Clustering algorithms for disease classification using mass spectrometry data

Vikram Chandramohan

A thesis submitted

in fulfillment of the requirements for the Degree of

Master of Science (Research)

School of Maths, Physics and Information Technology

James Cook University

May, 2008

Clustering algorithms for disease classification using mass spectrometry data

Declaration

I declare that this thesis is my own work and has not been submitted in any form for another degree or diploma at any university or other institute of tertiary education. Information derived from the published and unpublished work of others has been acknowledged in the text and a list of references is given.

Vikram Chandramohan
December 22, 2008

Clustering algorithms for disease classification using mass spectrometry data

Statement of Access

I, the undersigned, author of this work, understand that James Cook University will make this thesis available for use within the University library and, via the Australian Digital Thesis network, for use elsewhere.

I understand that, as an unpublished work, a thesis has significant protection under the Copyright Act and;

I do not wish to place any further restriction on access to this work.

Vikram Chandramohan
December 22, 2008

Acknowledgments

First, I wish to express my gratitude to my primary supervisor, A/Prof Tuan D. Pham for his supervison and his guidance during my research at James Cook University. I am also thankful for the advice, he gave me inspite his busy schedule and helping me in organizing the thesis.

I would like to take this oppurtunity to thank, A/Prof Bruce Litow, as my secondary supervisor. I would also like to thank Dr. Donggang Yu, Dominick Beck, Miriam Brandl and Peter Philips for their incredible support during my research at James Cook University. My gratitude is, to all lecturers and colleagues at the School of Maths, Physics and IT, James Cook University, Australia.

Many special thanks to my family members. I am indebted to my parents for the sacrifices they have made for me. I would like to thank my sister, who gave this encouragement to do my research in Australia.

Vita

Publications arising from this thesis include:

Chandramohan, V. and Tuan D. Pham (2008), Cancer classification using kernelized fuzzy c-means. In *9th WSEAS International Conference on FUZZY SYSTEMS.* Sofia, Bulgaria.

Chandramohan, V. and Tuan D. Pham (2007), Analysis of mass spectrometry data using kernel based fuzzy c-means. In *CMLS'07 International Symposium.*, Gold coast, Australia(Poster).

Clustering algorithms for disease classification using mass spectrometry data

Abstract

Besides the availability of genomic data, life-science researchers study proteomics in order to gain insight into the functions of cells by learning how proteins are expressed, processed, recycled, and their localization in cells. Proteomics are defined as the study of proteome which refers to the entire set of expressed proteins in a cell. In particular, functional proteomics involves the use of mass spectrometry (MS) to study the regulation, timing, and location of protein expression. It has been recently realized that the use of MS coupled with pattern recognition methodology can offer tremendous potential for the early detection of complex human diseases, and biomarker discovery. However, given the promising integration of several machine-learning methods and MS data in high-throughput proteomics, this biotechnology field still encounters several challenges in order to become a mature platform for clinical diagnostics and protein-based biomarker profiling. Some of the major challenges include noise filtering of MS data, feature extraction, feature reduction of MS datasets and selection of computational methods for MS-based classification. The main objective of this research is to classify diseases using MS data. First, we investigated feature extraction of MS data based on the fundamentals of signal processing such as the theory of linear predictive coding. Then we present an unsupervised kernel based fuzzy c-means (KFCM) approach, which is shown to be more robust to noise than fuzzy c-means (FCM) for mass spectrometry dataset. The KFCM is realized by modifying the original Euclidean distance in FCM by a kernel-induced distance. We evaluated the performance of our classification methods with some popular classification techniques such as support vector machine (SVM), principle component analysis (PCA), linear or quadratic discriminate analysis (LDA/QDA) and random forests.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Cancer research is a major research area in the medical field. Modern proteomics have made enormous progess in the past years, providing tools that have been applied to the study of biological information. Proteomics refers to the entire set of expressed proteins in a cell and it helps to study and understand biological information, which will lead to the discovery of pathways involved in normal processes and in disease pathogenesis [78].

In its current state, Surface Enhanced Laser Desorption/Ionization Time-Of-Flight mass spectrometry (SELDI-TOF MS) is the technology used to acquire the proteomic patterns to be used in the diagnostics setting. In such an approach, human tissue samples are collected from some easily accessible body fluids such as serum, urine, or saliva to produce protein spectra [4, 14]. Every protein spectrum consists of a sequence of peaks, each of which corresponds to a specific protein and is characterized by its mass to charge ratio (m/z) and intensity. The intensity distributions of control and disease are distinct at a specific m/z, this m/z ratio is a useful feature for the classification of healthy and disease. The use of SELDI-TOF MS profiling of serum proteins combined with advanced computation models, to detect protein patterns associated wiht disease, has been reported as a promising field of research to achieve the goal of early cancer detection. Research has shown the potential of this proteomic method to diagnose the difference between control and cancer datasets [45, 58].

Integration of mass spectrometry data in high-throughput proteomics still encounters several challenges such as feature extraction, feature reduction of MS data and selection of computational models for MS-based classification [72, 73]. Biomedical data is noisy and notoriously complex, so machine learning techinques cannot be applied directly to the mass spectrometry dataset [72], so feature selection steps are performed to find a moderate number of proteins (features), that contribute most to correct classification [16]. Much of the effort in this thesis focuses on using unsupervised clustering algorithms to evaluate prostate cancer datasets using the

features obtained from linear predictive coding, which try to detect the patterns that allow the diagnosis of cancer versus non-cancer. Thus, this field represents an active area of current research.

1.1 Proteomics for disease classification

Mass spectrometry (MS) is an experimental method for protein identification, which provides the ability to characterize thousands of proteins present in a complex biological mixtures. MS plays a vital role in proteomics to analyse protein, peptides, oligonucleotides, identification of protein by database searching, sequence confirmation and protein structure prediction. In particular, data produced by mass spectrometers are affected by errors and noise due to sample preparation, sample insertion into the instrument (different operators can lead to different results using the same sample) and the instrument itself. Mass spectrometry based proteomics experiments usually comprise a data generation phase, a data pre-processing phase and a data analysis phase (data mining, pattern extraction or peptide/protein identification). Mass spectrometry produces a huge volume of data, called spectra, that are represented as a very large set of measures $(intensity, m/z)$, representing the abundance (intensity) of biomolecules having certain mass to charge ratio (m/z) values. The capabilities of a mass spectrometer are determined by its ion source, mass analyser and detector. Protein profiling of plasma and serum has been performed primarily with a matrix-assisted laser desorption ionization ion source (MALDI) or its derivatives, and the surface-enhanced laser desorption ionization (SELDI) coupled to time-of-flight (TOF) mass analyser. Data are recorded as plots of intensity versus mass-to-charge ratio (m/z), and referred as mass spectrum. For large samples such as biomolecules, molecular masses can be measured within an accuracy of 0.01% of the total molecular mass of the sample i.e. within 4 Daltons (Da). This is sufficient to allow minor mass changes to be detected, e.g. the substitution of one amino acid for another or a post–translational modification. Though large sequences of m/z data contain a lot of information in an implicit way, manual inspection of experimental data is difficult, so to tackle this problem, computational and soft computing methods are used. Serum proteomic signatures obtained from mass spectrometry are used as a diagnostic classifier of proteomic signatures from high dimensional MS data. Such a proteome has given promising results in the detection of disease in an early stage [33, 43, 57].

Instrumentation

Mass spectrometry can be divided into three fundamental parts, namely the ionisation source to produce ions from the sample, one or more mass analyser, to separate the ions according to their m/z ratios; a detector to register the number of ions emerging from the last analyser; and a computer, to process the data, to produce mass spectrum in a suitable form and to control the instrumentation through feedback. Each mass spectrometer also has an inlet device to introduce the analyte into the ion source, for example a direct insertion probe or liquid chromotography. The separated ions are detected and this signal is sent to a data system where the m/z ratios are stored together with their relative abundance for presentation in the format of m/z spectrum [32].

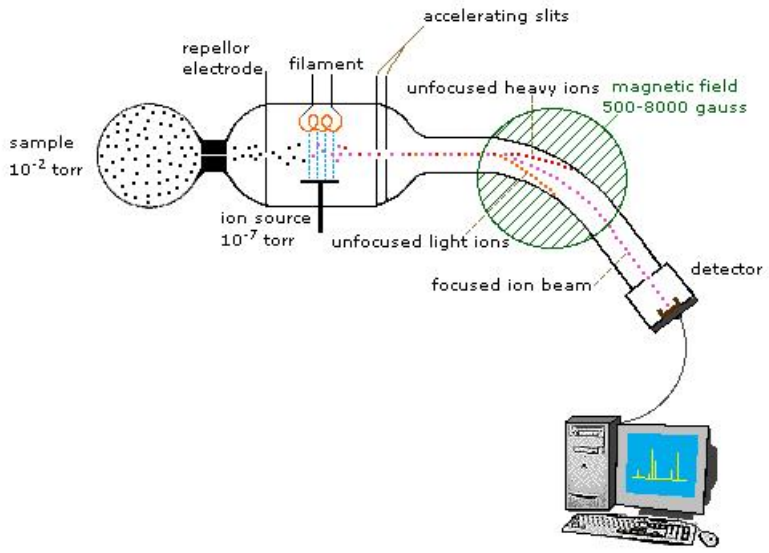


Figure 1.1: *Overview diagram of mass spectrometer*

Methods of ionisation

To analyse a sample by MS, it must be first vaporised and ionised. The ionisation techniques most commonly used for the mass spectrometeric analysis of proteins and peptides are electro spray inonisation (ESI), MALDI and SELDI [45].

Electrospray ionization (ESI)

The generation of ESI was first demonstrated by Dole et al. in 1968, but it was Fenn's group at Yale university that first coupled ESI with MS. The ESI process

transfers ions in solution into gaseous ions at atmospheric pressure, which are sampled into the vacuum system of the mass spectrometer through a series of sampling apertus separating successive vacuum stages. The sample solution flows at low flow rates through a capillary tube to which a high voltage (1-6 kV) is applied. The solution flowing through the capillary experiences an electric field set up between the capillary and a counter electrode and, assuming a positive potential is applied to the capillary, positive ions in solution will accumulate at the surface of the tip, which becomes drawn out, assuming a conical shape known as a "Taylor cone". As the liquid is forced to hold more electric charge, the cone is drawn out into a filament, when the surface tension is exceeded by the applied electrostatic force, produces positively charged droplets (<10 mm in diameter) via a "budding" process. Then the droples fly towards the counter electrode, which is opposite in charge to their own. As they fly towards the electrode, they pass through either a heated capillary $(180 - 270^oC)$ or a curtain of heated nitrogen to allow solvent to evaporate. Depending on the initial size of the droplets, the particles leaving can either be smaller droplets, or discrete solvated surface ions. At atmospheric pressure, collisions with surroundings gases quickly desolvate the solvent-clustered ion, resulting in quasi-molecular ion. Significantly, the ESI process occurs at relatively low temperatures, and so large, thermally liable, polar molecules can be ionised without decomposition. The pre-requiste for gaseous ion production with ESI is that analyte can be ionised in solution. If several ionisable sites are present, mutilply charged ions will be produced. By observing such multiply charged species, the effective mass range of the spectrometer can be extended to hunderds of thousands of daltons [1, 32].

Matrix assisted laser desorption/ionization (MALDI)

MALDI technique was first introduced in 1988 and successfully applied in biochemical analysis of proteins, peptides, glycoproteins and oligonucleotides. The mass accuracy depends on the type and performance of the analyser of the mass spectrometer. The sample to be analysed is co-crystallized with a large excess of matrix material that will strongly absorb the light from a laser. Irradiation of the matrix causes rapid heating and localised sublimation of the matrix crystals. Since the matrix is in large excess and contains a chromophore for the laser light it will absorb esstentially all of the laser radiation. As the matrix expands into the gas phase it takes with it intact analyte molecules, allowing ionisation without fragmentation. Ionisation can occur anytime during this process, but the extact origin of ions produced by the MALDI process is still not fully understood [43]. The most widely used mechanism involves gas phase proton transfer in the expanding matrix plume with photoionised matrix molecules, which is shown in Fig 1.2. The matrix molecules absorb the energy from the laser light and transfer it into excitation energy of the

solid system. The effect is an instantaneous phase transition of small molecular layers of the sample into a gaseous state. It is also a soft ionisation method and so results predominantly in the generation of single charged molecular-related ions regardless of the molecular mass, hence the spectra are relatively easy to interpret. The MALDI process is independent of the absorption properties and size of the compound to be analysed and therefore allows the desorption and ionisation of analytes with very high molecular masses [10, 32, 43].

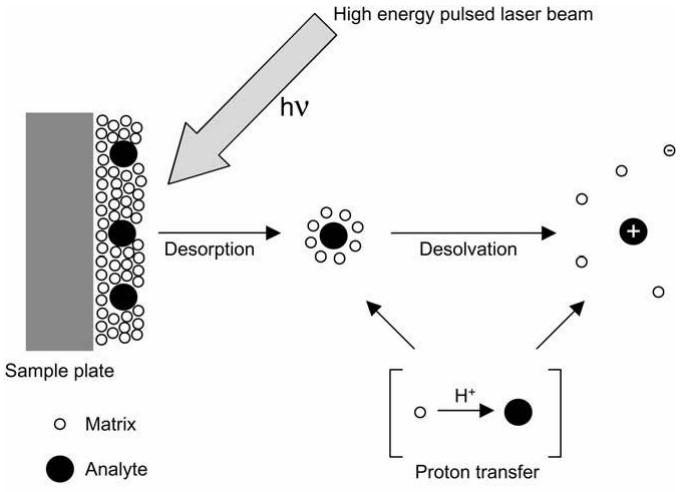


Figure 1.2: *MALDI Process*

Surface-enhanced laser desorption-ionisation (SELDI)

As with genomics, chip technology is beginning to be applied in the proteomics field. As proteins are heterogenous, a simple one-chip for all genes is not currently achievable as no capture molecules capable of binding all possible proteins are available. Therefore, a variety of protein and peptide arrays have been developed to analyse a specific protein or group of proteins. Affinity-based MS techniques represent a further proteomic tool. Ciphergen Biosystems, Inc. have developed SELDI protein chip technology that allows for the non-destructive analysis of both large and small molecules, coupled with automated MS analysis.

SELDI-TOF-MS can be considered as the extension of MALDI-TOF method, but it differs in the construction of the sample targets, the design of the analyser and the software tools used to interpret the acquired data. In the SELDI method, protein solutions are applied to the spots of Protein Chip Arrays, which have been derivatized with planar chromatographic chemistries. These protiens actively interact with the chromatographic array surface, and become sequestered according to their surface interaction potential as well as being separated from salts and other sample contaminats by subsequent on-spot washing with appropiate buffer solutions. Furthermore,

protein interation studies or enzymatic reactions may be carried out directly on-spot under physiological conditions. The chromatographic surfaces provide a very good support for the co-crystallization of matrix and target proteins, resulting in the information of a homogenous layer on the spot, thereby delivering an ideal crystalline surface for the subsequent analysis.
The SELDI platform has been successfully used to quantify relative levels of prostate-specific anitgen (PSA) from serum, and in combination with PSA, can discriminate between benign prostatic hyperlasia and prostate cancer patients. This approach is very useful for detecting markers profiles of disease, however its use is limited in discovery research due to the difficulties in determining the identity of the marker polypeptides. The most widely heralded proteomics study to date is that of Lance Liotta and Emanuel Petricoin III, who used SELDI to anlayse the protein patterns of serum from prostate cancer patients, which shows some promising results in detecting cancer in its early stage [58]. Though a vast amount of data has been produced by SELDI-MS, computer algorithms are vital to screen for potential biomarkers.

Analysis and separation of sample ions

The main function of the mass analyser is to separate, or resolve, the ions formed in the ionisation source of the mass spectrometer according to the mass-to-charge (m/z) ratios. There are number of mass analysers currently available, the better known of which include Time-Of-Flight (TOF) analysers, magnetic sectors, quadrupoles and both Fourier transform and quadrupole ion traps. They are diverse in terms of design and performance, and can be used as a stand-alone analysers or in some cases, put together in tadem to take advantage of their different strengths.

Time-of-Flight (TOF)

In Time-Of-Flight (TOF) instruments, positive ions are produced by periodic bombardment of the sample with brief pulses of electrons, secondary ions, or laser-generated photons. The ions produced by the laser are then accelerated by an electric field pulse and passed into a field-free drift tube. An accelerating potential (ν) will give an ion of charge z an energy of $z\nu$, which can be quated to the kinetic energy of the ion:

$$z\nu = \frac{m\nu^2}{2} \tag{1.1}$$

where $m = mass$, $\nu = velocity$ Ideally, all ions entering the tube will have the same kinetic energies, and their velocities must therefore vary inversely with their

masses, with lighter particles arriving at the detector earlier than the heavier ones. With velocity (ν)=distance (d)/$time(t)$, the equation can therefore be rewritten as $\frac{m}{z} = \frac{2\nu t^2}{d^2}$. The ions therefore drift through a field-free path and are separated in space and time-of-flight. Mass-to-charge ratios are determined by measuring the time that ions take to move through a field-free region between the source and the detector [32].

1.2 Background and Motivation

The proteomics research field is progressing through the development of novel technology and the diagnosis of disease based on mass spectrometry is an emerging field to revolutionize early medical diagnosis. It has recently been realized that the use of MS coupled with pattern recognition methodology can offer tremendous potential for the early detection of complex human diseases, and biomarker discovery. Because of the multi-factorial nature of MS data, it is clear that computational methods are needed to analyse the given datasets which will help in detecting the disease. The combination of data mining techniques with SELDI-TOF- MS must overcome several challenges to become a mature platform. Thus, for early detection of cancer based on pattern analysis, we need more rigorous and systematic approaches. Recent studies confirm that there is no universal pattern recognition and classification model to predict molecular profiles across different datasets and medical domains [72]. Many classification and knowledge discovery problems may require the combination of multiple techniques not only to improve the accuracy and efficiency of the analysis tasks, but also to support evaluation process. This motivates the need for a comparative study on mass spectrometry datasets using different machine learning approaches.

