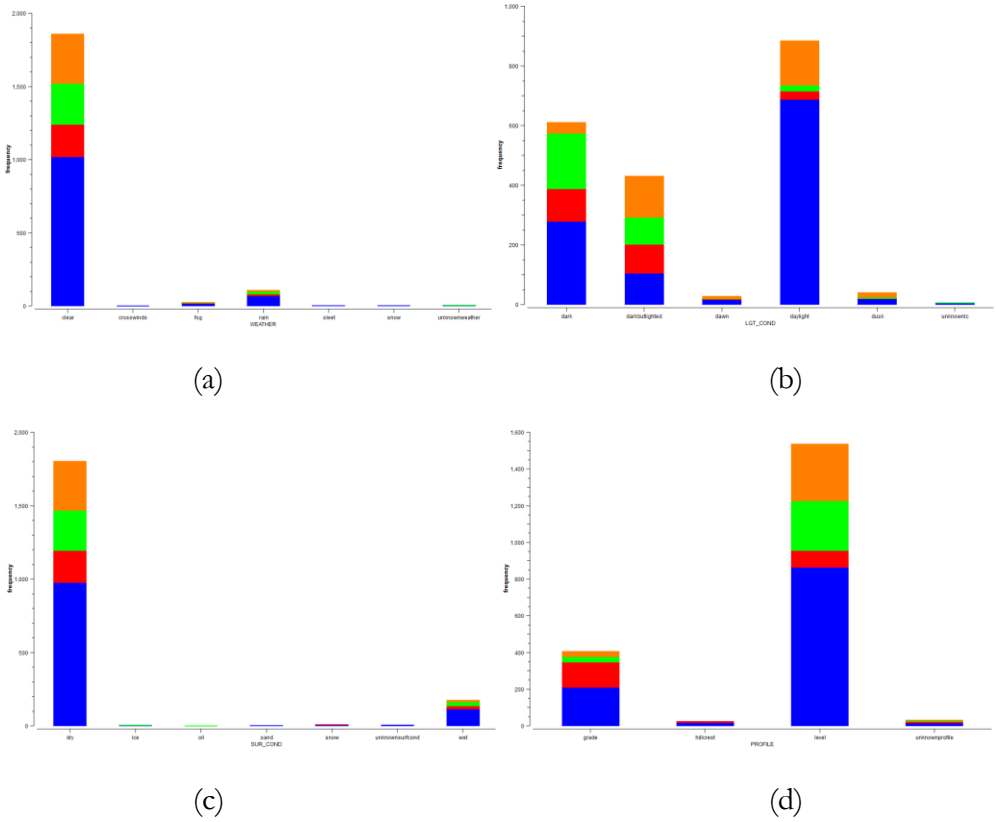
(ii) A good number of accidents occurred, when the weather was clear, in daylight, where the surface condition of the road was dry and on a level profile, but where there was no traffic controls and the number of lanes were more than 1. This feature was evident from cluster 1

(iii) In most of the cases of accidents happened in an intersection or intersection related places, the light condition was dark or dark but lighted and the traffic system were working properly, based on cluster 2.

(iv) Most of the accidents happened at places where speed is limited from 40 miles/ hour to 60 miles/ hour, based on cluster 3.

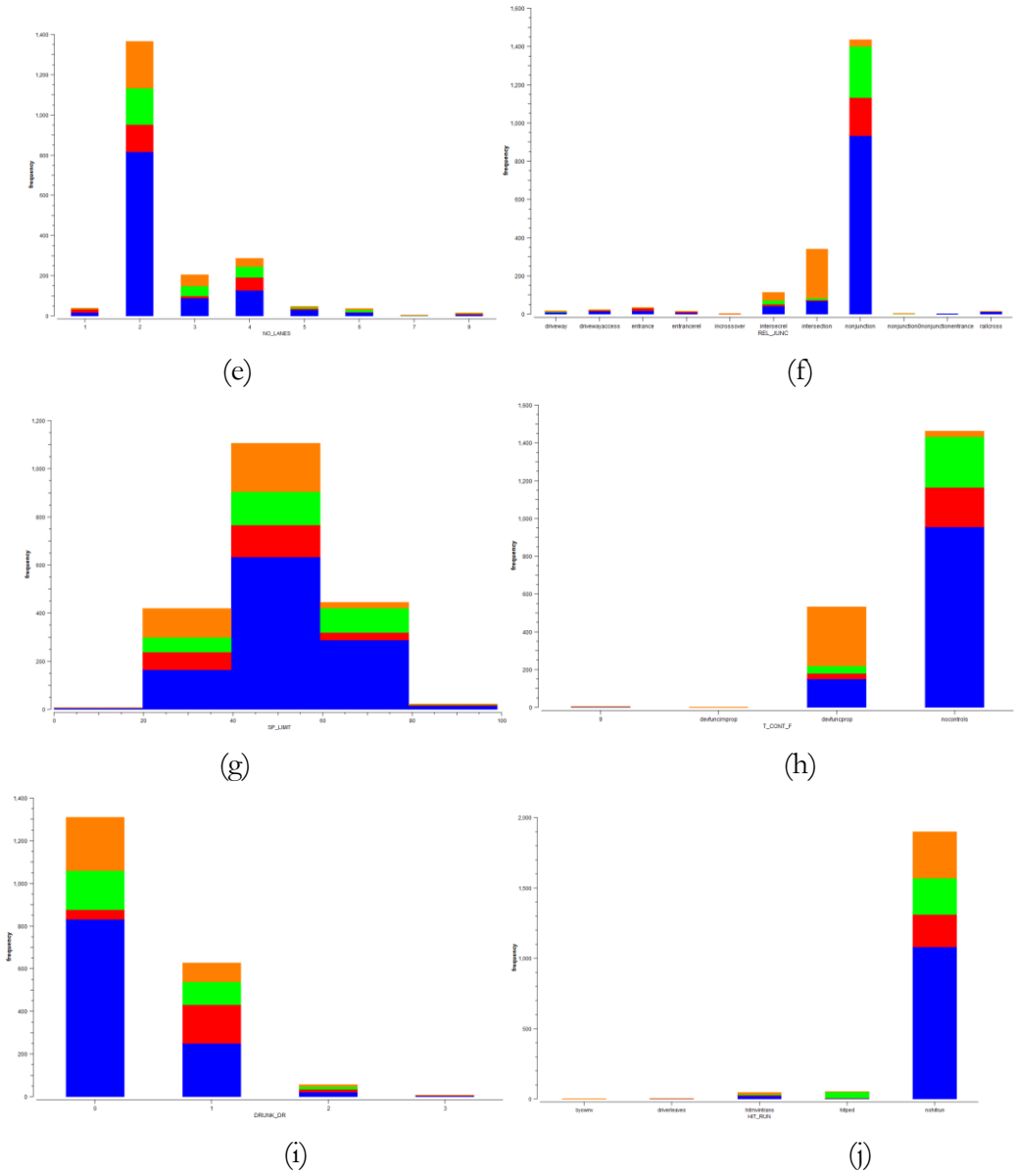
(v) Most of the accidents in which drunken drivers are involved, the light conditions were dark or dark, but lighted, based on cluster 3.

(vi) The accidents which hit a pedestrian mostly happened on dark light conditions, based on cluster 4.



(a) (b)

(c) (d)



(e) (f)

(g) (h)

(i) (j)

Figure 6-6 Influence of variables on road accidents

(a) Distribution of Weather; (b) Distribution of light condition; (c) Distribution of Surface condition (d) Distribution of road profile; (e) Distribution of Number of lanes,(f) Distribution of Junction-related; (g) Distribution of Speed limit; (h) distribution of Traffic controls; (i) Distribution of Drunken drivers; (j) Distribution of Hit and Run

6.8. Chapter Summary

The present chapter provided the analysis of FARS data set extensively using the techniques developed in the thesis. The multi category data could be efficiently represented in 2 D using the SVD and clustered using the spatial clustering technique proposed in the thesis. The properties of the cluster formed at the 2 D level brought out many interesting features accounting for accidents.

Chapter 7

Conclusion and Future Scope

.....

This chapter wraps up the findings of the proposed framework for clustering mixed high dimensional data on various data sets and gives directions for any future research.

.....

7. Conclusions and Future Scope

7.1. Conclusion of the research

The thesis has focused on the representation and visualization of multi dimensional mixed category data on lower dimensional space, thereby ensuring the representational characteristics of original data and facilitating the utilization of spatial clustering algorithms. In addressing the visualization of mixed category data, it is observed that many of existing data mining techniques like clustering centers around data sets of a single attribute which is numerical.

The present thesis therefore examined the existing algorithms for converting mixed category data sets into numerical data in detail. The algorithm due to Ming_Yi Shih et.al. , based on the co-occurrence matrix was examined in detail to note that by choosing a categorical entity in the record as the base attribute, a numerical attribute having a minimum variance with the base attribute is identified. Each base item is quantified with the mean value corresponding to the chosen numerical attribute. All non base items are quantified by computing the similarity with respect to the base item and entities are also converted into numeric attributes. After extensive study of the algorithm, it was concluded that for handling the mixed category data for the rest of the thesis, the said algorithm could be used. In order to test the efficacy of the algorithm for clustering, a crime data set with mixed attributes were simulated based on the inputs taken from accessible crime records. The data set consists of unique -id, crime name, weapons used if any, and type of crime. Other attributes like victim gender, victim area, victim age, crime location were also included, while generating the data set. The cluster number was identified using agglomerative clustering, followed by K-means clustering. The resulting dendrogram, scatter matrix plot and pair wise plot of various attributes, confirms the correctness

of the clustering approaches on crime data set (Figure 3.2 to 3.5). The cophenetic coefficient and the silhouette plot bring out the

(i) the efficacy of representing multi category data in a numerical format and

(ii) Validity of using the same for K- means clustering.