1.3 Aims and Objectives

This research is based on existing theories available in the pattern recognition techniques. The primary aim of the research is to apply machine learning techniques to classify mass spectrometry datasets that comprises a collection of methods for extracting features from prostate cancer datasets and classifying healthy men from disease using clustering techniques. The objectives of this research are:

- To understand, to a certain extent, the context and knowledge of medical experts, in order to develop a knowledge base.

- To design a model that can analyse and classify mass spectrometry dataset. This model includes fuzzy clustering and kernel based fuzzy clustering algorithms.
- To evaluate the performance of our clustering algorithms, by comparing the results with popular classification methods.
- To identify the effectiveness and limitations of several machine learning techniques in analysing a prostate cancer dataset.

1.4 Problem description

Fuzzy clustering is an unsupervised clustering algorithm that has been widely studied and applied in a variety of key areas. FCM attempts to find the most characteristic point in each cluster, which can be considered as the centroid c of the cluster and then, the grade of membership for each object in the clusters. Recent studies in cluster analysis suggest that a user of clustering algorithm should keep certain things in mind: 1) every clustering algorithms will find clusters in a given dataset whether they exist or not; the data should, therefore subjected to tests for clustering tendency before applying a clustering algorithms, followed by a validation of the clusters generated by the algorithm; 2) there is no best clustering algorithms on a given dataset. Further, issues of data collection, data normalization, representation and cluster validity are as important as the choice of clustering stagery. However, the implementation of unsupervised clustering algorithms require a priori selection of cluster centers c, so it is necessary to validate each of the fuzzy c-partitions once they are found. In addition to the number of clusters c, the FCM clustering algorithm and its various extensions require a priori choice of the "degree of fuzziness" parameter m, also called as fuzzy exponent. When FCM is used in unsupervised mode, cluster center c is determined by fuzzy validity measure and it leaves the value of the fuzzy exponent m to be determined, which remains an open problem. Therefore, we experimented our dataset with three traditionanl validity measures to solve the above specficied problems in clustering algorithms. In literature about FCM, m is commonly fixed to 2 for easy computation, but when applied to mass spectrometry datasets, we observed the membership values are similar, FCM failed to extract useful clustering structure. Similar to the work of Dembele et al.(2003), we determined the upper bound value m for the given feature extracted MS dataset, which helped us to decide to choose m lower or equal to 2, to get high membership values for data points related to clusters. According to my knowledge, it was noted that, there was no strong theoretical justification or emprical evidence for these choices.

1.5 Thesis outline

This thesis is mainly concerned with feature extraction and classification methods for analysing mass spectrometry dataset. It is organized as follows:

- Chapter 2: This chapter gives the basis of machine learning perspective in analysing mass spectrometry dataset. The literature is reviewed based on five stages used for data analysis, namely 1)Pre-processing, 2)Feature extraction, 3)Feature selection, 4)Classifiers. These five stages are mutually dependent and the best combination of methods to be used at each stage must be determined empirically.

- Chapter 3: This chapter describes the research methodology, structure and major contributions of this thesis.

- Chapter 4: This chapter details the extraction of features from the mass spectrometry dataset using the principle of linear predictive coding (LPC). As the data size of the protein spectra obtained by SELDI is 15,154 points, it is impractical to use all data as the input features to the classification because (a) some data points may contain noise, which reduces the performance of classification methods; (b) a large number of features increase computational complexity. Therefore, we performed feature extraction to select the most significant points out from the SELDI data as the features for cancer detection. We applied LPC to extract or select the features from the given MS dataset (m/z), though the raw form doesn't convey useful information for the task of classification. Considering MS data as a signal, the features can be extracted and it can be represented as LPC coefficients.

- Chapter 5: This chapter describes a number of unsupervised pattern recognition techniques. In section 5.1, we describe fuzzy clustering algorithms, which can be used for clustering when the number of clusters is known. In section 5.2, we discuss popular kernel based unsupervised fuzzy clustering algorithms, which includes the formulation, brief review about kernel functions and the implementation of kernel trick into fuzzy c-means called kernel fuzzy c-means (KFCM). KFCM is realised by replacing the original Euclidean distance in the fuzzy c-means with a kernel induced distance metric. In section 5.3 and section 5.4, we discuss some of the popular cluster validation techniques. As we know fuzzy clusterings are mainly influenced by cluster centers c and exponent value m, we carried out our experiments with traditional validity indices namely V_{PC}, V_{CE}, V_{XB} to determine the cluster centers and we followed the procedure of Dembele (2003) to determine the exponent value.

- Chapter 6: This chapter gives a broad overview of the application of distortion measures to calculate the dissimilarities between two feature extracted vectors. In addition, cross-validation is performed to evaluate the accuracy of our clustering algorithms.

- Chapter 7: This chapter first introduces the dataset used in our experiments, then shows the outcomes of FCM and KFCM with different parameters initialization. In addition, clustering algorithms is experimented using different distortion measures and the results are compared with popular machine learning techniques.

- Chapter 8: This chapter concludes the whole thesis and summarizes the conclusions obtained in each chapter.

Chapter 2

Literature review

2.1 Introduction

Computational methods are needed in all levels of proteomics analysis [33]. Software packages and tools are being developed to analyse protein patterns, which can help in analysing protein spots on images, matching and editing. Data warehouse technology is used to improve efficiency and accuracy in accessing the databases and to enhance the schema to be flexible and comprehensive. New techniques and new collaborations between computer scientists, biostatisticians and biologists are needed to develop an integrated database for the various sources of data, to develop tools for transforming raw primary data into forms suitable for public dissemination or formal data analysis, to develop user interfaces to store and retrieve, to visualize data from databases and to develop efficient methods to analyse data. Distinguishing correct from incorrect pattern assignments can be regarded as machine learning or supervised learning, a major topic in the machine learning field [33, 35]. Many powerful methods have been developed to identify samples into different groups based on the spectra generated by the mass spectrometer. Due to the huge number of clustering and classification algorithms available, it is somewhat imperative to study the comparative performances of such algorithms [31]. The literature is reviewed on the basis of machine learning prespective in analysing mass spectrometry datasets. There are five stages of data analysis available, namely 1). Pre-processing to reduce the influence of aspects of the data that are not expected to aid in the goal of discrimination between diseases and healthy, 2). Feature extraction aims to reduce the dimensionality of the data. Following feature extraction step it is necessary to perform, 3). Feature selection in which subset of features that best enable discrimination between two groups, 4). Machine models which are designed to distinguish control from diseased samples based on the selected features, will increases the likelihood of successful classification. The five stages are mutually dependent

and the best combination of methods to be used at each stage must be determined empirically [33, 67, 76]. The goal of these efforts is to improve diagnostic methods by either discovering the serological tests or bio-markers, or to improve pathological analysis using tissue proteomics [72].

2.2 Preprocessing

Unfortunately, before standard classification algorithms can be employed, the "curse of dimensionality" needs to be addressed. Due to the sheer amount of information contained within the mass spectra, most standard machine learning techniques cannot be directly applied. Therefore preprocessing has to be performed to reduce the noise in the data such that machine learning can tease out the key information and correctly classify new samples based on a limited set of examples [3, 35]. In mass spectrometry, the noise is the undesired interfering signal caused by sources unrelated to the biochemical nature of the sample being analysed and the signal is the relative abundance of ions originating from the proteins in the sample. Many studies to date have not employed explicit noise reduction schemes other than basic noise reduction methods implemented on commercial mass spectrometers. However, some investigators have explored methods for reducing noise, particularly the baseline and high frequency noise [30, 39]. A variety of approaches have been explored to estimate the baseline from mass spectra namely heuristic or model-based. Heuristic approaches form non-parametric estimates of the baseline from a set of mass spectra and is one of the most commonly used methods to estimate and eliminate the baseline. Model-based approaches build a mathematical model of the baseline based on the physics of the mass spectrometer and estimate the parameters of the model. A local average or minimum intensity within a moving window has been used as a local estimator of the baseline and the overall baseline is estimated by sliding the window over the mass spectrum. Piecewise linear regression has been applied to the regions with a monotonically decreasing baseline. For methods in which a sliding window or piecewise linear regression are employed for baseline elimination, the window size is a critical factor for determining the overall performance. If the window size is too large, these methods may oversimplify the curvature of the baseline with a long straight line. If the window size is too small, they may produce an overly complex estimate of the baseline, which is very sensitive to high frequency noise [27, 31, 63]. All of the methods have made considerable contributions to high frequency noise reduction in mass spectra. However, since no study has extensively compared the methods introduced above on the same data set, it is difficult to conclude if one method is better than the others. Moreover, the overall performance of those high frequency noise reduction methods is highly dependent on the choice of the filter

parameters (e.g., the size of the sliding window or the kernel weights) and the true effectiveness of those methods is difficult to measure due to the lack of knowledge on the statistical characteristics of the signal and noise in mass spectra [1, 31].

2.3 Feature extraction

In decision support systems utilizing mass spectra, feature extraction can be defined as a process of extracting summary information reflecting the pathological status of a sample from pre-processed mass spectra. These techniques are crucial for learning high-accuracy classifiers and realizing the full potential of mass spectrometry for disease diagnosis. In this section we discussed in detail about some of the existing feature extraction techniques and the details are as follows.

2.3.1 Wavelets/Principle Component Analysis (PCA)

Wavelets are mathematical functions that divide data into different frequency components, and then study each component with a resolution matched to its scale. This technique achieved a broad and successful application to pattern recognition in the last decades. It is also an efficient way of reducing or comprising the similar data, and localizing a signal in both time and frequency. In recent years, there has been a growing interest in the application of wavelet methods to biomolecular related signals. In [21], the researchers applied discrete wavelet transform (DWT) to extract useful features from mass spectrometry dataset. The discrete wavelet transform is like the Fourier transform, and can be used to obtain meaningful features by mapping the spectrum into another space [64]. In Fourier analysis, the analyzing functions are the set of sine function, whereas for DWT, wavelets are the analyzing functions. The DWT is defined as

$$x(t) = \sum_{j=1}^{l} \sum_{k=0}^{2^l} c_{j,k} \psi_{j,k} \tag{2.1}$$

where $\psi_{0,0}$ is the father wavelet, which helps to calculate $\psi_{j,k}$, $x(t)$ is the spectrum, l is the decomposition level for the DWT and $c_{j,k}$ is the wavelet coefficient calculated between the inner product $x(t)$ and $\psi_{j,k}$.
Considering mass spectrometry data MS of length N, the DWT consists of $Log_2 N$ levels at most. The first level produces two sets of coefficients: approximation coefficients and detail coefficients. These vectors are obtained by convolving MS with the low-pass filter for approximation, and with the high-pass filter for detail.

The next level splits the approximation coefficients in two parts using the same scheme and so on. Though there are many types of analysis functions (wavelet) available, but there is no clear idea about the selection of the wavelets, therefore the researchers used linear combination of wavelets and referred as super-wavelets. The wavelet transform (WT) has been also employed not only to reduce noise but also to extract features from mass spectra in a similar fashion as PCA is used. The WT also compresses data by projecting the original data onto pre-specified orthogonal directions (wavelets) [49]. The coefficient of each wavelet becomes a feature in this case, thus the wavelets representing high frequency components are usually ignored, and noise reduction is simultaneously accomplished with feature extraction. In few studies, features extracted by projecting the signals from the original space onto another are done by principle component analysis (PCA), for more details refer to section 2.5. PCA identifies the orthogonal directions in which data vary maximally using the eigenvalue/eigenvector decomposition of the covariance matrix. Then the original signals are projected onto those directions, the number of which is usually smaller than the original dimension. The projections are called principle components and are often used as features [16]. Both the approaches are very sensitive to the choice of components (i.e., principle eigenvectors in PCA or wavelets in the WT); therefore, it is important to determine criteria for selecting eigenvectors or wavelets prior to feature extraction.

2.3.2 Genetic algorithms (GA)

Genetic algorithms (GA) are a family of computational models inspired by evolution and can be used to solve problems efficiently for which there are many possible solutions. GA's are often viewed as function optimizers, although the range of problems to which genetic algorithms have been applied is quite broad. In a broader usage of the term, a GA is any population based model that uses selection and recombination operators to generate new sample points in a search space. Many genetic algorithm models have been introduced by researchers largely working from an experimental perspective. Many of these researchers are application oriented and are typically interested in genetic algorithms as optimization tools. In GA's, the initial step is to generate random data, consisting of predefined of individuals (rows) and variables (columns). Each individual represents a subset of the original variables with the larger superset of data under analysis. The next step in the GA is analogous to the process of Darwinian evolution whereby, through the process of crossover, mutation and survival of the fittest, individuals are selected for the next generation until a particular stopping criterion has been reached. The GA algorithm uses a fitness function to assess the robustness of the model proposed by

each individual [38, 48].
In [58], GA is applied effectively to extract features from the given MS data. The GA starts randomly selecting many small subset of key values along the spectrum x axis using an iterative algorithm searching process. The fitness was conducted to plot the patterns in N-dimensional space, where N is the number of m/z values in the test set. The pattern formed by the iterative amplitude of the spectrum data for this set of chosen values is rated for its ability to distinguish the two preliminary population. Then they reshuffled the highest rated sets to form new subset candidates and the resultant y-axis-defined amplitudes are rated iteratively until the set that fully discriminates the preliminary set emerges. This subset which was selected considered as important because the pattern of amplitudes at these m/z values completely segregated the serum from patients with prostate cancer from the unaffected populations.

2.3.3 Peak detection techniques

In feature extraction, a variety of peak detection and alignment algorithms are being developed and tested. It is complicated to identify the peaks in mass spectrometry dataset due to the mass error rate. The main goal of feature extraction is to identify sets of m/z values which comprises peaks that are higher than the noise level of mass spectrum. In many studies, commercial software has been used to find as many peaks as possible and then applied threshold levels to select the peaks far higher than the noise level [81]. In Ciphergen biosystems, they developed a software to detect the peak from the noisy background. The working principle of this software is as follows; first it selects the peaks with a high signal to noise ratio with in individual spectra and then it search for the moderate peaks [72]. In [20], the researchers explored alternative peak detetection algorithms for more rigorous peak finding. Most of the peak detection algorithms in the literature find local maximum intensities from the given mass spectrometry dataset and choose the local maxima higher than a thershold of the noise level as peaks. But the researchers used a simple algorithm to register all the m/z values with local maximum intensity, and then used both absolute threshold and relative threshold, exceeding user specified threshold. In addition to the peak detection, they used time wrapping method to align the peaks obtained from each sample. Finally, they obtained reasonable amount of peaks to get good classification rate. They reported even other spectra alignment algorithms are also good candidate for the task. After peak detection and peak alignment, one must define the metrics of a peak group that will serve as features. Feature metrics related to peak heights have been used in most studies. Instead of retaining the peak height as continuous feature data, binary and discretized feature values

have also been investigated as a way to alleviate the variability of feature values across samples that can deteriorate the generalization of the classifier. In feature extraction, a variety of peak detection and alignment algorithms are being developed and tested. It is necessary to take into consideration about the resolution and noise of mass spectrometry, because it can be easily affected by the noise. It is possible to develop better diagnostic systems if we could have the prior knowledge of mass spectrometry and the proteins present in blood for the feature extraction process.

2.4 Feature selection

Feature selection is defined as a series of actions to choose a subset of features that are relevant to correct classification based on specified evaluation and selection criteria. Feature selection methods are often categorized as filters, wrappers, or embedded methods. These techniques are now discussed in detail as follows:

2.4.1 Filter method

Filter methods attempt to select the features based on auxiliary criteria, such as feature correlation, to remove redundant features. A filter method evaluates and ranks individual features based on a selection criteria (e.g., t statistic). Then, a subset of features for classification is determined based on individual feature ranks [20, 47]. It is the most commonly used feature selection method for complex disease classification. A variety of statistical tests such as student-t test (T-test), the Kolmogorov-Smirnov test (KS-test) and the P-test, have been investigated to define selection criteria for the relevancy of individual features. The above specified tests determine the feature values of samples belonging to class 1 to feature values of sample belonging to class 2. The key difference between these tests is the way it makes the assumption in selecting the features. In [45], the researchers used the univariate statistical techniques to rank the individual features, instead of using wrapper-based approaches as a classifier to invoke the feature selection. They reported that T-test assumes that both distributions have identical variance and makes no assumptions as to whether the two distributions are discrete or continuous. In T-test, the null hypothesis is $\mu_1 = \mu_2$, indicating that the mean of the feature values for class 1 is the same as the mean of the feature values for class 2. In the case of KS-test, the null hypothesis is represented as $cdf(1) = cdf(2)$, meaning that feature values from both classes have an identical cummulative distribution. Thus, the features are ranked based on the statistic significance score. In addition they performed simple feature ranking criteria called as P-test, which is the simplified version of T-test and can be defined as

$$P - test = \frac{\|\mu_1 - \mu_2\|}{\sigma_1 + \sigma_2} \tag{2.2}$$

where σ_i is the standard deviation. The above specified P-test basically ignores the sample size and rank features solely based on their mean and standard deviation. The researchers even used greedy forward selection using internal leave-one-out cross-validation (LOOCV), which can automatically detect the feature subsets. In [50], the researchers studied the efficacy of relevancy measures on the basis of information theory and signal processing as filters. For the given MS data, they computed the mean and standard deviation of the feature i across the positive and negative examples. They defined the signal-to-noise, known as MIT correlation as:

$$MIT(i) = \frac{\left|\mu_i^+ - \mu_i^-\right|}{\sigma_i^+ + \sigma_{\bar{i}}} \tag{2.3}$$

When making selection they simply take those features with highest scores as the most discriminatory features. The wavelet transform, which we discussed early in this chapter, can also be used as a filter method for feature selection techniques. The features are considered to be relevant when the features receives high scores from multiple methods. This approach enables one to obtain features from different perspectives and to make a more reliable decision regarding the selected subset of features. Both the benefits and drawbacks of these statistical tests stem from the assumption that features are independent. For more technical details of these and other statistical tests can be found in [29].