However as an extension of the work of Ming_Yi Shih et.al, the thesis has developed an algorithm for handling data which is entirely categorical, by introducing the concept of frequency in the category data set. The representation was subsequently demonstrated in lower dimensions in Chapter 4 and spatially clustered in chapter 5. The inferences from the clusters from all categorical breast cancer data were clearly listed out in section 5.8.5.

In all data mining efforts, it is well known that visualizing high dimensional data is a major requirement. After a careful study of the techniques reported in literature which handles visualization using scatter plot, parallel coordinates, survey plots, pixel oriented techniques and icon based techniques, it was concluded that most of the reported techniques are data dependent and the visualization demands further domain specific support for clarity. On the other hand, the techniques based on Eigen decomposition, for points in a higher dimensional space give effective representation. Since the representation of the mixed attribute data and categorical data finally represents the data asset as appoint in higher dimensional space, the techniques of singular value decomposition was applied for representation in reduced dimensions(2 and 3). The relative predominance of the singular values suggested itself the possibility of projecting the data set to lower dimensions corresponding to prominent singular values. The technique was applied to the flowing data sets :

Data type	**Size**	**Dimensions**	**Numeric/mixed/categorical**
Iris	150	4	Numerical

Yeast	1484	8	Numerical
Wine	178	13	Numerical
Breast Cancer	286	9	All Categorical
Thyroid	9172	29	Mixed
Crime	600	7	Mixed

The similarity of data set reconstructed from reduced presentation was computed with the original data set (all converted to numerical attributes).

The representation in lower dimensions opened the way for gridded representation of data, which in turn facilitated the utilization of spatial clustering algorithms. The spatial clustering algorithms, which are based on segmentation techniques, have the advantage of removing the outliers of the clusters. Among the many algorithms like k-D tree, R- tree and Quad tree available for spatial clustering, the Quad tree based algorithm was chosen because of the systematic nature of the decomposition of the space. The quad tree algorithm is based on decomposition based on the homogeneity of the quadrant, at any level The present work has examined the definition of homogeneity based on mean and variance of the region. In order to alleviate the problem of square tessellation in the boundaries of the clustered regions, a Fuzzy rule base system was used to look at properties like low variance, medium and high variance . Subsequently after merging neighbouring dense regions, clusters were formed. The resulting clusters showed smooth boundaries. (Fig. 5.9 to 5.19) As was expected, the outliers were getting eliminated from the clusters. The silhouette plots of the clusters formed out of reduced dimension matched very well with the silhouette plots of original data in clustered in higher dimensions. It is also interesting to note that the representation in lower dimensions have not degraded the original properties of the data in higher dimensions viz.

(i) formation of clusters and

(ii) Relative sizes and the coverage of clusters.

On the other hand, the clustering in reduced dimensions has helped to eliminate the outliers, which in turn helps in the definition of clusters

A case study on the FARS data set [(FARS Analytic Reference Guide 1975 to 2007)] was carried in corporating all the techniques developed thus far viz.

(i) multi category data representation based on occurrence matrix

(ii) dimensionality reduction to 2 and 3 dimensions using SVD and

(iii) Spatial clustering using quad trees based fuzzy rules for defining homogeneity.

The major features of the study presented in the thesis were fully demonstrated, using the FARS data having 55 variables of mixed category. Though 37,000 records were available in the FARS data set, the study was limited to 6000 records, chosen through random sampling. The conversion to numeric category was done using the method of co-occurrence and the representation using the Singular value decomposition. The spatial clustering using the quad trees with fuzzy rules for defining homogeneity finally resulted in the data set with most of the outliers eliminated. The clusters formed were mapped back to the original data set to observe the inferences given in Section 6.7

Thus the results presented in the thesis have effectively brought out the following aspects:

(i) Conversion to numeric values of mixed category data effectively helps to apply statistical techniques in clustering multi category data. Extension to the techniques to handle only categorical data by converting the same to numerical data also supported the statistical techniques.

(ii) Reduced representation using techniques like SVD does result in the loss

of information as far as properties like clustering are concerned. The silhouette plot and the cophenetic factor computed on clusters formed out of the reduced and the original data set underscored this fact.