2.4.2 Wrapper or embedded methods

Wrappers assess the relevancy of a subset of features based on evaluation metrics of a classifier trained using that subset of features. A search algorithm is used to explore the space of feature subsets and identify a high-performing subset of features. There are two types of search engines available, namely heuristic/greedy methods. Sequential forward selection (SFS) and sequential backward selection (SBS) are the most widely used greedy algorithms in search of features from MS data, which starts with empty features and add up single feature to increase the performance. The SFS technique is easily applicable to the MS data, where as SBS like full search over all the subsets, and is computational intractable [16]. In [50], the researchers implemented several heuristic approaches to track SBS algorithms. To do so they used

KS-test to re-order and to rank all the features. Thus the SBS starts with testing the features deemed most likely to be irrelevant. The second heuristic method added to SBS helps to record the stopping position of the last iteration. Hence this makes the SBS start at the previous stopping position rather than starting from the beginning. Wrappers are different from filters in that classifier evaluation metrics are used rather than selection criteria for individual features and wrappers assess features in groups rather than individually. Filters employ selection criteria such as statistical tests to evaluate individual features, while wrappers use evaluation metrics of classifiers to estimate the discriminating power of a candidate subset of features. The wrapper approach typically has better performance than the filter approach. The combination of genetic algorithm with wrapper method is popular in this field. Several kinds of classifiers have been combined with genetic algorithms, including self-organizing maps (SOM), support vector machines (SVM) and simple distance based classifiers (e.g., Mahalanobis distance). Embedded methods implicitly perform feature selection as a part of the classifier training process. Feature selection can help to reduce running time and avoid overtraining if it succeeds in finding a subset of independent and discriminating features. Unfortunately, there is no guarantee that the feature selection process will improve the classification performance. Moreover, features selected are relevant for classification still need to be biologically validated in future studies [66, 75].

2.4.3 Nearest shrunken centroid

This is the special purpose selection algorithm developed by Tibshirani et al. This algorithm tries to shrink the class prototypes μ_{C_j} towards the overall mean:

$$\mu = \frac{1}{m}\sum_{i=1}^{m} x_i \tag{2.4}$$

Their idea is to shrink the class centroid towards the overall centroid. Therefore, they normalized with-in class deviation for each data and is defined as

$$d_j = \frac{\mu_{c_j} - \mu}{m_j(s)} \tag{2.5}$$

where $m_j = \sqrt{\frac{1}{|C_j|} - \frac{1}{m}}$, s is a vector of pooled within class variance for each feature and division is done component wise. We can view the class centroid as:

$$\mu_{c_j} = \mu + m_j(s.d_j) \tag{2.6}$$

By decreasing d_j, we can move the class centroid towards the overall centroid. To decrease d_j soft thresholding is used to produce $\acute{d_j}$ with:

$$\acute{d_j} = sign(d_j(i))(|d_j(i)| - \delta) \tag{2.7}$$

where $d_j(i)$ is the i^{th} component of the vector d_j. Then the centroid is calculated by replacing d_j with $\acute{d_j}$ in equation (2.6). In [45], the researchers effectively applied the above specified shrunken centroids method for mass spectrometry datasets, using different values for δ. They reported that this method doesn't perform well with prostate cancer dataset, when compared to rest of the feature selection techniques.

2.5 Classifiers

Machine learning is a branch of artificial intelligence that is concerned with design and application of algorithms that enable computers to learn from experience. In recent years, several unsupervised (clustering) and supervised (classification) techniques have been used for identifying samples into different classes based on the spectra generated from the mass spectrometer [72]. In this section, we will provide details about current state of research using classification methods within the development of clinical decision supports systems utilizing mass spectrometry of blood samples.

2.5.1 Support vector machine

The support vector machine (SVM) is a machine learning technique that produces non-linear classification, applied successfully to diverse scientific and engineering problems, including the biomedical sciences. It is applied both for classification and dimensionality reduction of mass spectrometry (m/z) datasets. SVMs are a type of kernel learning method, which project data from the current vector space to another vector space where linear learning programming is applicable. The function that projects the data onto the new space, which usually has a higher dimension than the original, are called the kernel functions. Improper kernel functions may worsen the classification accuracy, so care must be taken for choosing a kernel function when using SVM. But there is no proper guidelines for choosing the best kernels

for a given data set. The most popular kernel functions are the polynomial, radial basis function and sigmoid kernels [13, 23].
After data projection into linear space, SVMs guarantee the maximal margin between normal and disease samples through the optimization of the decision boundary such that overtraining can easily be avoided. A penalty is given to the objective function, to optimize the margin size and misclassification rate. A small penalty maximizies the margin size but increases the classification rate, where as large one decreases the margin size, but minimizes the missclassification rate. SVM can be utilized without any data projection if the data are linearly separable in the current vector space. This method is usually called linear-SVMs that allows for the maximum-margin separation based on few data samples closet to the decision boundary, which are called support vectors, SVMs implicity reflect the contribution of each features to successful classification and reduce the effect of irrelevant features by performing dot product between the gradient and each sample [5, 24].

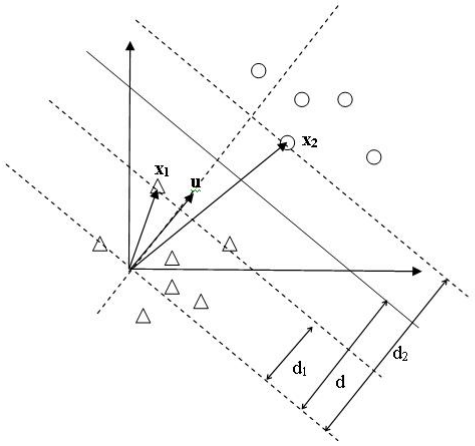


Figure 2.1: *Diagramatic representation of SVM*

The linear discriminant function can be defined as $f(x) = w^t x + w_0$ and can be seen as defining a hyper–plane that maps from the space of data D^n to a space of classes C^m, where in most cases $m << n$. When using SVM as a classification method, feature selection has been performed as an embedded part of the training process and it is better to select multiple classifiers to avoid complexity of mass spectra patterns between normal and disease. In [20], the reseachers employed *link-test* for finding the cancer biomarkers from SELDI mass spectromety. They used 16 unique biomarkers to train SVM with five-fold-cross-validation on prostate cancer samples and obtained the average classification accuracy of $85.3 \pm 1.9\%$. This study states that mass spectrometry intensities are not reliable measurement of protein concentration, so the models for extracting biomarkers from mass sepctrometry are not fully quantitative. To avoid this, they cross-validated mass spectrometry dataset

with micro array to find gene (protein) biomarkers, which will be helpful in pulling out confirmed mass spectra markers. In [50], MIT correlation method was used as the feature selection technique to extract the features from the mass spectrometry dataset and they classified healthy men from those infected using support vector machine (SVM). In this study, they performed ten-fold cross-validation to train SVM with different kernel functions, which splits the dataset into ten subsets. They utilized nine subsets to train the model and one subest to test the model. This approach indicates that SVM with polynomial kernel worked well with prostate cancer data, when compared to linear and radial kernel functions, achieving the selectivity of 89.0% and the sensitivity of 79.0%, for an overall classification accuracy of 81.0%. When using linear kernel with SVM, they obtained the selectivity of 86.0% and the sensitivity of 76.0%. In general, the researchers reported that SVM performed well with prostate cancer dataset, in terms of sensitivity, selectivity, and in accuracy. The weakness of the SVM is that it only estimates the category of the classification, while the assignment probability $p(x)$ may be of interest itself, where $p(x) = P(y = 1/X = x)$ is the posterior probability of a sample being in class 1. Another problem with the SVM is that it is not trivial to select the optimal parameters for the kernel and difficult to understand the structure of algorithm [51].

2.5.2 Self-organizing maps (SOM)

Self-organizing mapping are a particular type of neural network or pattern recognition method known as unsupervised learning, which provide a very convenient 2-dimensional visual representation of multi-dimensional data. It was widely applied in the field of speech analysis, robotics, industrial and medical diagnostics. Every neuron i of the map is associated with an n-dimensional reference vector, where n denotes the dimension of the input vectors. The reference vectors together form a codebook. The neurons of the map are connected to adjacent neurons by a neighbourhood relation, which dictates the topology, or the structure of the map. The most common topologies in use are rectangular and hexagonal. Adjacent neurons belong to the neighbourhood N_i of the neuron i. In the basic SOM algorithm, the topology and the number of neurons remain fixed from the beginning [41, 42]. During the training phase, one MS data say X is randomly drawn from the input data set and its similarity (distance) to the codebook vectors is computed by using Euclidean distance measure. Every node is examined to calculate which one's weights are most like the input vector. The winning node is commonly known as the Best Matching Unit (BMU). After the BMU has been found, the codebook vectors are updated. The BMU itself, as well as its topological neighbours are moved closer to the input vector in the input space i.e. the input vector attracts them.

The magnitude of the attraction is governed by the learning rate. As the learning proceeds and new input vectors are given to the map, the learning rate gradually decreases to zero according to the specified learning rate function type. Along with the learning rate, the neighbourhood radius decreases as well. The training phase is repeated until there is no desirable changes. The number of training steps must be fixed prior to training the SOM because the rate of convergence in the neighbourhood function and the learning rate is calculated accordingly [42]. In [58], the researchers used genetic algorithms for feature extraction and self organizing map to differentiate healthy men from those infected with prostate cancer. This study used SELDI-TOF MS to acquire the mass spectra which corresponds to our PC-H4. This approach achieved a selectivity of 95% and a sensitivity of 71%, though cross-validation was carried out, the results were not presented.

2.5.3 Linear or quadratic discriminate analysis

Statistical discriminant analysis is frequently and widely applicable tool in biology and some related research areas. In practice, standard linear and quadratic methods are often applied which assume equal costs of misclassification. The aim of the discriminant analysis is to assign a unit to one of several groups on the basis of a number of feature variables. The most widely used methods are parametric analysis and Linear discriminate analysis (LDA) [16, 29]. The LDA is popular because of its robustness against deviations from the assumption of multivariate normality of the feature variables. Let us consider $X = (X_1, ..., X_p)$ which denotes the p-dimensional random vector of feature variables which are used for the allocation of a unit to one of g (≥ 2) groups $G_1, ..., G_2$. It is assumed that the vector of feature variables X is multivariate normally distributed in group G_i, $i = 1, ..., g$ with mean vector mu_i and common covariance matrix $\sum$ in case of LDA or group specific covariance matrix $\sum_i$ for QDA. The $f_i(x)$ probability density function of X for group G_i, $i = 1, .., g$. The $f_i(x)$ is defined as

$$LDA : f_i(x) = (2\pi)^{-p/2} \left| \sum \right|^{-1/2} \tag{2.8}$$

$$QDA : f_i(x) = (2\pi)^{-p/2} \left| \sum_i \right|^{-1/2} \tag{2.9}$$

The posterior probability π_i for group G_i, $i = 1, .., g$, $\pi_i(x) = \frac{\tau_i f_i(x)}{\sum_j^g \tau_j f_j(x)}$

For the practical application of the allocation rule the probability density functions $f_i(x)$ and the prior probabilities τ_i have to be estimated. For the standard LDA and QDA the estimated group specific density for group is given by simply plugging in the sample mean vector and the sample covariance matrix s or s_i into the formula of the multivariate normal density function. As we know, mass spectra m/z has a relatively higher dimension than the intrinsic dimensionality of the training set, this method cannot be applied directly to m/z data, though the dimensionality is larger than the set intrinsic dimensionality. That is, in order to guarantee a nondegenerate solution for LDA, the dimensionality of the data must be reduced to at most $n - k$, where n is the number of samples and k is the number of classes. In [45], the researchers used PCA for dimensionality reduction and LDA for classification. They used three-fold-cross-validation procedure, to train/test LDA and obtained a selectivity of 71.0% and sensitivity of 62.3% for an overall BACC of 69.2%. In [21], the researchers again analysed SELDI TOF MS data using LDA as a classifier. In their study, they reported LDA model gave the better classification rate of 89.47% for control, 94.73% for benign, 90.47% for cancer, when compared to Treeboost and Random Forests. The primary weaknesses of LDA/QDA are they are not stable for large number of datasets. It is unable to handle classes that curve around another class, or clusters of points that are contained entirely within the outer radius of another cloud of points [45, 77].

2.5.4 Centroid classification methods

In general, there are two types of pattern recognition techniques: supervised methods and unsupervised methods. In supervised learning we give a set of training samples in different classes:

$$\begin{aligned}
&\text{Samples in class 1: } x_1^{(1)}, x_2^{(1)}, ..., x_{n_1}^{(1)} \\
&\text{Samples in class 2: } x_1^{(2)}, x_2^{(2)}, ..., x_{n_2}^{(2)} \\
&\qquad\qquad ..... \\
&\text{Samples in class c: } x_1^{(c)}, x_2^{(c)}, ..., x_{n_c}^{(c)}
\end{aligned}$$

where $x_k^{(i)}$ represents sample k in class i. For these training data, we need to find a mapping function $\Phi(x_k^{(i)})$, or to build a classifier, which can be a set of fuzzy rules, a neural network, a decision tree or simply mathematical equations. Once the mapping function is determined, it can be used to classify an unseen sample x [11].

Unsupervised classification refers to situations where the objective is to construct decision boundaries based on unlabeled training data $(x_1, x_2, ..., x_n)$ and is also known as data clustering, which is a generic label for a variety of procedures

designed to find natural groupings, or clusters, in multidimensional data similarities among the patterns [37]. Clustering can be considered as the important unsupervised learning problem, it deals with finding a structure in a collection of unlabeled data. A loose definition of clustering is process of organizing objects into groups whose members are similar in some way, therefore a collection of objects which are "similar" between them and are "dissimilar" to the objects belonging to other clusters, which is shown in Fig: 2.2. The goal of clustering is to determine the intrinsic

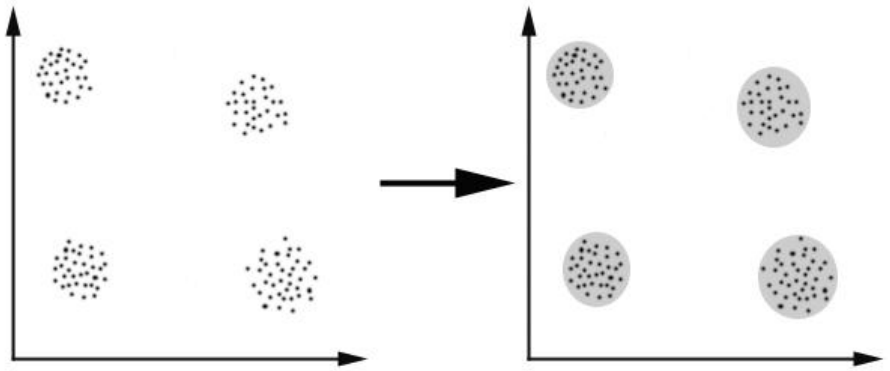


Figure 2.2: *Graphical representation of clustering algorithm*

grouping in a set of unlabeled data. The speed, reliability and consistency with which a clustering algorithm can organize large amounts of data constitute overwhelming reasons to use it in applications such as data mining, image segmentation, signal compression and coding, and machine learning [36].

A fast and simple clustering algorithm for classifying mass spectrometry in literature is the centroid method. This algorithm assumes that the target classes correspond to individual clusters and uses the cluster means to determine the class of a new sample point. A prototype pattern for class C_j is defined as the arithmetic mean

$$\boldsymbol{\mu} = \frac{1}{|C_j|} \sum_{x_i \in C_j} x_i \tag{2.10}$$

where x_i are the training m/z samples labeled as class C_j. It can work well with many features and its run time complexity is proportional to the number of features and the complexity of the distance or similarity metric used. During training, two prototypes are computed and the cost of computing each prototype is $O(mN)$, where N is the number of features extracted from the mass spectrometry dataset and m

is the number of training samples which belong to a given class. Though m only varies between datasets and not during the feature selection process, so we can able to conclude that the centroid classifier has $O(N)$ cost in the training phase [29, 45]. In [45], the researchers used a special purpose feature selection algorithm called nearest centroid algorithm to extract the features from the PC-H4 dataset and applied centroid classification method to discriminate between healthy and cancer. In their experiments they used 20 different values for δ and achieved a selectivity of 73.6% and sensitivity of 83.3% for an overall BACC of 79.1%.

2.5.5 Boosting and Random forests (RF)

Boosting is a machine learning meta-algorithm used for improving the accuracy of any machine learning algorithm. It is a procedure that combines many weak classifiers to achieve a final powerful classifier. Most boosting algorithms proceed in a series of rounds in which a new simple rule is trained according to the labeled training examples. After each round, the training examples are updated to increase the weight of those examples that were improperly classified in the current round. A general boosting framework says neither how distributions and weights been updated nor how the weak rules are to be combined [68]. The input to AdaBoost is a set of m training examples (x_i, y_i), $1 \leq i \leq m$ where x_i is a feature vector drawn from some domain X and y_i is drawn from a label set Y. For T rounds, a new simple rule, or "weak learner", is trained using examples drawn from the training set such that example i is given weight $D_t(i)$ on round t. Starting from the uniform distribution (i.e. $D_1(i) = 1/m, \forall_i$), each round selects a new weak rule $h_t(x_i)$ that minimizes the error: $\in_t = \sum_{i:y_i \neq h_t(x_i)} D_t(i)$. A weight ∂_t is calculated: $\partial_t = \frac{1}{2} ln \frac{1-\in_t}{\in_t}$ Next, the distribution D_t is updated according to the rule

$$D_{t+1}(i) = D_t(i) exp(-\partial_t y_i h_t x(i))/z_t \tag{2.11}$$

where z_t is chosen such that $\sum_i D_{t+1}(i) = 1$.
The equation $H(x) = sign(\sum_t^T = 1 \partial_t h_t(x))$ is the final strong classification rule. Typically one may build hundreds or thousands of classifiers by this way. A final score is then assigned to any input x, defined to be a linear (weighted) combination of the classifiers. A high score indicates that the sample is most likely correctly assigned and the low score indicating that it is most likely an incorrect hit. By choosing a particular value of the score as a threshold, one can select a desired selectivity or a desired ratio of correct to incorrect assignments. In [45], in addition to SFS and SBS the researchers used a boosting method to increase the classification

performance, which attained equal or balanced accuracy (BACC) than any other algorithms tested. To determine the merit of the feature selection approach, first they used a standard boosting algorithm, followed by extended version of boosting algorithm called boosted feature extraction (boosted FE), which is similar in performance to SFS. This approach achieved a selectivity of 100% and sensitivity of 81.2% for an overall BACC of 96.0% and this is the highest reported accuracy of this dataset. Therefore, the researcher came to the conclusion that boostedFE fullfills both roles as a feature selection and classification algorithm.

In a similar way to boosting, the random forests are also an ensemble method that combines many decision trees, defined by Breiman [8]. Decision trees are presented in a binary tree structure by repeatedly splitting data subsets into two descendant subsets. Each terminal subset is assigned a class label and the resulting partition of the dataset corresponds to the classifier. A random forests contains many decision trees and outputs the class that is the mode of the classes output by individual trees. The RF algorithms combines bagging idea to construct a collection of decision trees with controlled variations [8]. The RF enjoys several nice features: like boosting, it is robust with respect to input variable noise and over fitting, it can simultaneously estimate the importance of variables in determining the classification and it can efficiently handle high dimension data. In [21], the researchers effectively applied random forests as a classifier and obtained correct classification rate (CCR) of 100% for control, 68.7% for cancer, 98.24% for benign.

2.5.6 Principle component analysis

Principle component analysis(PCA) is a useful statistical technique that has found application in fields such as face recognition and image compression, and is a common technique for finding patterns in data of high dimension. PCA aims at reducing the dimensionality while determining orthogonal axes of maximal variance from the given m/z data. For PCA to work properly, the mean has to be subtracted from each dimension. The mean subtracted is the average across each dimension. So all the x values have $\overline{x}$ (the mean of the x values of all the data points) subtracted, and all the y values have $\overline{y}$ subtracted from them. This produces a data set whose mean is zero. Therefore, the new dimensionality-reduced dataset can be derived by projecting the original dataset onto these principle components. The drawbacks of PCA is the computational complexity, which is known to be $O(d^2n)+O(d^3)$, where d is data dimensionality and n is the number of cases. PCA is computationally costly because it performs the eigen decomposition of the covariance matrix. In [45], the researchers used nearest centroid method for feature extraction and used stratified three-fold-crossvalidation to train PCA. This approach obtained the selectivity of

51.0% and sensitivity of 54.0% for an overall BACC of 53.0%. The researhers formally states that PCA had poor performance with PC-H4 dataset. Although PCA may be carried out more efficiently by using SVD decomposition and by omitting zero eigen values in calculation, the computational overhead is still too high for high-dimensional data sets like protein MS data.

2.6 Conclusion

This review has provided only a condensed snapshot of applications of machine learning and the development of clinical decision support systems for disease screening from proteomic patterns obtained by mass spectrometry of blood samples. The prior studies are presented in an explicit machine learning framework consisting of five stages: pre-processing, feature extraction, feature selection, classifier training and evaluation. Current techniques have already yielded putative molecular targets, uncovered signal pathways and advanced early disease detection. The co-evolution of genomics and proteomics as complementary approaches to complicated disease will allow us to move closer to the goals of early detection, improved prevention, and tumour-specific approach to the treatment of individual patient.

Chapter 3

Reflection on the Research Method

This research uses a combination of mathematics, statistics, logic and bio-medical science. The research objective is to create an effective technique that would classify healthy men from those with cancer. This research combines knowledge of the different feature extraction and classification methods. The following sections discuss the appropriateness of the research design for this research.

3.1 Research method construction

This research depends on an understanding of the expert knowledge contained in the literature. This will be gathered during the requirements collection phase of the system development. Current concepts of machine learning techniques will allow the application of suitable concept combining knowledge to analysis a set of selected mass spectrometry datasets. An in-depth review of the literature on machine learning techniques and bio-medical science will help to generate knowledge for this multidisciplinary research. The existing knowledge from literature will be fed into the whole research process to achieve a better integration of pattern recognition methods with proteomics.

This research addresses the combination of signal processing techniques with unsupervised clustering algorithms. The underlying theory behind this research is the implementation of kernel trick into fuzzy clustering algorithms to classify mass spectrometry datasets. The research will implement those concepts by constructing a prototype that will also allow researchers to better understand the underlying theory.

However, before we start discussing about the classification methods, it is important

to discover what features (patterns) we most need to extract. One mass spectrometry data may contain many important features, however, the nature of the features is complicated. Due to the time constraints of this research, we concentrated in extracting features from prostate cancer datasets (PC-H4) to get better classification. Furthermore, to apply these techniques to a set of dataset it is also essential to have some understanding of the human and expert knowledge. The success of the research lies on a thorough understanding of the datasets followed by application of unsupervised clustering algorithms in a computer system.

3.2 Data collection procedure

An in-depth literature review will be the first source for data collection. The different source of literature include-

- Journals
- Books
- Reliable online resources

This literature review will be the basis for creating the information about the datasets and in gaining knowledge about the current state of research in the field of proteomic pattern recognition.

3.3 Data analysis procedure

The analysis phase of the data is represented as the prototype testing phase. A set of datasets have been selected for the whole analysis procedure to maintain the consistency of the findings. During the data analysis phase, establishing the relationship between the data is the primary task. Reading the literature and observing the prototype of the framework will allow me to write notes and help to get a better understanding of the datasets and machine learning methods. On the otherhand, it is helpful to write memos about related factors to develop organisational categories. Developing organisational categories will be used to analyse the data. Finally, the data will be theoretically categorised to formally answer the research questions.

3.4 My contribution to the research community

The contributions to this thesis includes two parts: (1) Linear predictive coding (LPC) for feature extraction and (2) kernelized fuzzy c-means (KFCM) for classification.

- LPC for feature extraction : The mass spectrometry datasets downloaded from the public database do not convey valuable information to classify healthy men from cancer. Therefore, its important to extract the features to get better classification, which is carried out using the principle of linear predictive coding (LPC). LPC-based feature extraction is more straight forward when compared to other feature extraction techniques available in the literature [45, 58, 21, 50]. The related work is published in:
  Pham, T. D., Chandramohan, V., Zhou, X., and Wong, S. T. C. Robust feature extraction and reduction of mass spectrometry data for cancer classification.*In ICDMW '06: Proceedings of the Sixth IEEE International Conference on Data Mining* - Workshops (Washington, DC, USA, 2006), IEEE Computer Society, pp. 202-206.

- Kernelized fuzzy c-means for classification: In unuspervised learning only the feature extracted data $a_1, ..., a_n \in \Re^n$ is given, i.e., the labels are missing. Standard questions of unsupervised learnings are clustering, density estimation, and data description. As we know from the review, there are lot of methods available to classify the mass spectromertry datasets, but the application of fuzzy clustering algorithm in combination with signal processing techniques is a fairly new idea in analysing mass spectrometry datasets. Biomedical data are usually corrupted with noise, which may degrade the performance of fuzzy c-means (FCM) in classifying mass spectrometry datasets. Inorder to overcome the limitations of FCM, we propose kernelized fuzzy c-means (KFCM), which is realized by replacing the original Euclidean distance with kernelized distance metric.

The main objective of this framework is to reduce false positive and to improve the accuracy of diagnosis. Moreover, the framework will enable the researchers and scientists to understand the effectiveness and the limitations of several existing pattern recognition techniques for processing mass spectrometry datasets. All the prototypes of the framework and methods have been written in MATLAB programming language.

3.5 Overview of the framework

Step1: The prostate cancer datasets are collected from FDA-NCI Clinical Proteomics Program Databank.

Step2: The feature extraction and reduction has been investigated using liner predictive coding.

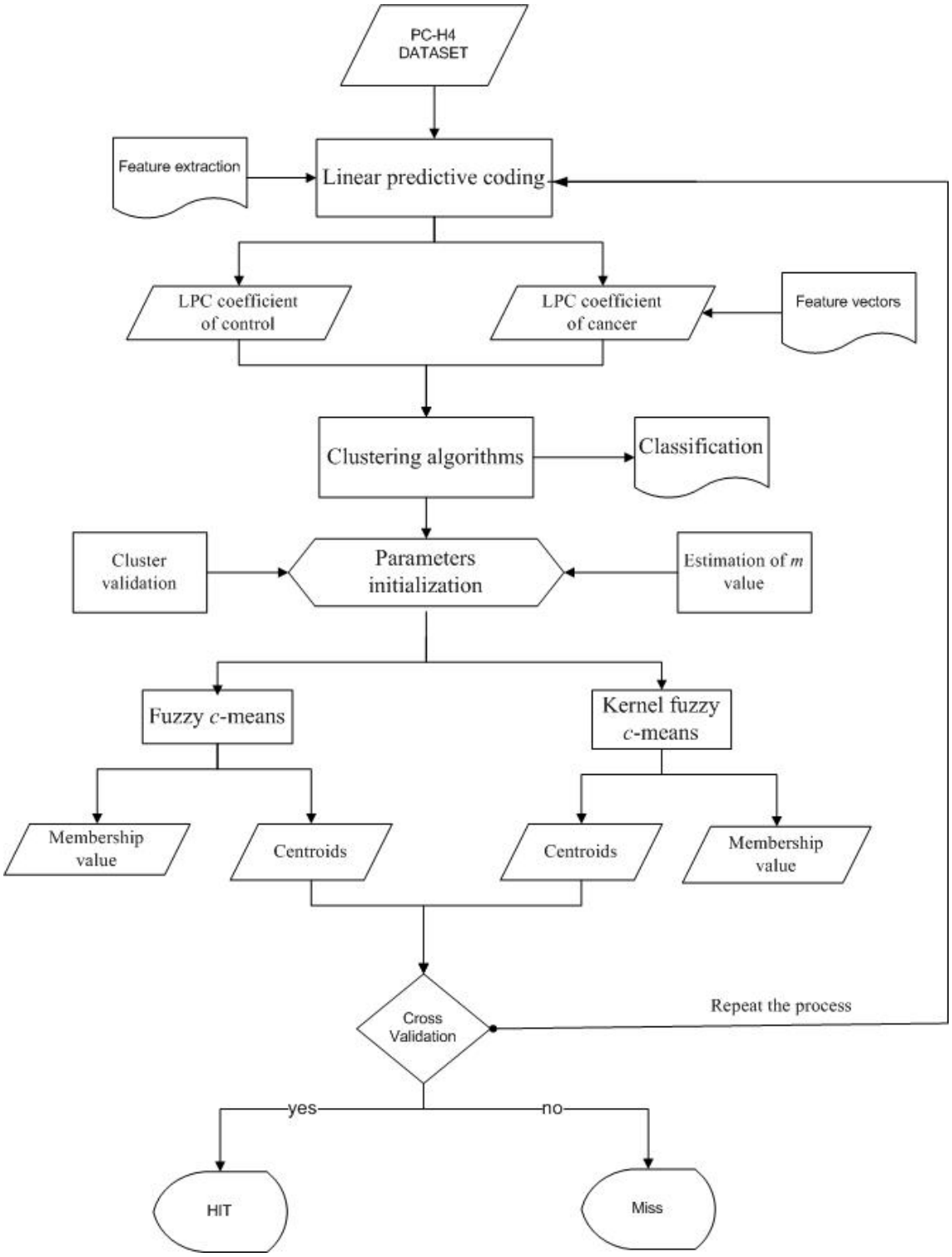


Figure 3.1: *Overview of the analysis pipeline*

Step3: The clustering parameters for fuzzy clustering algorithms has been evaluated.

Step4: The features extracted from the datasets were classified using fuzzy c-means and kernel fuzzy c-means.

Step5: Cross-validation is carried out to check the performance of the clustering algorithms.

Chapter 4

Feature extraction from MS data

In this chapter, we present feature extraction method for mass spectrometry data set, that retains as much as possible initial information content of learning examples, extracts biological meaningful features, reduces the degree of spatial redundancy, and achieves a significant level of dimensionality reduction. This chapter is organized as follows: section 4.2 explains the principle of linear predictive coding (LPC) and its applications to mass spectrometry datasets, section 4.3 explains in detail about semi-variograms and its application in estimating the number of pole values p for the LPC analysis of mass spectra.

4.1 Introduction

Mass spectrometry datasets have several imperfections, thus direct application of machine learning methods are not possible, therefore feature extraction methods are used to determine an appropriate subspace of dimensionality m in the original feature space $d(m < d)$. Features are variables constructed from preprocessed data to summarize the properties of the data and the process of constructing feature are called "feature extraction". Before the definition of feature extraction, i would like to describe the difference between feature selection and feature extraction. The term feature selection refers to algorithms that select the best subset of the input feature set. Methods that create new features based on transformations or combinations of the original feature set are called feature extraction algorithms. Feature extraction can be defined as tool for extracting useful information or patterns from the preprocessed datasets, though good choice of patterns can lead to improvements in clustering performance. Note that feature extraction precedes feature selection; first, features are extracted from the sensed data and then some of the extracted features with low discrimination are discarded [66, 72]. The features generated by feature extraction may provide a better discriminative ability than the best subset of given features, but these new features may not have a clear physical

meaning. The simplest approach to feature extraction from mass spectra is to use abundance(intensity) information of every m/z measured as features. While this approach to feature extraction is straight forward, it places additional demand on the feature selection and classification stages since a very large number of features are used ($\approx 15,000$) and most studies employ a modest number of cases ($\simeq 500$). Moreover, mass spectrometers can only distinguish the masses of proteins within a finite resolution level. More than one m/z measures can correspond to the same protein. Thus, high level of correlation are expected between close m/z values [77, 46]. From a biomedical perspective, it is important to find a moderate number of proteins that most contribute to correct classification, such that these potential biomarkers can be identified and biochemically validated. Therefore it is necessary to extract useful information or patterns from the preprocessed MS datasets, though good choice of patterns can lead to improvements in clustering performance. Principal component analysis (PCA), factor analysis and linear discriminative analysis has been widely used methods in pattern recognition for feature extraction and dimensionality reduction. But recently being realized that signal processing based pattern recongition can provide a set of novel and useful tools for solving highly relevant problems in genomics and proteomics. In [62], the principle of linear predictive coding (LPC) has been effectively applied in SELDI-MS datasets and promising results were obtained in distinguishing healthy from cancer using simple LPC-based decision logic. The researchers reported that the applications of signal-processing based pattern ananlysis can offer effective tools for the study of complex biological problems. Therefore in our experiment, we applied linear predictive coding (LPC) to extract or select the features from the given MS dataset (m/z), though the raw form doesn't convey useful information for the task of classification. Considering MS data as a signal, the features can be extracted and it can be represented as LPC coefficients[60, 62].

4.2 Linear predictive coding (LPC)

The theory of LPC, has been well understood for many years in the field of speech recognition. In this section, we describe in detail about the basis of LPC and its mathematical notations. Before we start describing about LPC, I would like to describe why LPC has been so widely used:

- LPC provide a good model for signal processing.
- LPC is analytically tractable. The method is precise and is simple and straightforward to implement either in the software or hardware.
- LPC model works well in recognition applications.

4.2.1 LPC model

LPC is defined as the correlation between the n-th sample and the p previous samples of the target signal. Let us consider, MS data as a digital signal, the intensity value s_m at position or time n, denoted as $\tilde{s}(n)$, can be calculated as linear combination of previous p samples which can be defined as[44, 62]

$$\tilde{s}(n) = a_1 s(n-1) + a_2(s(n-2) + ... + a_k s(n-k) = \sum_{k=1}^{p} a_k s(n-k) \tag{4.1}$$

where $\tilde{s}(n)$ is the prediction of $s(n)$, $s(n-k)$ is the $k-th$ step previous sample, a_k are called the linear predictive coefficients and p the number of poles. We now perform the prediction error $e(n)$ between the observed sample $s(n)$ and predicted value $\tilde{s}(n)$

$$e(n) = s(n) - \tilde{s}(n) = s(n) - \sum_{k=1}^{p} a_k s(n-k) \tag{4.2}$$

From the above equation, we can optimally determine the predictor coefficients a_k directly from the MS signal by minimizing the sum of squared errors.

$$E_m = \sum_{n} e^2(n) = \left[s(n) - \sum_{k=1}^{p} a_k s(n-k) \right]^2 \tag{4.3}$$

To solve the above the equation for the predictor coefficients, we differentiate E_n with respect to each a_k and set the result to zero.

$$\frac{\partial E_m}{\partial a_K} = 0 \tag{4.4}$$

The result is a set of p linear equations with p unknowns, which can be expressed in matrix form as

$$\boldsymbol{Ra} = \boldsymbol{r} \tag{4.5}$$

where $\boldsymbol{R}$ is a $p \times p$ autocorrelation matrix (Toeplitx matrix symmetric with all diagonal elements being equal), $\boldsymbol{r}$ is a $p \times 1$ autocorrelation vector and $\boldsymbol{a}$ is a $p \times 1$ vector of prediction coefficients:

$$\boldsymbol{R} = \begin{bmatrix} r_n(0) & r_n(1) & \dots & r_n(p-1) \\ r_n(1) & r_n(0) & \dots & r_n(p-2) \\ r_n(2) & r_n(1) & \dots & r_n(p-3) \\ . & . & \dots & . \\ r_n(p-1) & r_n(p-2) & \dots & r_n(p-3) \end{bmatrix}$$

$\boldsymbol{a^T} = [a_1, a_2, ...a_p]$,
where $\boldsymbol{a^T}$ is the transpose of $\boldsymbol{a}$, and
$\boldsymbol{r^T} = [r(1)r(2)r(3)...r(p)]$,
where $\boldsymbol{r^T}$ is the tranpose of $\boldsymbol{r}$.
Thus the LPC coefficient can be obtained by solving [44]

$$\boldsymbol{a} = \boldsymbol{R^{-1}r} \tag{4.6}$$

The feature vectors extracted from MS dataset, will be used for data classification[59] and in the following section we discussed in detail about variograms and its application in estimating the pole values p for LPC analysis.

4.3 Variograms

4.3.1 Introduction

Geostatistics is a term commonly used to describe a set of techniques that model spatial variation in data and researchers use these models to estimate or classify other data based on these models. Geostatistics developed out of empirical approaches developed by (Krige 1989) and were given theoretical validity by the development of random function theory in 1960s (Matheron 1970). The application of geostatistics to the estimation of ore reserves in mining is most well known use. Though, it has been emphasised with time, this estimation techniques can be used wherever a continuous measure is made on a sample at a particular location in space (or time), i.e., where a sample value is expected to be affected by its position and its relationship with neighbours. Therefore, these methods are now widely applied to many areas of mathematical geology and science as a special branch of applied statistics. In order

to provide sufficient background for the present work, we discussed in detail about semi-variograms and interested readers can refer to [17, 34, 61].