(iii) Representation in 2 D helps to effectively apply the spatial clustering techniques, with the added advantage of eliminating outliers. The idea of using Fuzzy set to define homogeneity required in the quad tree decomposition, helped to mark the cluster boundaries effectively.

The large number of case studies has revealed the confirmation of the facts brought out above.

7.2. Directions for further work

The work reported in the thesis has opened up the whole area of spatial clustering to multi dimensional multi category data set. More techniques based on image representation and analysis can be used to further amplify this direction. The idea of defining homogeneity using entropy is a fertile filed to be explored to the quad tree or oct tree based decomposition of data represented in lower dimensions. While there are techniques to estimate parameters using entropy maximization, further exploration in this direction could look into defining and estimating a parameter that could uniquely define the extent of homogeneity. Facility to list out exemplar cases corresponding to the clusters formed will go a long way in drawing effective conclusions from large and complex data, with multiple attributes.

List of papers published from the thesis

International journals

1. Bindiya M Varghese, Unnikrishnan A, Poulose Jacob, K, "Spatial Clustering Algorithms- An Overview", Asian Journal of Computer Science And Information Technology 3: 1 (2013) 1 - 8
2. Bindiya M Varghese, Jose Tomy, Unnikrishnan A, Poulose Jacob K, " Clustering Student Data to Characterize Performance Patterns", Special Issue on Artificial Intelligence, IJACSA, 0(3), 138 - 140.
3. Bindiya M Varghese, Unnikrishnan A, Poulose Jacob K, ,Justin Jacob, "Correlation Clustering Model for Crime Pattern Detection", IJACT : International Journal of Advancements in Computing Technology, Vol. 2, No. 5, pp. 125 ~ 128, 2010.
4. Bindiya M Varghese, Unnikrishnan A, Poulose Jacob K , " 2-D and 3-D representation of mixed sequenced dataset" , Accepted for Publication in International Journal of Information Processing and Management.
5. Bindiya M Varghese, A. Unnikrishnan, Paulose Jacob. K, "H-K Clustering on mixed categorical and numeric crime dataset for crime pattern detection" (Communicated to The International Arab Journal of Information Technology)

Proceedings of International Conferences

6. Book Chapter, "Enhanced Spatial Mining Algorithm Using Fuzzy Quadtrees", Bindiya M. Varghese, A. Unnikrishnan and K. Poulose Jacob, Communications in Computer and Information Science, 1, Volume 250,

Computational Intelligence and Information Technology, Part 1, Pages 110-116, Springer. ISBN- 978-3-642-25734-6_17

7. Bindiya M Varghese, Unnikrishnan A, Poulose Jacob K, " Information Content Extraction on Quad Trees for Active Spatial Image Clustering" , World Congress of Computer Science Information Engineering, IEEE Computer Society Press, CSIE (4) 2009: 306-310.
8. Bindiya M Varghese, Unnikrishnan A, "Recursive Decision tree induction based on homogeneousness for data clustering", Proceedings of International Conference on Cyber worlds, IEEE Computer Society Press, China, CW 2008: 754-758.

Minor Research project

9. Project Report, Enhanced Spatial Mining Algorithm Using Image Extraction on Quad trees. Computer Society of India. File No : 1-14/2009-04, March 2009.

References

1 Usama Fayyad, Gregory Piatetsky-Shapiro, Padhraic Smyth. From Data Mining to Knowledge Discovery in Databases. American Association for Artificial Intelligence. 1996.

2 Usama Fayyad, Gregory Piatetsky-shapiro,Padhraic Smyth. Knowledge Discovery and Data Mining: Towards a Unifying Framework. 1996. p. 82-88.

3 Ester, M., Frommelt, A., Kriegel, H.-P., and Sander, J. Algorithms for characterization and trend detection in spatial databases. In: Int. Conf. on Knowledge Discovery and Data Mining; 1998; New York City, NY. p. 44-50.

4 Rui Xu, Don Wunsch. Clustering. IEEE Computer Society; 2008.

5 MacQueen J. Some Methods for classification and Analysis of Multivariate observations. In: Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability; 1957. p. 281-297.

6 Kaufman, L. and Rousseeuw, P.J. Clustering by means of Medoids, in Statistical Data Analysis Based on the L1–Norm and Related Methods. North Holland 1987.

7 Ng, Raymond T. and Jiawei Han. Efficient and Effective Clustering Methods for Spatial Data Mining. In: Proceedings of the 20th International Conference on Very Large Data Bases.; 1994; San Francisco, CA. p. 144-155.

8 M. Ester, H.-P. Kriegel, S. Jörg, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining ; 1996. p. 226-231.

9 Mohamed A. Mahfouz, d M. A. Ismail. Fuzzy Relatives of the CLARANS Algorithm With Application to Text Clustering. International Journal of Electrical and Computer Engineering. 2009 370-377.