4.3.2 Semi-variogram

A variogram is a statistically-based, quantitative and characterizes the spatial continuity or roughness of MS dataset. A variogram is a function of separation vector: this includes both distance and direction. The variogram function yields the average dissimilarity between points separated by the specified vector.
Let we assume, if we had a pairs of samples for a specific h then we could calculate an experimental value for $m(h)$:

$$m(h) = \frac{1}{n} \sum [g(x) - g(x+h)] \tag{4.7}$$

where g stands for grade, x denotes the position of one sample in the pair and $x+h$ denotes the position of other sample, and n is the number of pairs which we have. Having a rig ourselves of $m(h)$, let us turn to the variance of differences. This is called as $2\gamma(h)$ and is usually known as variogram, since it varies with time h. In practice, having made our notrend assumption, we can calculate:

$$2\gamma(h) = \frac{1}{n} \sum [g\,(x) - g\,(x+h)]^2 \tag{4.8}$$

The 2 in front is for the mathematical convenience. The term $\gamma(h)$ is called the experimental semi-variogram. Once a experimental semi-variogram are computed, we can built the model variogram to fit the experimental variogram.

Four types of model functions are supported for building model variograms, but we discussed in detail about two models, which are very commonly used, are as follows
1. Spherical model or Matheron model:

The ideal semi-variogram is the spherical model which was mathematically derived by Matheron. The spherical variogram begins at the origin (zero), raises smoothly to an upper limit, then continues constantly at that level; that is

$$\gamma(h) = \begin{cases} c[\frac{3}{2}\frac{h}{a} - \frac{1}{2}\frac{h^3}{a^3}] & \text{if } \frac{h}{a} \leq a, \\ c & \text{if } \frac{h}{a} \geq a, \\ 0 & \text{if } \frac{h}{a} = 0 \end{cases} \tag{4.9}$$

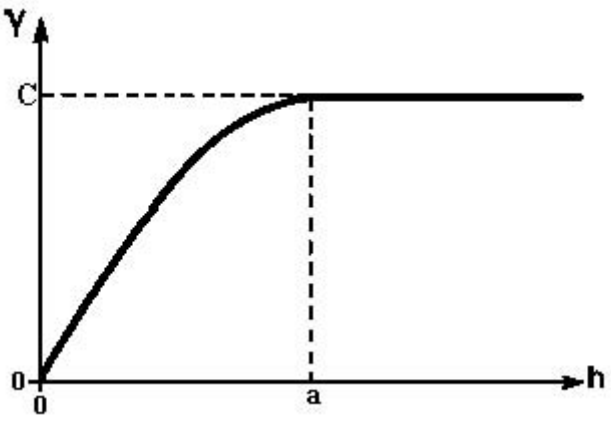


Figure 4.1: *Spherical representation of semi-variogram*

where c is called the "sill" of the semi-variogram, a is the range of the semi-variogram, which can be considered as an optimal number of poles p in the LPC analysis, and h is the lag distance. From the Fig 4.1 [34], it can be seen that the function smoothly increases for all distances up to the range of the semi-variogram, then beyond the range it stays constant at the value of the sill. This model was originally derived from theoretical grounds but has been found to be widely applicable in practice.

2. Exponential model:

   The exponential model is defined as

$$\gamma(h) = c[1 - exp(\frac{3h}{a})] \tag{4.10}$$

This model rises more slowly from the origin than the spherical and never quite reaches its sill. Fig 4.2 [34] shows the spherical and exponential models with the same "range" and "sill", and even shows the comparison of spherical with exponential model, in which the distance between pairs of samples is plotted along the horizontal axis and the value of the semi-variogram along the vertical. The experimental and spherical variograms plotted using prostate cancer dataset (PC-H4) is discussed in chapter 5.

4.4 Conclusion

Mass spectra exhibits an additive high frequency noise component, which affects the both classification methods and human observers in finding meaningful patterns in mass spectra. As we know from the review, many pattern recognition techniques

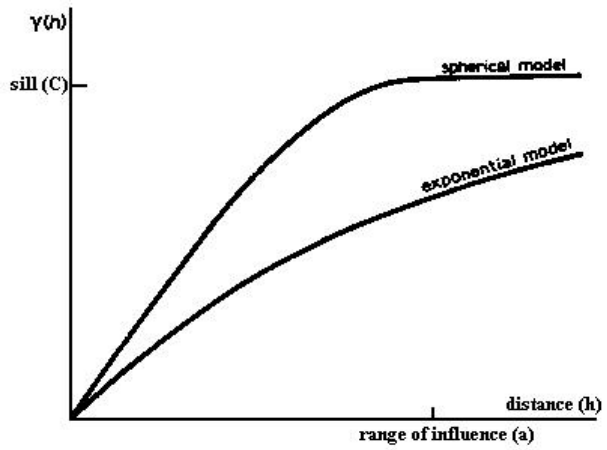


Figure 4.2: *Comparison of the exponential and spherical models*

were not designed to cope with large of irrelevant features. Therefore, feature extraction is performed to extract summary information reflecting the pathological status of a sample from preprocessed mass spectra. We performed the feature extraction for our given mass spectrometry dataset using LPC, and obtained the feature vectors as LPC coefficient. In order to determine the pole value p for LPC analysis, we first used experimental semi-variogram, then we constructed spherical model to fit experimenal semi-variogram. Thus, a suitable number of poles should be assumed equal to the range of the spherical semi-variogram, which is shown in Fig 7.2.

Chapter 5

Clustering algorithms

In this chapter, we describe fuzzy clustering algorithms, particularly those related to the fuzzy c-means (FCM) algorithms. Our aim in this chapter is to define and describe the FCM model. We then describe kernel based algorithms that are based on this model. We also highlight the strength and shortcomings that these various algorithms have in the following subsections.

5.1 Fuzzy clustering

In this section a detailed discussion of fuzzy clustering algorithms is presented. Implementations and results are presented in chapter 6.

5.1.1 Introduction

Clustering involves the task of dividing data points into homogeneous classes or clusters so that items in the same class are as "similar" as possible and items in different classes are as "dissimilar" as possible. Clustering can also be thought as a form of data compression, where a large number of samples are converted into a smaller number of prototypes or clusters. Depending on the data and the application, different types of similarity measures control, how the clusters are formed. Some examples of values that can be used as similarity measures include distance, connectivity, and intensity.

For clustering techniques generating crisp partitions, each data point belongs to exactly one cluster. This requirement has led to the development of fuzzy clustering methods. One of the widely used fuzzy clustering methods is the fuzzy c-means (FCM) algorithm. FCM is a fuzzy partitional clustering approach, and can be seen as an improvement and a generlisation of k-means. In fuzzy clustering, the data points can belong to more that one cluster center, and associated with each of the points are membership grades which indicate the degree to which the data points

belong to different clusters. However, it is often useful for each data point to admit multiple and non-dichotomous cluster memberships.

5.1.2 Fuzzy *c*-means algorithm (FCM)

Fuzzy clustering algorithms has been widely used in the field of bioinformatics, engineering and pattern recognition. It was proposed by Dunn [9] and generalised by Bezdek [6] and best understood by contrasting it with more common hard clustering of a dataset. The FCM algorithm, partitions the data into groups with different membership grade between 0 and 1.

The aim of the FCM algorithm is to find the desired point in each cluster, which can be considered as the centroid of the cluster, and then, the grade of membership for each object in the clusters. Such an aim can be achieved by minimizing the objective function and can be defined as follows [6]

$$J_{FCM}(U,V) = \sum_{i=1}^{n}\sum_{j=1}^{c} u_{ij}^{m} \left\|a_i - v_j\right\|^2, 1 < m < \infty \tag{5.1}$$

where

- n is the total number of patterns in a given data set, and c is the number of centers.
- $A = \{a_1, a_2, .., a_n\} \subset R^s$ and $V = \{v_1, v_2, .., v_c\} \subset R^s$ are the feature extracted MS training set and cluster centroids.
- m is the fuzziness parameter that determines the fuzziness of the centroids. At $m = 1$ FCM collapses to HCM, giving crisp results. At very large values of m, all the points will have equal memberships with all the clusters.
- u_{ij} is the degree of membership.
- v_j is the cluster centers, and $\|a_i - v_j\|$ denotes the Euclidean distance.

5.1.3 Conditions for optimality

Fuzzy partitioning is carried out by iterative optimization with the update of membership u_{ij} and the cluster centers v_j. These conditions are derived in [Bezdek, 1981] and are defined as follows

$$u_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{\|a_i - v_j\|^2}{\|a_i - v_k\|^2} \right)^{\frac{2}{m-1}}} \tag{5.2}$$

$$v_j = \frac{\sum_{i=1}^{n} u_{ij} a_i}{\sum_{i=1}^{n} u_{ij}^m} \tag{5.3}$$

For the calculation of a cluster center v_j, all input samples are considered and the contributions of the samples are weighted by the membership values. For each sample, its membership value in each class depends on its distance to the corresponding cluster center. The weight factor m reduces the influence of small membership values. The large the value of m, the smaller the influence of samples with small membership values.

5.1.4 The algorithm

1. Input the number of clusters c, fuzzifier m and the distance function $\|*\|$.
2. Intialize the cluster centers $v_j^0 (j = 1, 2, ..., c)$.
3. Calculate $u_{ij} (i = 1, 2, ..., n; j = 1, 2, ..., c)$ using Eq. (5.2).
4. Calculate the centroids $v_j^1 (j = 1, 2, ..., c)$ using Eq. (5.3).
5. If $max_{1 \le j \le c} (\left\| v_j^0 - v_j^1 \right\| / \left\| v_j^1 \right\|) \le \epsilon$ then go to step 6; else let $v_i^0 = v_i^1 (1 = 1, 2, ..., c)$ and go to 3.
6. Output the clustering results: cluster centers $v_j^1 (j = 1, 2, ..., c)$, membership matrix U.
7. Stop.

By iteratively updating Eqs. (5.2) and (5.3), fuzzy partition u_{ij} and cluster center v_j are updated, until the cost function reaches the minimal value or can't be reduced further.

5.1.5 Strength and weakness

The FCM algorithm has proven a very popular method of clustering for many reasons. It is relatively straight forward in the programming implementation. It employs an objective function that is intuitive and easy-to-grasp. FCM works well

with data sets composed of hyper-spherically-shaped well-separated clusters in determining the clusters accurately. Though FCM is based on fuzzy basis, it performs robustly: it always converges to a solution, and it provides consistent membership values.
The drawbacks of FCM are as follows :

1. Cluster centers to be determined based on priori knowledge.
2. It requires the initialization of prototypes, good initialization are difficult to assess.
3. It is an iterative algorithm aims for finding the solutions of the objective function, it may find more than one solution depending on the initialization.
4. It look for same cluster shapes, different shapes cannot be mixed.
5. Its accuracy is sensitive to noise and outliers. This is studied comprehensively in chapter 7.

5.2 Kernel based fuzzy *c*-means algorithm(KFCM)

Despite the weakness of FCM have led researchers to generalize and extend it further to make a mature platform for clustering. To overcome the above mentioned problems in FCM, the researchers proposed a lot of algorithms by replacing the original Euclidean measure with other similarity measures. A recent development is to use kernel trick to construct the kernel versions of the FCM algorithm. In this section, we discussed about kernel based methods for unsupervised learning algorithms [71, 23, 15, 84]. The common philosophy of these clusterings algorithms is to perform the clustering in the feature space. Then we discussed about the implementation of kernel methods into fuzzy clustering algorithms and some variants. Finally, we described about strength and short comes out of KFCM.

5.2.1 Kernel methods

In machine learning, the use of the kernel functions has been introduced by *Aizerman et al.*[2] in 1964. Kernel representations offer an alternative solution by projecting the data into a high dimensional feature space to increase the computational power. In the mid of 90's Cortes and Vapnik introduced support vector machines (SVM) which perform better than other classification algorithms in several problems. The success of SVM has bought to extend the use of kernels to other learning algorithms (e.g Kernel PCA, KFD)[70]. Figure 5.1 shows an example of a feature mapping from an two dimensional input space to a two dimensional feature space, where the

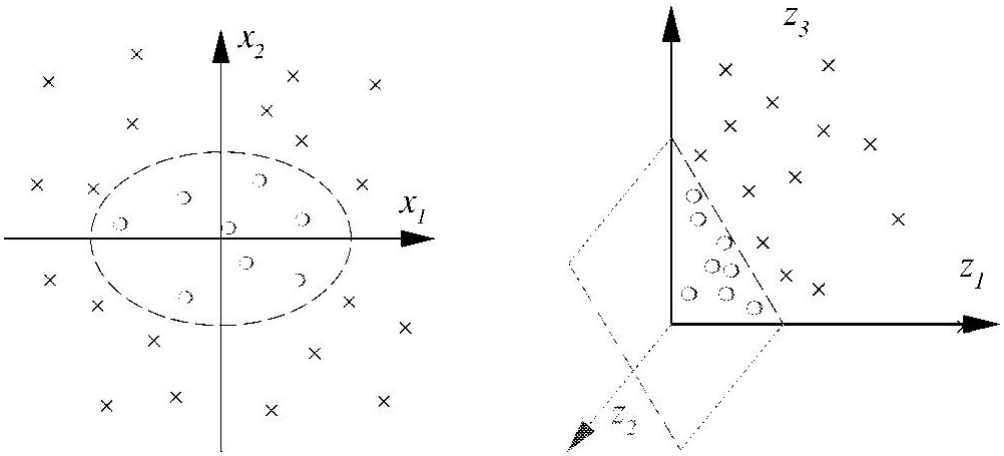


Figure 5.1: *Two-dimensional classification example, using the second-order monomials $x_1^2, \sqrt{2}x_1x_2$ and x_2^2 as features a separation in feature space can be found using hyperplane [69]*

data cannot be separated by a linear function in the input space, but can be in the feature space. Since the computation of a linear classifier in the feature space uses only the scalar products, the whole leearning process can be expressed in terms of kernels [53].
Consider the mapping function

$$\phi : \Re^N \rightarrow \tau$$

$$\mathrm{x} \rightarrow \phi(x)$$

and the computation between two scalar product between two feature space vectors, can be readily formulated in terms of kernel function K

$$\begin{aligned} \phi(x_i).\phi(x_j) &= \left(x_1^2, \sqrt{2}x_1x_2, x_2^2\right)\left(y_1^2, \sqrt{2}y_1y_2, y_2^2\right)^T \\ &= \left((x_1, x_2)(y_1, y_2)^T\right)^2 \\ &= (x.y)^2 \\ &= K(x, y). \end{aligned} \tag{5.4}$$

This find generalizes:

- For x, $y \in \Re^N$, and $d \in N$ the kernel function

$$K(x, y) = (x.y)^d \tag{5.5}$$

computes a scalar product in the space of all products of d vector entries (monomials) of x and y.

- If $K : C \times C \to \Re$ is a continuous kernel of a positive integral operator on a Hilbert space $L_2(C)$ on a compact set $C \in \Re^N$, i.e.,

$$\forall f \in L_2(C) : \int_{C \times C} K(x, y)\ f(x) f(y) dx dy \geq 0 \tag{5.6}$$

then there exists a space τ and a mapping $\phi : \Re^N \to \tau$ such that $K(x, y) = (\phi(x_i).\phi(x_j))$, which can be seen directly from the Mercer's theorem [69].

One of the most relevant aspects in applications is that it is possible to compute Euclidean distance in τ without knowing explicitly ϕ. This can be done using the so called distance kernel trick. The kernel trick transforms any algorithm that solely depends on the dot product between two vectors. Wherever a dot product is used, it is replaced with the kernel function. Thus, a linear algorithm can easily be transformed into a non-linear algorithm. This non-linear algorithm is equivalent to the linear algorithm operating in the range space of ϕ and defined as follows [23]:

$$\begin{aligned} \|\phi(x_i) - \phi(x_j)\|^2 &= (\phi(x_i) - \phi(x_j)).(\phi(x_i) - \phi(x_j)) \\ &= \phi(x_i).\phi(x_i) + \phi(x_j).\phi(x_j) - 2\phi(x_i).\phi(x_j) \\ &= K(x_i, x_i) + K(x_j, x_j) - 2K(x_i, x_j) \end{aligned} \tag{5.7}$$

in which the computation of distances of vectors in feature space is just a function of the input vectors. Inorder to simplify the Gram matrix K where each element K_{ij} is the scalar product $\phi(x_i).\phi(x_j)$. Thus, Eq. (5.7) can be rewritten as

$$\|\phi(x_i) - \phi(x_j)\|^2 = K_{ii} + K_{jj} - 2K_{ij} \tag{5.8}$$

Three commonly used kernel functions are
(1) Gaussian radial basis function (RBF) kernel :

$$K(x,y) = exp(\frac{-\|x-y\|^2}{\sigma^2}) \tag{5.9}$$

(2) Polynomial kernel :

$$K(x,y) = (1 + \langle x,y\rangle)^d \tag{5.10}$$

(3) Sigmoid kernel:

$$K(x,y) = tanh(\alpha \langle x,y\rangle + \beta) \tag{5.11}$$

where σ, d, α, β are the adjustable parameters of the above kernel functions. For the sigmoid function, only a set of parameters satisfying the mercer theorem can be used to define a kernel function.
In literature, there are some applications of kernels in clustering. These methods are broadly classified into three categories which are based on:

- Kernelization of the metric: Kernelization of metric methods are based on centroids and the distance between patterns (a) and centroids (v) is computed by means of kernels:

$$\|\phi(a_i) - \phi(v_j)\|^2 = K(a_i,a_i) + K(v_j,v_j) - 2K(a_i,v_j) \tag{5.12}$$

- Clustering in feature space: Clustering in feature space is made by mapping each pattern using the function ϕ and then computing in feature space. Considering $\phi(v_i)$ the centroids in feature space and $\phi(a_i)$ as the feature extracted MS data, it is possible to calculate the distances $\|\phi(a_i) - \phi(v_j)\|$ using kernel trick.

- Description via support vectors: The description via support vectors make use of one class SVM to find a minimum enclosing sphere in feature space. The support vector clustering algorithm allows to assign labels to patterns (a) in input space enclosed by the same surface[23].

5.2.2 Kernel fuzzy *c*-means

Given the feature extracted MS data set $A = \{a_1, a_2, a_3, ..., a_n\}$, we map our data into some feature space τ, by means of nonlinear mapping ϕ, therefore the clustering process can be carried out in feature space rather than input space, which avoids the restriction of FCM [**?**]. Minimization of objective function with respect to U is given by [15, **?**]

$$J_{KFCM}(U,V) = \sum_{i=1}^{n}\sum_{j=1}^{c} u_{ij}^{m} \left\|\phi(a_i) - \phi(v_j)\right\|^2, 1 \leq m \leq \infty \tag{5.13}$$

where ϕ is an implicit nonlinear map, then with kernel trick we have

$$\left\|\phi(x_i) - \phi(v_j)\right\|^2 = K(a_i, a_i) + K(v_j, v_j) - 2K(a_i, v_j) \tag{5.14}$$

Here we adopt Gaussian RBF kernel, so $K(x,x) = 1$ and can be simplified to

$$J_K FCM(U,V) = 2\sum_{i=1}^{n}\sum_{j=1}^{c} u_{ij}^{m}(1 - K(a_i, v_j)) \tag{5.15}$$

In similar way to standard FCM algorithm, the objective function J_{KFCM} can be minimized under the constraint of U.

$$u_{ik} = \frac{(1/(1 - K(a_i, v_j)))^{-1}/(m-1)}{\sum_{k=1}^{c}(1/(1 - (K(a_i, v_k)))^{-1}/(m-1)} \tag{5.16}$$

$$v_j = \frac{\sum_{i=1}^{n} u_{ij}^{m} K(a_i, v_j) a_i}{\sum_{i=1}^{n} u_{ij}^{m} K(a_i, v_j)} \tag{5.17}$$

The above Eqs. (5.16) and (5.17) are derived using Gaussian RBF kernel. The reason is that the derivative of $J_{FCM}(U,V)$ with respect to v_j using a Gaussian kernel is particularly simple since it allow us to use the kernel trick and even more appropiate for noisy data.

5.2.3 Strength and weakness

As we know FCM uses the square-norm to measure similarity between prototypes and data points, it can be effective in clustering spherical clusters, which is being overcome by replacing the original distance metric with kernel-induced distance metric. The problem of outliers and noise is rectified by using gaussian RBF kernel functions. The drawbacks are as follows:

1. In kernel methods, the choice of the kernel has a crucial effect on the performance, i.e., if does not choose the correct kernel property, one will not achieve the excellent performance in classification.

2. In kernel clustering, the clustering prototypes lies in high dimensional space and hence lack clear and intuitive descriptions unless using additional projection approximation from the feature to the data space.

5.3 Cluster validation

5.3.1 Introduction

Clustering plays a vital role in many engineering fields such as pattern recognition, system modeling, image processing, communication, data mining and so on. It serves as a tool to assess the relationships among patterns of the data set by organizing the patterns into groups or clusters. Many algorithms for both hard and fuzzy clustering were developed to accomplish this. An intimately is the cluster validity which deals with the significance of the structure imposed by clustering method. In practical application, we need cluster validity methods to measure the quality of the clustering results. Many factors can influence the quality of clustering results such as the method of initialization, the choice of the number of classes c, and the clustering method. A validity function is a function which assigns the output of FCM a number which is intended to measure the quality of the clustering provided by the output. By evaluating the output for a variety of choices of c, one hopes to be able to determine the values of these parameters for which the corresponding clustering best identifies the structure in the data. The quality of a clustering algorithm is indicated by how closely the data points are associated to the cluster centers.

5.3.2 FCM-based model selection algorithm

1. Choose c_{min} and c_{max}

2. For c= c_{min} to c_{max}

(a) Initialize cluster centers (V).

(b) Apply clustering algorithm to update the membership matrix (U) and the cluster centers (V).

(c) Test for convergence; if not, go to (b).

(d) Compute a validity value $V_d(c)$.

3. Compute c_f such that the cluster validity index $V_d(c_f)$ is optimal.

Fuzzy clustering algorithm and kernel based fuzzy clustering algorithm are run over a range of c values$(2, ..., c_{max})$, and the resulting fuzzy partition is evaluated with the validity indices to identify the optimal number of clusters. A number of validity measures for fuzzy clusters exist in the literature, but we take into consideration three well known measures to validate FCM and KFCM with our dataset.

5.3.3 Validity indices

A validity function has been performed to measure the quality of the clustering provided by FCM and KFCM. We briefly review some of the most frequently referred validity indices for fuzzy clustering [7, 65, 80, 79].

Bezdek proposed two cluster validity indices for fuzzy clustering. These indices, which are referred to as partition coefficient (V_{PC}) and classification entropy (V_{CE})[6, 79]. Partition coefficient (V_{PC}) is defined as

$$V_{PC} = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{c}(u_{ij})^2 \tag{5.18}$$

Classification entropy is defined as

$$V_{CE} = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{c}(u_{ij})log_a(u_{ij}) \tag{5.19}$$

where u_{ij} ,$(i = 1, 2, ..., n; j = 1, 2, ..., c)$ is the membership of datapoint i in cluster j.

The above mentioned V_{PC} and V_{CE} are based on using the membership values u_{ij} of fuzzy partition. The drawback of these indices are monotonous dependency on the number of clusters c and lack of direct connection to the geometry of the data(Dave, 1996), since they do not use the data itself. The following indices simultaneously take into account the membership functions and the structure of data.

Xie and Beni (1991) [79] proposed a validity index V_{XB} which overcome the above problems in V_{PC} and V_{CE} and mainly focus on two following properties: compactness and separation [65, 79]. In the equation (5.20) for V_{XB}, the numerator indicates the compactness of the fuzzy partition, while the denominator indicates the strength of the separation between clusters.

$$XB(c) = \frac{\sum_{c}^{j=1} \sum_{n}^{i=1} \left[(u_{ij})^m \left\| a_i - v_j \right\|^2 \right]}{n \left[min_{i,j} \left\| a_i - v_j \right\|^2 \right]} \tag{5.20}$$

The main disadvantage of this index is that it tends to decrease monotonically when c is very large. They stated that good partition produces a small value for the compactness, and that well-separated v_j will produce a high value for the separation [79]. Hence the most desirable partition is obtained by minimizing V_{XB} for $c_{j=2,3,\ldots,c_{max}}$.

Zahid et al. (1999) [83] proposed the validity index V_{ZLE}, based on the concepts of fuzzy compactness and fuzzy separation to the traditional validity indices, which considers the geometrical properties of the data structure and membership functions. It is defined as

$$V_{ZLE} = SC_1(c) - SC_2(c), \tag{5.21}$$

where

$$SC_1(c) = \frac{\sum_{j=1}^{c} \left\| v_j - \bar{v_j} \right\|^2 / c}{\sum_{j=1}^{c} (\sum_{i=1}^{n} u_{ij}^m \left\| a_i - v_j \right\|^2 / \sum_{i=1}^{n} u_{ij})} \tag{5.22}$$

and

$$SC_2 = \frac{\left(\frac{\sum_{j=1}^{c} \sum_{l=i+1}^{n} (\sum_{i=1}^{n} \left[min(u_{ij}, u_{lj}) \right]^2}{\sum_{i=1}^{n} min(u_{ij}, u_{ij})} \right)}{\left(\frac{\sum_{i=1}^{n} (max_{1 \le j \le c} u_{ij})^2}{\sum_{i=1}^{n} max_{1 \le j \le c} u_{ij}} \right)} \tag{5.23}$$

The index V_{ZLE} uses a fuzzy union and a fuzzy intersection to obtain the fuzzy compactness/fuzzy separation degree. The maximum of V_{ZLE}, as a function of the number of clusters c, is sought for a well-defined c-partition.

Geva et al. (2000) [28], proposed the fuzzy hypervolume validity V_{FHV}, based on the concepts of hyervolume and density, which is defined as

$$V_{FHV} = \sum_{j=1}^{c} [det(F_j)]^{1/2}, \tag{5.24}$$

where

$$F_j = \frac{\sum_{i=1}^{n}(u_{ij})^m(a_i - v_j)(a_i - v_j)^T}{\sum_{i=1}^{n}(u_{ij})^m.} \tag{5.25}$$

The matrix F_j denotes the fuzzy covariance matrix of cluster j. A fuzzy partition can be expected to have low V_{FHV} value if the partition is tight. Thus, we can find the optimal c, by solving $min_{2\leq c\leq n-1}V_{FHV}$ to produce best clustering performance for the given dataset.

Sun et al. (2004) [74] proposed an index that measures separation between the clusters and the cohesion within clusters. This index is based on the linear combination of the average within cluster scattering (inversely realted to compactness) and between–cluster separation. A cluster number which minimizes V_{WSJ} corresponds to the best clustering.

$$V_{WSJ}(U, V, c) = Scat(c) + \frac{Sep(c)}{Sep(c_{max})}, \tag{5.26}$$

where $scat(c)$ represents the compactness of the obtained clusters.

$$Scat(c) = (\frac{1}{c}\sum_{j=1}^{c} \|\sigma(v_j)\|)/(\|\sigma(A)\|) \tag{5.27}$$

and the separation between clusters is defined as

$$Sep(c) = \frac{D_{max}^2}{D_{max}^2}\sum_{j=1}^{c}\left(\sum_{k=1}^{c}\|v_j - v_k\|^2\right)^{-1} \tag{5.28}$$

Bougessa et.al (2006) [7] proposed a validity measure, based on the concept of separation and compactness, which utilize the covariance structure of clusters. It is defined as

$$V_{SC}(c) = Sep(c)/Comp(c) \tag{5.29}$$

In the above formula, $Sep(c)$ represents the fuzzy separation of fuzzy clusters given by $Sep(c) = trace(S_B)$ where S_B is the fuzzy scatter matrix, defined as

$$S_B = \sum_{i=1}^{n}\sum_{j=1}^{c} u_{ij}^{m}(v_j - \bar{v})(v_j - \bar{v})^T \tag{5.30}$$

The large value of $Sep(c)$ indicates that the fuzzy c-partition is characterized by well-separated fuzzy clusters.

$Comp(c)$ represents the total compactness of the fuzzy c-partition, and is given by

$$Comp(c) = \sum_{j=1}^{c} trace(\sum_{j}) \tag{5.31}$$

where $\sum_j$ represents the covariance matrix. Hence the most desirable partition is obtained by maximizing V_{SC} for $c_{j=2,3,\ldots,c_{max}}$.

For our experiments, we choose three traditional validity indices namely V_{PC}, V_{CE}, V_{XB} for fuzzy clustering. They have a common objective of finding a optimal c with each of these c centers. Note that since no single validity indices performs well for all the datasets. As Bezdek [55] stated, "no matter how good your index is, there is a data set out there waiting to trick it".

5.4 Exponent value validation

5.4.1 Introduction

As we know that, fuzzy clustering results are mainly influenced by two factors namely the cluster center c and the weighting exponent or smooth factor m. In unsupervised fuzzy clustering, the number of classes is determined using validity measures, which we discussed early in this chapter. This leaves the value of the fuzzy exponent m to be determined and its determination is problematic. As m approaches 1, the clustering becomes harder. As m becomes very large (i.e $m \geq 100$), the membership

becomes almost constant and so fuzzy that virtually no cluster would be distinguished. Since the parameter m is not constrained at the upper end, it poses the question, what m value should be taken and whether there is a value that optimizes classification. In the literature about FCM, various proposals or the use of single m-value(m=2 being the most popular) for the FCM, processing of any particular datatype would not be good enough and may be misleading. This is because every dataset has a unique data structure. The optimal $m - value$ should be peculiar to each particular data set and should be sought from within the data structure of each data set. Thus an initial stage in any further application of fuzzy methods is the determination of optimal number of exponent value. Over the years, different range and values for the optimal choice of m have beeen proposed and used by different researchers. In the following subsection we discussed about some of the techniques, which is available to determine the exponent value m for fuzzy clustering.

5.4.2 Estimation of m value

Bezdek suggested the range 1-30, with the range 1.5-3 gives good results. He also gave an interesting interpretation of the special case where $m = 2$. It was noted, however there is no strong theoretical justification or emprical evidence for these choices[54].

McBratney and Moore (1985) [52], made investigations about determining the optimal m-value for fuzzy clustering algorithms. They reported that objective function, decreases monotonically with increasing number of groups and increasing values for m, and thats its rate of change with changing m is not constant. Then they observed the greatest change occured around when $m = 2$. Their procedure involves the combination of optimal number of classes c and the fuzzy exponent m, which is defined as

$$\phi = -\left[\left(\frac{dJ_m}{dm}\right)\sqrt{c}\right] \tag{5.32}$$

where $\frac{dJ_m}{dm}$ is the derivative of the objective function J_m and the fuzzy exponent value m. They carried out some numerical tests using this procedure and got $m = 2$ as an optimal value for fuzzy clustering.

Choe and Jordan (1992) [12], proposed fuzzy decision theory to determine the optimal value m for fuzzy c-means. They defined fuzzy goal as a good cluster criteria and a fuzzy constraint for minimizing the sum of square errors, they choose the value of m based on the maximum membership value obtained by the intersection of the fuzzy goal and fuzzy constraint. They reported that the FCM algorithm was

relatively insensitive to the value choosen for m in the range between 8 and 30, and they further suggested that the value $m = 2$ was optimal.

Deer and Eklund (2003) [18], investigated the value of the fuzzy exponent, m, in a supervised mahalanobis distance fuzzy classifier by requiring that the fuzzy class memberships reflect proportions of contributing classes in the pixels of a remotely sensed image. They finally came to the conclusion that the range lies between 1.6 and 3 to obtain good classification accuracy using FCM.

Dembele et al. (2003) [19], presented an investigation to determine the fuzzy exponent value m for micro array datasets. As a intial start they decided to choose the upper bound value m_{ub} for m, above which the membership value resulting from FCM are equal to $\frac{1}{j}$, where j is the cluster centers, which is showm by (Bedzek 1981, p.73). In their study, they hypothesise that when m varies, there might be a relationship between the FCM membership values and the coefficient of variation (CV) of the set of distance between genes, which is defined as

$$Y_m = \left\{ \left[d^2(x_i, x_j) \right]^{\frac{1}{m-1}} ; j \neq 1, 2, ..., c \right\} \tag{5.33}$$

They carried out some numerical experiments by varying m and determined CV of Y_m. In each case, they observed that the values of m which leads to membership values close to $\frac{1}{j}$ gave a CV of Y_m close to $0.03p$, p represents the dimensionality of the data. But they dont have any theoretical justification for that observation. They proposed the following equation to evaluate the upperbound value m_{ub}, which is given as

$$cv\left\{Y_m\right\} = \frac{\sigma_{Y_m}}{\bar{Y}_m} \approx 0.03p \tag{5.34}$$

where σ_{Y_m} and $\bar{Y}_m$ are respectively the standard deviation and the mean of the set Y_m. They solved the above equation using dichotomy search strategy. After conducting the experiment, they decided the fuzzy exponent value m lies between 1-3.

Yu et al. (2004) [82], presented theoretical and numerical analysis of the fuzzy exponent. They proposed a new approach to determine the weighting exponent in the FCM. The two theoretical rules for selecting the fuzzy exponent value are as follows:

$$\alpha : m \leq \frac{min(s, n-1)}{min(s, n-1) - 2}, if\ min(n-1, s) \geq 3. \tag{5.35}$$

$$\beta : m \leq \frac{1}{1 - 2\lambda_{max}(Fu)}, if\ \lambda_{max}(Fu) < 0.5. \tag{5.36}$$

where $Fu = H^T H/n; s$ and $\lambda_{max}(Fu)$ are the number of nonzero eigen-values and the maximum eigen-value of the matrix Fu respectively and

$$H = \left[\frac{(x_1 - x)}{\|x_1 - x\|}, \frac{(x_2 - x)}{\|x_2 - x\|}, ..., \frac{(x_n - x)}{\|x_n - x\|}\right] \tag{5.37}$$

The above specified rules will provide theoretical upper bounds for the valid fuzzy exponent in the FCM. However, when $_{max}(Fu) \geq 0.5$, both the rules become invalid and the selection of fuzzy exponent then depends on the discretion of the user.

Most recently, Francis et al. (2006) proposed a linear mixture model approach to select the fuzzy exponent value in the fuzzy c-means algorithm, which is the extended work of Deer and Eklund (2003). They determined the optimal cluster centers using existing validity measures and implemented the FCM algorithm to compute fuzzy prototypes and fuzzy membership grades with intializing $m = 1.1$. They computed and recorded σ, the difference between the original data set and the predicted data, which is calculated using Euclidean distance measure and choose the appropiate maximum value for m. They reported that the optimal value for the fuzzy clustering algorithm lies between 1.4 and 2.5 [54].

Inorder to determine the exponent value m, we carried out our experiments, similar to the work done by Dembele et al. (2003). Generally, there has not been universely accepted procedures or unified approach for choice of optimal exponent value for m. In review study, they stated that all the proposals lack appropiate approaches for selecting optimal m-value and that the open problem of choice for optimal m-value still calls for future investigation.

5.5 Conclusion

Unsupervised clustering is the classical problem in pattern recognition. Many clustering algorithms using Euclidean distance construction may have problems with different sizes and cluster shape and even sensitive to noise environment. However, to overcome this above mentioned problem we used kernel based fuzzy clustering algorithm which seems to be robust to noise and outliers and also tolerates unequal cluster size. As we know there are different types of kernel functions available, but we used gaussian RBF kernel, because their derivatives are simple and even this

is the most commonly cited kernel functions in the literature. The relationship between FCM and KFCM was also being discussed. The experimental results dicussed in chapter 7 shows that KFCM has the best performance among other classification methods available in the literature. We report that KFCM is the better choice in analysing prostate cancer dataset.

In fuzzy clustering algorithm, it aims to identify the structure that is present in the dataset. Though the environment is fuzzy, the aim of the clustering algorithm is to generate well defined fuzzy$c$-partition that is as close as to the structure of the given data. Thus, it raises a question like how partition fits the "unknown" structure that is imposed by the used clustering algorithms? and which groupings is better?. Therefore, it's necessary to validate the cluster, which helps to determine the optimal number of clusters. In addition to the number of clusters c, FCM requires a priori choice of the degree of fuzziness $(m-value)$, called as fuzzy exponent value. In the literature of clustering, a large number of cluster validity indices and even various other methods being proposed for determining the optimal m-value, but still it remains a open problem. More developments are expected before it can be effectively used in practical applications.