10 S.Vijayarani, S.Nithya. An Efficient Clustering Algorithm forOutlier Detection. International Journal of Computer Applications. 2011 22-27.

11 Jain AK. 50 years beyond k means. Pattern Recoginition letters. 2009.

12 Jay G Wilpon, Lawarence R Rabiner. A Modified K-Means Clustering A Algorithm for Use in Isolated Work' Recognition. IEEE TRANSACTIONS ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING. 1985 587-594.

13 Sara Nasser, Rawan Alkhaldi, Gregory Vert. A Modified Fuzzy K-means Clustering using Expectation Maximization. [Internet].

14 Adil M. Bagirov, Karim Mardaneh. Modified global k-means algorithm for clustering in gene expression [Internet].

15 Phillips SJ. Acceleration of K-Means and Related Clustering Algorithms. Algorithm Engineering and Experiments. 2002 166-177.

16 D.Charalampidis. A modified K-means algorithm for circular invariant clustering. IEEE Transacton on Pattern Analysis and Machine Intelligence. 2005 1856-65.

17 Grigorios F. Tzortzis ,Aristidis C. Likas. The Global Kernel K-Means Algorithm forClustering in Feature Space. IEEE TRANSACTIONS ON NEURAL NETWORKS. 2009 1181-1193.

18 Pushpa.R. Suri, Mahak. Image Segmentation With Modified K-Means Clustering Method. International Journal of Recent Technology and Engineering. 2012 176-179.

19 van der Laan, Mark J.; Pollard, Katherine S.; and Bryan, Jennifer. A New Partitioning Around Medoids Algorithm. Journal of Statistical Computation and Simulation. 2002 575-584.

20 Lamiaa Fattouh Ibrahim, Manal Hamed Al Harbi. Using Modified Partitioning Around Medoids Clustering Technique in Mobile Network Planning. CoRR. 2013;abs/1302.6602.

21 V.B. Nikam,Vinod J. Kadam,B. B. Meshram. Image Compression Using

Partitioning Around Medoids Clustering Algorithm. IJCSI International Journal of Computer Science Issues. 2011 399-401.

22 Pakhira MK. Fast Image Segmentation Using Modified CLARA Algorithm. In: International conference on information Technology; 2008. p. 14-18.

23 Rao, Y.V. Ramana and Abirami, S. A Fuzzy C-Medoids-Based CLARA Algorithm for Fast Image Segmentation. The IUP Journal of Computer Sciences. 2012 7-16.

24 Jiawei Han , Yandong Cai , Nick Cercone. Knowledge Discovery in Databases: An Attribute-Oriented Approach. In: Proceedings of the 18th International Conference on Very Large Data Bases; 1992. p. 547-559.

25 Larry Anderson,Jay L.Weiner. Actionable Market Segmentation Guaranteed. Ipsos-Insight. 2004.

26 Sudipto Guha , Rajeev Rastogi , Kyuseok Shim. CURE: an efficient clustering algorithm for large databases. In: Proceedings of the 1998 ACM SIGMOD international conference on Management of data; 1998; Seattle. p. 73-84.

27 Sudipto Guha, Rajeev Rastogi, Kyuseok Shim. ROCK: A Robust Clustering Algorithm for Categorical Attributes. In: Proceedings of the 15th international Conference on Data Engineering ; 1999. p. 512-521.

28 George Karypis , Eui-Hong (Sam) Han , Vipin Kumar. Chameleon: Hierarchical Clustering Using Dynamic Modeling. Computer. 1999;32:68-75.

29 Wei Wang , Jiong Yang , Richard R. Muntz. STING: A Statistical Information Grid Approach to Spatial Data Mining. In: Proceedings of the 23rd International Conference on Very Large Data Bases; 1997. p. 186-195.

30 Wei Wang, Jiong Yang, Richard Muntz. STING+: An Approach to Active Spatial Data Mining. In: Proceedings of the 15th International Conference on Data Engineering; 1999. p. 116-125.

31 Tian Zhang , Raghu Ramakrishnan , Miron Livny. BIRCH: an efficient data clustering method for very large databases. In: Proceedings of the 1996 ACM SIGMOD international conference on Management of data; 1996; Montreal,

Canada. p. 103-114.

32 Gholamhosein Sheikholeslami , Surojit Chatterjee , Aidong Zhang. WaveCluster: a wavelet-based clustering approach for spatial data in very large databases. The VLDB Journal — The International Journal on Very Large Data Bases. 2000;8(3-4):289-304.

33 Erich Schikuta , Martin Erhart. The BANG-Clustering System: Grid-Based Data Analysis. In: Proceedings of the Second International Symposium on Advances in Intelligent Data Analysis,Reasoning about Data; 1997. p. 513-524.