When specially considering RBF kernel for replacing the original Euclidean distance in fuzzy clustering, it raised the following questions regarding 1) Choice of the type of kernel: This is one of the major questions under consideration regarding research being undertaken in kernel methods. Clearly the choice of kernel will depends on the data, however in the specific case of data partioning then a kernel will have universal approximation qualities such as the RBF is most appropiate [23]. This specific RBF kernel provides a simple and elegant method of feature space data partitioning based on a sum-of-squares as defined in Eq: (5.13), 2) Choice of the kernel width: The other problem raises out of this method is then the choice of the kernel width σ. This particular concern is pervasive in all methods of unsupervised learning, the selection of an appropiate model parameter, or indeed model, in an unsupervised manner. Clearly cross-validation are required to estimate the width of the kernel in the model. Therefore we carried out leave-out-one cross-validation method to determine the σ (kernel width) and the parameter value is shown in chapter 7.

Chapter 6

Classification measure

6.1 Clustering-based decision rule

6.1.1 Introduction

The MS cancer classification based on LPC and unsupervised fuzzy clustering algorithms is as follows: As first step towards classification, we analysed PC-H4(prostate cancer) dataset using LPC to extract useful information and then the resultant LPC vectors $A_i = \{a_{1i}, a_{2i}, ..., a_{ni}\}$, where $i = \{1, ..., n\}$, are then grouped into c cluster centers $V = \{v_1, v_2, ...v_c\}$ using fuzzy clustering algorithms according to the number of different classes. The distortion with respect to the fuzzy cluster centers are accumulated across the whole test to determine the minimum distortion measure beteween an unknown sample A_m^* and the particular known class i.
Measuring the dissimilarities between two feature vectors is the key component of most pattern-recognition algorithms. Let us consider, two vectors, x and y defined on a vector space $\Re$. We can define the cartesian product as $\tau \times \tau$ as a real-valued function in the distance function d on the vector space τ, if it satisfies the following properties [44]:

1. $0 \leq d(x, y) < \infty$ for $x, y \in \tau$ and $d(x, y) = 0$ if and only if $x = y$
2. $d(x, y) = d(y, x)$ for $x, y \in \tau$
3. $d(x, y) \leq d(x, z) + d(y, z)$ for $x, y, z \in \tau$

The above specified properties are commonly referred as the positive definiteness, symmetry, and the triangle conditions. If a measure of dissimilarity, satisfies the positive definiteness, then we say as distortion measure when vectors are representations of signal spectra. In the following subsection, we discussed about the mathematical properties of these distortion measures and we denote distortion measure in terms of D. To calculate a distortion measure between two vectors x and y, denoted as

$D(x, y)$, we calculate a cost of reproducing any input vector x as a reproduction of vector y. Given a distortion measure, the mismatch between two signals can be quantified by an minimum distortion between the input and the final reproduction. The distance measure for comparing vectors x and y is of the form

$$D(x, y) = D \begin{cases} = 0 & if x = y \\ > 0 & otherwise \end{cases} \tag{6.1}$$

There are several measures of distortion developed for speech recognition [59] such as the log-likelihood-ratio distortion, Itakura-saito distortion measure, likelihood-ratio distortion and cepstral distortion measure. But the most commonly used distortion measures for LPC vectors are cepstrum and likelihood, which is discussed in detail in the following subsection.

6.1.2 Cepstral distortion measure

A formal way of justifying the use of cepstral window is to consider the Fourier representation of the log magnitude spectrum. The Fourier transform is used to transform a continuous time signal into the frequency domain. It describes the continuous spectrum of a nonperiodic time signal [44]. Therefore, we consider the complex cepstrum of MS signal as the Fourier transform of the log of the signal spectrum. For the power spectrum $S(\omega)$, which is symmetric with respect to $\omega = 0$ and is periodic for sampled data sequence, the Fourier series representation of $logS(\omega)$ can be expressed as

$$logS(\omega) = \sum_{n=-\infty}^{\infty} a_n e^{-jn\omega} \tag{6.2}$$

where $a_n = a_{-n}$ are real and referred as cepstral coefficients, $e^{-jn\omega}$ is the fourier transform of the given MS signal. Note that

$$c_0 = \int_{-\pi}^{\pi} logS(\omega) \frac{d\omega}{2\pi}. \tag{6.3}$$

Consider $S(\omega)$ and $S'(\omega)$ to be the power spectra of two MS signals and apply the Parvesval's theorem, the L_2-norm cepstral distance between $S(\omega)$ and $S'(\omega)$ can be

related to the root-mean-square log spectral distance as [44]

$$D_c^2 = \int_{-\pi}^{\pi} |logS(\omega) - logS'(\omega)|^2 \frac{d\omega}{2\pi} = \sum_{n=-\infty}^{\infty} (a_n - a'_n)^2, \tag{6.4}$$

where a_n and a'_n are the cepstral coefficients of $s(\omega)$ and $s'(\omega)$, respectively. Though the power spectra are even functions, the cepstral coefficients are real. Since the cepstrum is a decaying signal, the summation in Eq (6.4) doesnot require infinite number of terms [59]. For LPC models it represents the highly smoothed signal and it is usually truncated to only small number of terms. A turncated cepstral distance is defined by [44, 59]

$$D_c^2(L) = \sum_{m=1}^{L} (a_m - a'_m) \tag{6.5}$$

6.1.3 Likelihood distortion measure

Consider the two spectra, magnitude-squared Fourier transforms, $S(\omega)$ and $S'(\omega)$ of the two signals s and s', where ω is the normalized frequency ranging from $-\phi$ to ϕ. The log spectral difference between the two spectra is defined by [44].

$$V(\omega) = logS(\omega) - logS'(\omega) \tag{6.6}$$

which is the basis for the distortion measure proposed by Itakura and saito in their formulation of linear prediction as an approximate maximum likelihood estimation The Itakura-saito distortion measure (D_{IS}) in the formulation of linear prediction as an approximate maximum likelihood estimation is defined as

$$D_{IS} = \int_{-\pi}^{\pi} [e^{V(\omega)} - V(\omega) - 1] \frac{d\omega}{2\pi} = \int_{\pi}^{-\pi} \frac{S(\omega)}{S'(\omega)} \frac{d\omega}{2\pi} - \log \frac{\sigma_\infty^2}{\sigma_\infty'^2} - 1 \tag{6.7}$$

where σ_∞^2 and $\sigma_\infty'^2$ are the one-step prediction error of $S(\omega)$ and $S'(\omega)$, respectively, and defined as

$$\sigma_\infty^2 \approx exp \left\{ \int_{-\pi}^{\pi} \log S(\omega) \frac{d\omega}{2\pi} \right\} \tag{6.8}$$

It was stated that Itakura-Saito distortion measure is connected with many statistical and information theories [44] including the likelihood ratio test, discrimination information and the Kullback-Leibler divergence.
The Itakura-Saito distortion measure can be used to illustrate the matching properties of linear prediction by replacing $S'(\omega)$ with a p^{th} order all-pole spectrum $\sigma^2/\left|A(e^{j\omega)}\right|^2$, leading to

$$D_{IS}\left(S, \frac{\sigma^2}{|A|^2}\right) = \frac{\boldsymbol{a}^T \boldsymbol{R}_p \boldsymbol{a}}{\sigma_\infty^2} - log\sigma_\infty^2 - 1. \tag{6.9}$$

where a^T is the transpose matrix, R_p is the autocorrelation matrix and a is the LPC coefficient. Based on the above information, we can gain-independent distortion measure derived directly from the Itakura-Saito distortion measure. Traditionally it is called the likelihood distortion measure and defined as [44]

$$\begin{aligned} D_{LR}\left(\tfrac{1}{|A_p|^2}, \tfrac{1}{|A|^2}\right) &= D_{IS}\left(\tfrac{1}{|A_p|^2}, \tfrac{1}{|A|^2}\right) \\ &= \int_{-\pi}^{\pi} \frac{\left|A(e^{j\omega}\right|^2}{|A_p(e^{j\omega}|^2} \frac{d\omega}{2\pi} - 1 \\ &= \frac{\boldsymbol{a}^T \boldsymbol{R}_p \boldsymbol{a}}{\sigma_p^2} - 1. \end{aligned} \tag{6.10}$$

That is, when the distortion is small, the Itakura distortion measure is not very different from the likelihood ratio distortion measure. To illustrate the above discussion at this point, the LPC likelihood ratio distortion measure can be derived [62] and defined as follows

$$D_{LR} = \frac{\boldsymbol{a'}^T \boldsymbol{R}_p \boldsymbol{a'}}{\boldsymbol{a}^T \boldsymbol{R}_p \boldsymbol{a}} - 1 \tag{6.11}$$

where $\boldsymbol{R_p}$ is the autocorrelation matrix of sequence s associated with its LPC coefficient vector $\boldsymbol{a}$, and $\boldsymbol{a'}$ is the LPC coefficient vector of signal s'. Thus, the distortion measure between an unknown MS samples s_m and from a particular known class i can be determined using the minimum rule as follows:

$$D_{min}(X_m, c^i) = min_j \; D(X_m, c_j^i) \tag{6.12}$$

where D is the distortion measure, X_m is an LPC vector of s_m, c_j^i is the j cluster center of a particular class represented by centroids c^i is minimum.
Using this decision logic, the unknown signal s_m is assigned to class i if the minimum distortion measure of its LPC vector x_m and the corresponding cluster center for LPC features c_i is minimum, that is

$$s_m \rightarrow i^*, i^* = arg\ min_i\ D_{min}(x_m, c^i) \tag{6.13}$$

6.2 Accuracy estimation

Basically there are two reasons for wanting to know generalization rate of classifier on a given problem. One is to see if the classifier perform well enough to be useful; another is to compare its performance with that of a competing design. Estimating the final generalization performance invariably requires making assumptions about the classifier or the problem or both, and fail if the assumptions are not valid. We should stress, then, that all the following methods are heuristic. Occasionally our assumptions are explicit, but more often than not they are implicit and difficult to identify or relate to the final estimation (as empirical methods). One approach to estimating the generalization rate is to compute it from the assumed parametric model. In simple validation we randomly split the set of labeled training samples D into two parts: one is used as the traditional training set for adjusting model parameters in the classifier. The other set-validation is used to estimate the generalization error. Since our goal is to low generalization error, we train th classifier until we reach a maximum of this validation error. It is essential that validation (or the test) set into include points used for training parameters in the classifier-a methodological error known as "testing on the training set" [26, 29]. There are different types of validation techniques available in literature, like Bootstrap, Jackknife and crossvalidation. But we restrict ourself to leave-one-out cross-validation and discussed in detail about its own benefits and drawbacks in the following subsection.

6.2.1 Cross-validation

Cross-validation (CV) is an empirical approach that tests the classifier accuracy experimentally. Once we train the classifier using cross-validation, the validation error gives an estimate of the accuracy of the final classifier on the unknown set [22]. Let V be the space of an unlabeled instances (eg: features) and Y the set of possible labels (centroids). Let $\chi = V \times Y$ be the space of labelled instances and $D = \{a_1, a_2, ..., a_n\}$

be a feature extracted MS dataset (possibly multidimensional) consisting of n labelled instances, where $a_i = (v_i \in V, y_i \in Y)$. A classifier C maps an unlabelled instance $v \in V$ to a label $y \in Y$ and an inducer τ maps a given dataset D into a classifier C (FCM or KFCM). This notation $\tau(D, v)$ will denote the label assigned to an unlabelled instance v by the classifier built by inducer τ on dataset D, i.e., $\tau(D, v) = (\tau(D))(v)$ [40].

Leave-one-out cross validation(LOOCV)

LOOCV involves using a single observation from the original sample as the validation data, and the remaining as the training data. It can be easily confused with Jackknife, because both involve omitting each training case in turn and retraining the network on the remaining subset. But cross-validation is used to estimate generalization error, while the Jackknife is used to estimate the bias of a statistic. In the Jackknife, you compute some statistic of interest in each subset of the data. The average of these subset statistics is compared with the corresponding statistic computed from the entire sample in order to estimate the bias of the latter. You can also get a Jackknife estimate of the standard error of a statistic. Jackknifing can be used to estimate the bias of the training error and hence to estimate the generalization error, but this process is more complicated than leave-one-out cross-validation [22]. When using the leave-out-one method, the inducer is trained and tested k times, each time $t \in \{1, 2, ..., k\}$, it is trained on $\frac{D}{D_t}$ and tested on D_t. The cross-validation estimate of accuracy is the overall number of correct classification, divided by the number of instances in the dataset D. Formally, let D_i be the test set that includes feature vector $x_i = (v_i, y_i)$, where v_i is an unlabeled instance (eg: feature) and y_i is a labelled instance (centroid), then the LOOCV estimation of accuracy (acc) is defined as [40]

$$acc_{cv} = \frac{1}{n} \sum_{(v_i, y_i) \in D} \delta\left(\tau\left(D/D_i, v_i\right), y_i\right) \tag{6.14}$$

The form of the algorithm is as follows:

For i=1 to k (where k is the number of feature vectors)

1. Let (v_i, y_i) be the i^{th} record
2. Temporarily remove the i^{th} feature vector from the training set.
3. Train the learning algorithm on the remaining $k - 1$ feature vectors.

4. Test the removed data point and note the error.

Leave-one-out cross-validation is useful because it does not waste data and works well for estimating generalization error for continuous error functions such as the mean squared error. When training, all but one of the points are used, so the resulting regression or classification rules are essentially the same as if they had been trained on all the data points. The main drawback to the leave-out-on method is that it is expensive-the computation must be repeated as many times as there are training set data points. Repeated cross-validation runs can be used to estimate the average classification accuracy of the given classification methods. In order to evaluate the performance of our clustering algorithms, we adopt LOOCV and the results are discussed in chapter 6 [26, 29, 40].

Chapter 7

Experiments on PC-H4 Dataset

We conducted our experiments with prostate cancer dataset (PC-H4), which has been previously used in [20, 21, 45, 58]. The detailed description of these datasets and the experimental setup are as follows.

7.1 Overview of datasets

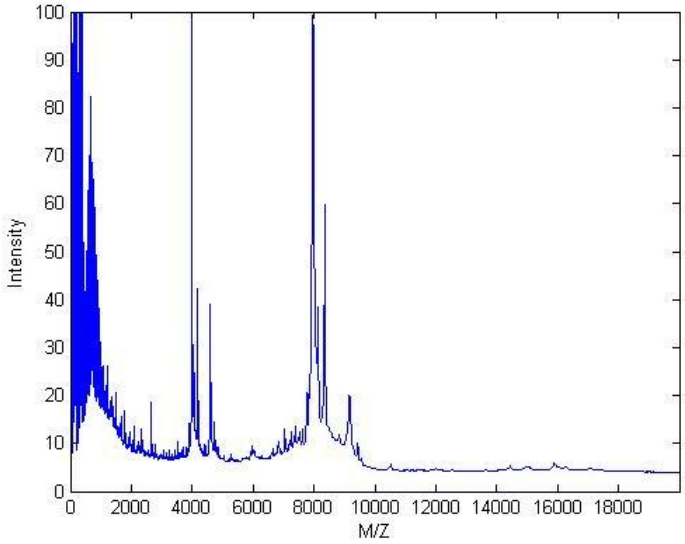
7.1.1 Prostate Cancer:

Cancer is a major public health concern in the world. Currently, the best way of reducing the morality of cancer is to detect and treat in the earliest stages. It is self-evident that the best way to cure cancer is to detect it before it has metastasized. Prostate cancer is now the most commonly diagnosed cancer in men and second leading cause of male cancer deaths, which is detected by measuring the concentration of the prostate specific antigen (PSA). Screening of prostate cancer detects the majority of prostate cancer patients; however severely hampered by a lack of selectivity. Among the men who are screened, 20%-26% will have an abnormal serum PSA and are likely to receive a recommendation for a prostate biopsy [25]. To avoid the unnecessary biopsies, efforts have focused on characterizing patient groups with an abnormal PSA who have a low likelihood of a positive biopsy. Proteomic technologies like mass spectrometry and micro array have been emerging to bring some hope for discovering biomarkers and building diagnosis models. Unfortunately, past biomarker discovery efforts have centered on laborious approaches looking for the elusive single over expressed protein in blood. Since there are tens to perhaps hundreds of thousands of intact, modified and cleaved protein isoforms in the human serum proteome, most of them has not be elucidated, therefore finding biomarkers is like searching for a needle in a haystack, requiring the separation and identification of each protein biomarker. The very small number of newly approved biomarkers is

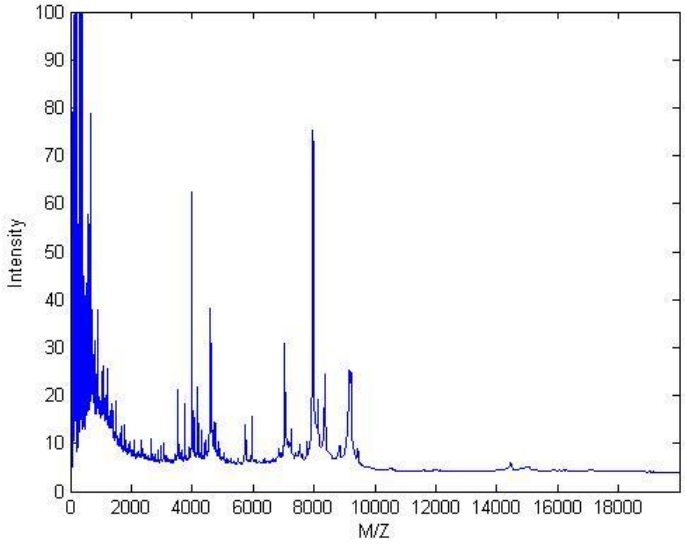
an unfortunate reflection of the inability and failure of hypothesis-driven and low-throughput approaches to deliver clinically useful biomarkers. A major source of this problem is due to our lack of basic knowledge about the proteomic components of serum and plasma [56]. Clinical applications will be eventually applied to a human population, not only in their respective proteomics, but also underlying disease process itself. Thus, it seems reasonable that the presence of cancer, with high sensitivity and selectivity, will be detected by multiplexed panels of clinical tests that measure modified and clipped/cleaved host and tumor-derived proteins, produced as a consequence of aberrant cellular function and cellular interactions [20, 63].
In recent years, the use of mass spectrometry for the identification of biomarkers seems to be giving promising results, reported that positive predictive value (PPV) is 90% for mass spectrometry. Thus, the patterns obtained from mass spectrometry can be immediately validated on blinded machine learning study sets. The National Cancer Institute-Food and Drug Adminstration (NCI-FDA) clinical proteomics program was formed to develop and apply novel technology to improve our ability to understand the biology of cancer. Applying this knowledge in practice, we hope to detect and identify molecular events that may be targets for prevention and treatment of cancer. Genomics and proteomics advances will help to guide our judgement with regard to the best treatment for each individual patients [14, 20].