34 E. Schikuta. Grid-Clustering: An Efficient Hierarchical Clustering Method for Very Large Data Sets. In: Proceedings of the 13th International Conference on Pattern Recognition; 1996. p. 101-105.

35 Rakesh Agrawal , Johannes Gehrke , Dimitrios Gunopulos , Prabhakar Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. In: Proceedings of the 1998 ACM SIGMOD international conference on Management of data; 1998; Seattle, Washington, United States. p. 94-105.

36 Jiyeon Choo, Rachsuda Jiamthapthaksin , Chun-sheng Chen, Oner Ulvi Celepcikay, Christian Giusti and Christoph F. Eick. C.F.: MOSAIC: A proximity graph approach for agglomerative clustering. In: International Conferenceon Data Warehousing and Knowledge Discovery; 2007.

37 Bernard Chen , Phang C. Tai , R. Harrison and Yi Pan. Novel Hybrid Hierarchical-K-means Clustering Method (H-K-means) fo rMicroarray Analysis. In: IEEE Computational Systems Bioinformatics Conference Workshops; 2005.

38 Murtagh F. A Survey of Recent Advances in Hierarchical Clustering Algorithms. The Computer Journal. 1983 354-359.

39 William H. E. Day, Herbert Edelsbrunner. Efficient algorithms for agglomerative hierarchical clustering methods. Journal of Classification. 1984 7-24.

40 Ying Zhao , George Karypis. Evaluation of hierarchical clustering algorithms for document datasets. In: Proceedings of the eleventh international conference on

Information and knowledge management; 2002. p. 515-524.

41 Sibson R. SLINK: an optimally efficient algorithm for the single-link cluster method. The Computer Journal. 1973.

42 Defays D. An efficient algorithm for a complete link method. The Computer Journal. 1977.

43 White, S. D. M, Frenk, C. S. Galaxy formation through hierarchical clustering. Astrophysical Journal. 1991 52-79.

44 Bandyopadhyay S. An energy efficient hierarchical clustering algorithm for wireless sensor networks. In: INFOCOM 2003; 2003. p. 1713-1723.

45 Katherine A. Heller, Zoubin Ghahramani. Bayesian Hierarchical Clustering [Internet].

46 Yih-Jen Horng, Shyi-Ming Chen, Yu-Chuan Chang, Chia-Hoang Lee. A New Method for Fuzzy Information Retrieval Based on Fuzzy Hierarchical Clustering and Fuzzy Inference Techniques. IEEE TRANSACTIONS ON FUZZY SYSTEMS. 2005 216-228.

47 Shivakumar Vaithyanathan, Byron E Dom. Model-Based Hierarchical Clustering. In: Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence ; 2013.

48 Jiewei Han, Micheline Kamber. Data Mining Concepts and Techniques. Elsevier; 2006.

49 M. Ester, H.-P. Kriegel, S. Jörg, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96); 1996. p. 226-231.

50 Jörg Sander , Martin Ester , Hans-Peter Kriegel , Xiaowei Xu. Density-Based Clustering in Spatial Databases: The Algorithm GDBSCAN and Its Applications. Data Mining and Knowledge Discovery. 1998;2(2):169-194.

51 Ankerst, Mihael, Markus M. Breunig, Hans-Peter Kriegel, and Jörg Sander. OPTICS: ordering points to identify the clustering structure. In: SIGMOD Rec.

28 ; 1999. p. 49-60.

52 Xiaowei Xu , Martin Ester , Hans-Peter Kriegel , Jörg Sander. Distribution-Based Clustering Algorithm for Mining in Large Spatial Databases. In: Proceedings of the Fourteenth International Conference on Data Engineering; 1998. p. 324-331.

53 Dash, M., Huan Liu , Xiaowei Xu. '1+1>2': merging distance and density based clustering. In: Database Systems for Advanced Applications, 2001. Proceedings. Seventh International Conference on; 2001. p. 32-39.

54 Qixiang Ye ; Wen Gao ; Wei Zeng. Color image segmentation using density-based clustering. In: ICASSP '03; 2003. p. 345-348.

55 Daxin Jiang, Jian Pei , Aidong Zhang. DHC: a density-based hierarchical clustering method for time series gene expression data. In: Bioinformatics and Bioengineering, 2003. Proceedings. Third IEEE Symposium on; 2003. p. 393-400.

56 Eshref Januzaj, Hans-Peter Kriegel, Martin Pfeifle. DBDC: Density Based Distributed Clustering. Advances in Database Technology. 2004 88-105.

57 Kriegel, H.-P, Pfeifle M. Hierarchical density-based clustering of uncertain data. In: Data Mining, Fifth IEEE International Conference on; 2005.

58 A local-density based spatial clustering algorithm with noise. Information Systems. 2007 978-986.

59 MacKay D. Information Theory, Inference, and Learning Algorithms. Cambridge University press; 2003.

60 A. Hinneburg and D. A. Keim. An Efficient Approach to Clustering in Large Multimedia Databases with Noise. In: International Conference on Knowledge Discovery and Data Mining; 1998. p. 58-65.

61 S. Kim, X. Jin, and J. Han. SpaRClus: Spatial Relationship Pattern-Based Hierarchical Clustering. In: SIAM International Conference on Data Mining - SDM; 2008. p. 49-60.

62 Nanopoulos, A., Theodoridis, Y., and Manolopoulos. C 2 P: Clustering based on

Closest Pairs. In: In Proceedings of the 27th international Conference on Very Large Data Bases ; 2001. p. 331--340.

63 Wang, X., Rostoker, C., and Hamilton, H. J. Density-based spatial clustering in the presence of obstacles and facilitators. In: European Conference on Principles and Practice of Knowledge Discovery in Databases; 2004; Pisa, Italy. p. 446-458.

64 M. Charikar, C. Chekuri, T. Feder, R. Motwani. Incremental Clustering and Dynamic Information Retrieval. In: Proc. 29th Annual ACM Symposium on Theory of Computing,; 1997.

65 Muna Al-Razgan, Carlotta Domeniconi, and Daniel Barbar´a. Clustering Ensembles for Categorical Data. [Internet].

66 Zadeh LA. Soft Computing and Fuzzy Logic. IEEE Software. 1994;11(6):48-56.

67 Sushmita Mitra, Sankar K. Pal, Pabitra Mitra. Data Mining in Soft Computing Framework: A survey. IEEE TRANSACTIONS ON NEURAL NETWORKS. 2002 January.

68 Haykin S. Neural Networks: A Comprehensive Foundation. Pearson Education; 1999.

69 Kohonen T. Self Organizing and Associative Memory. Springer-Verlag; 1988.

70 Holland JH. Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence. University of Michigan Press; 1975.

71 Usama M. Fayyad, Andreas Wierse, Georges G. Grinstein. Information visualization in data mining and knowledge discovery. Academic Press; 2002.

72 Tufte ER. The Visual Display of Quantitative Information. Graphics Press; 1983.

73 Tufte ER. Envisioning Information. Graphics Press; 1990.

74 Chambers JM. Graphical methods for Data Analysis. 1983.

75 A Inselberg ,Bernard Dimsdale. Parallel Coordinates: A Tool for Visualizing Multi-Dimensional Geometry. In: Visualization ‘90; 1990. p. 361-390.

76 Ramana Rao and Stuart K. Card. The Table Lens: Merging Graphical and Symbolic Representations in an Interactive Focus+Context Visualization for Tabular Information. In: Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems; 1994.

77 Lawrence AB. On Chernoff Faces Graphical Representation of Multivariate Data. Academic Press; 1978.

78 Pickett, R. M., Grinstein, G., Levkowitz H. and Smith S. Harnessing Preattentive Perceptual Processes in Visualization; Perceptual Issues in Visualization. 1999. p. 579-587.

79 Keim D. A. , Kriegel H.. Database Exploration Using Multidimensional Visualization. IEEE Computer Graphics and. 1994 40-49.

80 Keim DA. Information Visualization and Visual Data Mining. IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS. 2002;7(1):100-107.

81 Nahum Gershon, Stephen G. Eick. Information Visualization. IEEE Computer Graphics and Applications. 1997 29-31.

82 Dr. Mihael Ankerst, Prof. Daniel A. Keim. Visual Data Mining.

83 Plaisant C. The challenge of information visualization evaluation. In: Proceedings of the working conference on Advanced visual interfaces; 2004. p. 109-116.

84 Chen C. Top 10 unsolved information visualization problems. Computer Graphics and Applications, IEEE. 2005 12-16.

85 Milligan GW. An Examination Of The Effect Of Six Types Of Error Perturbation On Fifteen Clustering Algorithms. PSYCHOMETRIKA. 1980;45(3):325-342.

86 Sokal RR SP. Principles of numeric taxonomy [Internet]. 1963.

87 Aldenderfer M. S, Blashfield R K. Cluster Analysis. Sage Publications; 1984.

88 He Q. A Review of Clustering Algorithms as Applied in IR [Internet]. 1999.

89 Mike Ebbers, John Kettner, Wayne O'Brien, Bill Ogden. Introduction to the New Mainframe z/OS Basics. redbooks; 2011.

90 Ming-Yi Shih, Jar-Wen jheng, Lien-Fu Lai. A two step method for Clustering Mixed Categorical and Numeric Data. Tamkang Journal of Science and Engineering. 2010;13(1):11-19.

91 Huang Z. Extensions to K-Means Algorithm for Clustering Large Data Sets with Categorical values. Data MIning and Knowledge Discovery. 1998;2:283-304.

92 Sokal R. R, F. J. Rohlf. The comparison of dendrograms by objective methods. 1962.

93 Bartke K. 2D, 3D and High-Dimensional Data and information visualization [Internet].

94 Sándor Kromesch, Sándor Juhász. High Dimensional Data Visualization. [Internet].

95 Everitt BS. Graphical Techniques for Multivariate Data. Heinemann; 1978.

96 Daniel A. Keim, Hans-Peter Kriegel, Mihael Ankerst. Recursive Pattern: A Technique for Visualizing Very Large Amounts of Data. In: Proc. Visualization '95; 1995; Atlanda.

97 Daniel AK. Pixel-oriented Visualization Techniques for Exploring Very Large Databases. Journal of Computational and Graphical Statistics. 1996.

98 Chernoff H. The Use of Faces to Represent Points in K-Dimensional Space Graphically. Journal of the American Statistical Association. 1973 361-368.

99 Spence R. Information Visualization. Addison Wesley,ACM press; 2000.

100 L.J.P. van der Maaten, E.O. Postma, H.J. van den Herik. Dimensionality Reduction: A Comparative Review. 2008.

101 Strang G. Introduction to Linear Algebra. Wellesley-Cambridge Press and SIAM ; 2009.

102 Fisher RA. The use of multiple measurements in taxonomic problems. Annals of Eugenics. 1936 179-188.

103 Güting RH. An Introduction to Spatial Database Systems. VLDB:Special Issue on Spatial Database System. 1994;3(4).

104 D. Chaudhuri,A. Samal. A simple method for fitting of bounding rectangle to closed regions. Pattern Recognition. 2007;40:1981-1989.

105 Samet H. The Quadtree and Related Hierarchical Data Structures. ACM Computing Surveys. 1984;16(2):187-260.

106 Guttman A. Proceedings of the 1984 ACM SIGMOD international conference on Management of data - SIGMOD '84.; 1984. p. 47-57.

107 Bentley JL. Multidimensional binary search trees used for associative searching. Communications of the ACM. 1975;18(9):509-517.

108 Mamdani E. H, S. Assilian. An experiment in linguistic synthesis with a fuzzy logic controller. international journal of man machine studies. 1975;7:1-13.

109 Sugeno M. Industrial applications of fuzzy control. Elsevier Science Pub. Co; 1985.

110 Takag T, M. Sugeno. Fuzzy identification of systems and its applications to modeling and control. IEEE Trans. Systems,Man and Cybernetics. 1985;15:116-132.

111 Solomonoff RJ. A Formal Theory Of Inductive Reference. 1964;1(1):1-22.

112 Diamant E. Searching For Image Information Content, Its Discovery, Extraction And. Journal of Electronic Imaging. 2005;14(1).

113 M. Li , P. Vitanyi. An Introduction To Kolomogorov Complexity And Its Applications. NewYork: Springer –Verlang; 1997.

114 http://en.wikipedia.org/wiki/Iris_flower_data_set. [Internet].

115 FARS Analytic Reference Guide 1975 to 2007. National Technical Information Service, Springfield, Virginia 22161.

116 Bentley JL. Multidimensional Binary Search Trees Used for Associative Searching. Communications of ACM. 1975;18(9):509-517.

117 Agrawal R, Gehrke J, Gunopulos D, Raghavan P. Automatic subspace clustering of high dimensional data for data mining applications. In: Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data; 1998; Seattle Washington. p. 94-105.

118 Ester M., Kriegel H.-P, Jörg S, Xu X. A density-based algorithm for discovering clusters in large spatial databases with noise. In: 2nd International Conference on Knowledge Discovery and Data Mining ; 1996. p. 226-231.

119 Shlens J. A Tutorial on Principal Component Analysis [Internet]. 2009.

120 Ientilucci EJ. Using the Singular Value Decomposition [Internet]. 2003.

121 Wikipedia/GA. [Internet].

122 Wikipedia/Dataset. [Internet]. Available from: http://en.wikipedia.org/wiki/Data_set.

123 http://en.wikipedia.org/wiki/Data_set. [Internet]. Available from: http://en.wikipedia.org/wiki/Data_set.

124 K. Bache and M. Lichman. UCI Machine Learning Repository. [Internet]. Available from: http://archive.ics.uci.edu/ml.

125 Qingguang Cui, Matthew O. Ward , Elke A. Rundensteiner. Enhancing Scatterplot Matrices for Data with Ordering or Spatial Attributes. [Internet].