7.1.2 Dataset description

The mass spectra (MS) profile consisting of 15,156 features was downloaded from http://www.home.ccr.cancer.gov/ncidfaprotoemics/ppatterns.asp and the detailed description of the dataset can be found in [58]. The dataset contains 322 total samples collected to investigate the biomarkers presence for prostate cancer. Out of these 322 samples: 190 samples were diagnosed with benign prostate hyperplasia with PSA levels greater than 4, 63 samples diagnosed with no evidence of disease and PSA level less than 1, 26 samples diagnosed with prostate cancer with PSA levels greater than 10. This set of data was collected using H4 protein chip, and a Ciphergen PBS1 SELDI-TOF mass spectrometer. The chip was prepared by hand using the recommended protocol and the spectra were exported with the baseline subtracted.



(a) prostate control



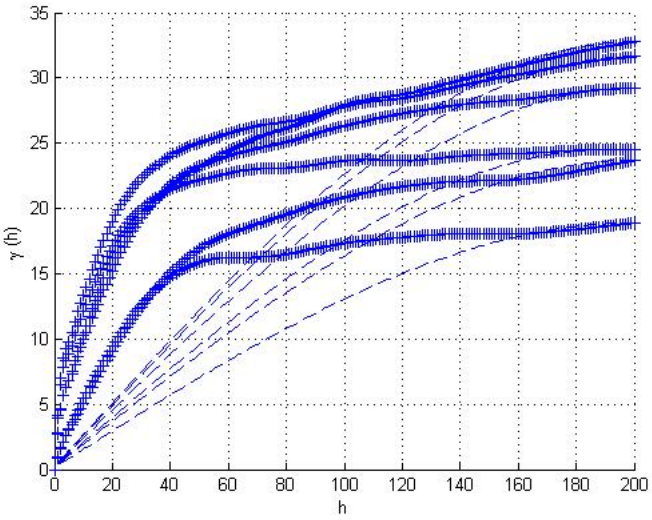
(b) prostate cancer

Figure 7.1: *Example of mass spectrum in which the relative intensity is plotted against mass-to-charge ratio(m/z). The data in this example are from the FDA-NCI Clinical Proteomics Program Databank. Every point of the mass-spectra is a candidate feature and usually the spectra of a cancer patient differs from that of a healthy person.*

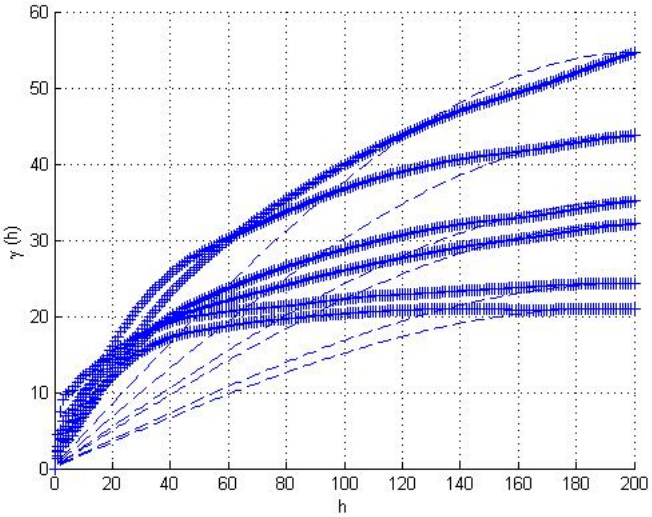
7.2 Experiment setup

Linear predictive coding (LPC) is applied to extract the features from mass spectrometry dataset as LPC coefficient. The number of poles p for the LPC is estimated using experimental semi-variogram (see chapter 4 for more details) from the given mass spectra, which revealed the number of poles lies between 50-70 (see

Figure 7.2) for LPC analysis. After constructing experimental semi-variogram, we build a model variogram using spherical model to fit the experimental variogram, which is shown in Figure 7.2. The dotted curves are constructed using spherical semi-variogram, whereas non-smooth curves are constructed using experimental semi-variogram. Based on the different pole values, features have been extracted from the given mass spectrometry dataset and are considered as the feature vectors for training and testing the proposed clustering algorithms.



(a) variograms of control



(b) variograms of cancer

Figure 7.2: *Experimental and spherical semi-variogram representation of SELDI-MS samples*

FCM and KFCM are the two proposed clustering algorithms used to extract the prototypes from the training set for effective classification accuracy. As we know

fuzzy clustering is greatly influenced by cluster centers c and the exponent value m, so it is a key issue for the users to properly evaluate the clustering parameters for fuzzy clustering algorithms.

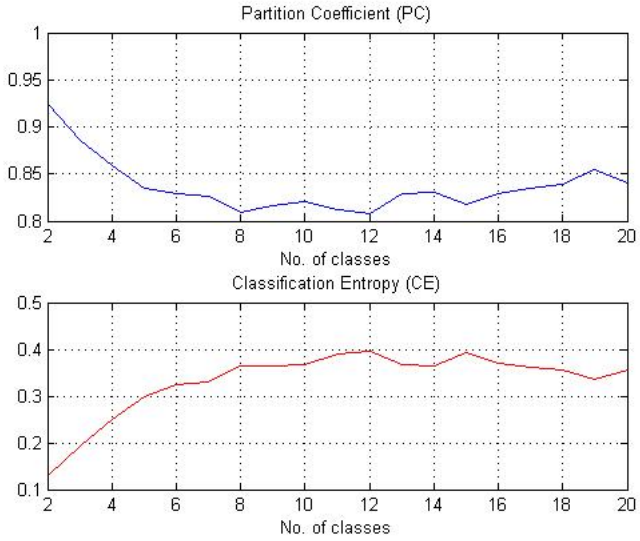


Figure 7.3: *Graphical representation of PC and CE*

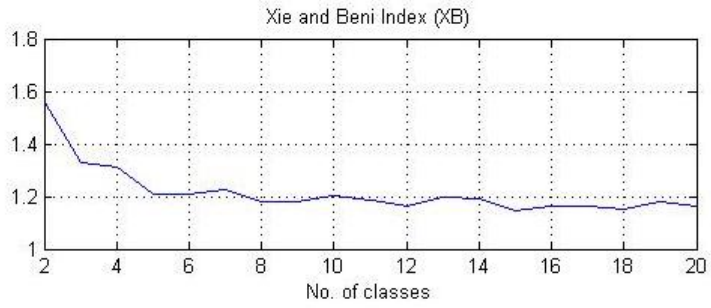


Figure 7.4: *Graphical representation of XB*

7.2.1 Parameters of study

Initially, we set the fuzzifier m in the algorithm to 2, the test for convergence in the basic FCM algorithm was performed using $\epsilon = 0.001$, and the distance function $\|*\|$ was defined as Euclidean distance. For the determination of the number of clusters,

the validity indices V_{PC}, V_{CE}, V_{XB} were used (for more details see chapter 4), which shows the cluster centers lies between $8 - 20$ and the graphical representation is shown in Figure 7.3 and 7.4. The maximum number of iteration and minimum amount of improvement was set to 1000 and 10^{-5} (the stopping criterion of the iteration). In the current implementation of KFCM, only gaussian kernel function is adopted by replacing the Euclidean distance (more details refer chapter 4) and the kernel parameter σ was set to 0.25 (Gaussian RBF kernel width), based on the variance obtained from the feature extracted datasets. For every method and every possible combination of parameters, we compute the classification rates and then choose the best parameters results based on selectivity and sensitivity. This process can be repeated until all the training samples have been counted, which is carried out using leave-one-out crossvalidation method (LOOCV). We estimated the performance accuracy of clustering algorithms using statistical methods.

7.2.2 Results

After initializing these parameters for the proposed classification methods, we carried out the experiments with different distortion measures such as cepstrum distortion measure and likelihood distortion measure for computing the dissimilarities between the given MS dataset. We calculated sensitivity (Sen) and selectivity (Sel) using all possible parameters.
Sensitivity is the percentage of diseased samples that are correctly classified;

$$Sensitivity = \frac{TP}{TP + FN} \tag{7.1}$$

and selectivity is the percentage of the healthy samples that are correctly classified;

$$Selectivity = \frac{TN}{TN + FP} \tag{7.2}$$

where true positive (TP) denote the correct classifications of positive samples; true negative (TN) are the correct classification of negative samples; false positive (FP) represent the incorrect classifications of negative samples into the positive class; and false negative (FN) are the positive samples incorrectly classified into the negative class. This process was repeated until each observation is used once as the validation data and the results are averaged, which is shown in the tables below:

Table 7.1 and Table 7.2 shows the classification accuracy of FCM and KFCM, when we used ceptrum and likelihood distortion measure to calculate the dissimilarties between the signals (MS data).

From Table 7.1 and Table 7.2, we can see the classification accuracy of the pro-

Case	Data dimension	FCM		KFCM	
		Selectivity%	Sensitivity%	Selectivity%	Sensitivity%
1	50	85.73	82.67	92.10	94.27
2	54	85.73	83.25	91.34	93.70
3	58	82.65	88.92	91.34	90.52
4	62	82.65	88.92	95.26	89.94
5	66	82.65	90.52	96.87	88.92
6	70	81.20	92.10	95.26	89.94

Table 7.1: Classification accuracy with $m = 2$ using cepstrum distortion measure

Case	Data dimension	FCM		KFCM	
		Selectivity%	Sensitivity%	Selectivity%	Sensitivity%
1	50	73.93	92.10	87.00	93.73
2	54	76.86	90.52	85.54	95.26
3	58	78.37	88.92	87.00	95.26
4	62	78.37	87.38	82.63	95.26
5	66	76.86	88.92	85.54	93.72
6	70	79.73	87.37	84.17	96.87

Table 7.2: Classification accuracy with $m = 2$ using likelihood distortion measure

posed computational models obtained using LOOCV, shows that KFCM has a best performance than FCM. FCM with membership value $m = 2$ failed to extract useful information to classify healthy and control from the given mass spectrometry dataset. Therefore, similar to the work of Dembele and Kastner (2003), we decided to select the exponent value for FCM algorithm. As a first step towards the evaluation, we first attempted to estimate upper bound value for $m(m_{ub})$. For our feature extracted mass spectrometry dataset, we varied m and determined the cv of Y_m. In each case, we observed that the values close to $\frac{1}{c}$, where c is the cluster center, gave a coefficient of variation (cv) of Y_m close to 0.05p, p being the data dimension, but there is no theoretical justification for this observation. Initially, we set $m = 2$ and computed $cv\{Y_2\}$. This value allowed us to decide the direction of search: in]1,2[if $cv\{Y_2\} < 0.05p$, in]2,∞[if $cv\{Y_2\} > 0.05p$ and $m_{ub} = 2$ if $cv\{Y_2\} \approx 0.05p$. If $m_{ub} \neq 2$, we performed successive choices of m in the correct direction and computed $cv\{Y_m\}$. Therefore, we decided to choose m lower or equal to 2, to get high membership values for data points related to clusters. We choose $m = 1 + m_0$, where $m_0 = 1$ if $m_ub \geq 10$ and $m_0 = \frac{m_{ub}}{10}$ if $m_{ub} < 10$. This choice leads to $m = 2$ when $m_{ub} > 10$ and to $m < 2$ when $m_{ub} < 10$. Thus for the given prostate canacer

dataset, we obtained the membership value ($m = 1.12$) and for more details refer to [19].

Case	Data dimension	FCM		KFCM	
		Selectivity%	Sensitivity%	Selectivity%	Sensitivity%
1	50	93.74	97.13	92.10	95.72
2	54	90.52	95.72	89.92	94.28
3	58	92.10	98.65	92.10	100
4	62	90.52	95.72	90.52	95.72
5	66	93.74	94.28	93.74	94.28
6	70	95.29	97.13	90.52	94.28

Table 7.3: Classification accuracy with $m = 1.12$ using cepstrum distortion measure

We followed the same principle for parameter initialization, as we discussed early in this chapter and the experiment is carried out once again by changing the exponent value for clustering algorithms. Table 7.3 and Table 7.4 shows the classification accuracy obtained with $m = 1.12$, using different distortion measures.

Case	Data dimension	FCM		KFCM	
		Sensitivity%	Selectivity%	Sensitivity%	Selectivity%
1	50	91.34	88.91	90.56	91.37
2	54	92.83	90.56	94.28	85.78
3	58	94.28	90.56	92.83	92.10
4	62	92.83	92.10	94.28	90.56
5	66	91.34	87.32	91.34	88.91
6	70	92.83	88.91	94.28	90.56

Table 7.4: Classification accuracy with $m = 1.12$ using likelihood distortion measure

7.2.3 Comparison

In this section, first we compared our results between different distortion measures and evaluated the classification accuracy with some popular machine learning techniques.

Table 7.5 shows the comparison of results obtained using two different distortion measures, we show here the best results obtained using different parameters on FCM and KFCM. Table 7.5, shows the effectiveness of the proposed distortion measures. The model is both physically and mathematically tractable. In speech recognition [44], as well as this study, the performance of LPC cepstral distortion measure appears to perform better than that of the LPC likelihood distrotion measure. Therefore, classification accuracy of the proposed methods are evaluated using the results obtained by cepstrum distortion measures.

Case	Classification methods	Exponent value	Cepstrum		Likelihood	
			Sel%	Sen%	Sel%	Sen%
1	FCM	$m = 2$	85.73	92.10	79.73	92.10
		$m = 1.12$	95.29	98.65	92.10	94.28
2	KFCM	$m = 2$	96.87	94.27	96.87	87.00
		$m = 1.12$	93.74	100	92.10	94.28

Table 7.5: Comparative study of cepstrum and likelihood distortion measure

Table 7.6 shows the comparative study of proposed methods with some sphosticated methods. In [45], the researchers increased the performance of PC-H4 dataset by the use of boosting. For feature selection, they used nearest shrunken centroid, filter-based feature selection and wrapper-based feature selection to extract useful information from the given MS dataset and they analysed the feature extracted dataset using several classifiers. The researchers used stratified three-fold cross-validation procedure to train the classifiers, whereby each dataset split into three subsets of equal size. They reported that boosted FE obtained good classification accuracy on PC-H4 with a selectivity of 100% and sensitivity of 81.2% and this is the highest reported accuracy of the dataset in [45]. The results in Table 7.6 shows that kernel based fuzzy clustering algorithm outperforms the Boosted (boosted nearest centroid algorithm) and Boosted FE (boosting based feature extraction), which was shown to be a superior method. In addition, Table 7.6, also shows some results obtained using various other methods [21, 45, **?**].

In [21], the researchers used LDA as classifier to classify the feature extracted data obtained using discrete wavelet transforms (DWT) method and they reported LDA model gave the better classification rate (for cancerous patients) of 89.47% for control, 90.47% for cancer, when compared to Treeboost and Random Forests. But our clustering algorithms obtained the classification rate of 100% for cancer and 93.7% for control, which outperformed the classification rate of LDA.

In [58], the researchers used self-organizing map to classify feature extracted PC-H4 dataset obtained using genetic algorithms. They performed cross-validation and obtained the selectivity of 95% and a sensitivity of 71%, which is compared with our clustering algorithms and the results are shown in Table 7.6.

In [50], MIT correlation method was used as the feature selection technique to extract the features from the mass spectrometry dataset and they classified healthy men from those infected using support vector machine (SVM). This approach indicates that SVM with polynomial kernel worked well with prostate cancer data, when compared to linear and radial kernel functions, achieving the selectivity of 89.0% and the sensitivity of 79.0%, for an overall classification accuracy of 81.0%, which is been referred as SVM1 in Table 7.6.

In [20], the researchers once again investigated linear SVM as a classifier on PC-H4. They performed feature selection using peak detection method and trained the linear SVM with five-fold cross-validation method and obtained the classification accuracy of 85.3±1.9. In our study of unsupervised fuzzy clustering algorithms on prostate cancer dataset, we used radial kernel functions by replacing the original Euclidean distance, which seems to give more promising results than linear kernel and polynomial kernel and the results are shown in Table 7.6.

The above discussion shows that we compared results with several well-known classification methods to distinguish prostate cancer patients from normal individuals based on MS data obtained on serum samples. But overall, we found that the kernel based fuzzy c-means approach leads to lower misclassification rate as well as to a stable assessment of classification errors, which is stated in Table 7.6. Although many methods have been compared in this research, there are also some additional methods, e.g. hidden markov models, that we have not yet compared. This is an ongoing endeavour, and we are in the process of evaluating those other methods for future work.

Case	Classification methods	Classification accuracy	
		Selectivity%	Sensitivity%
1	KFCM	93.74	100
2	FCM	95.29	98.65
3	SVM	85.30	85.30
4	SVM1	89.00	79.00
5	LDA/PCA	71.00	62.30
6	LDA	89.47	90.47
7	Random forest	94.73	76.12
8	Tree boost	100	68.75
9	PCA	54.00	49.30
10	SFS	92.90	72.50
11	SBS	80.60	65.20
12	Boosted	88.10	73.90
13	Boosted FE	100	81.20
14	SOM	95.00	71.00

Table 7.6: Comparative study with other techniques

Chapter 8

Conclusions and Future work

8.1 Conclusions

The proteomics research field is progressing through the development of novel-technology, with the hope of discovering biomarkers that can be used to diagnose diseases, predict susceptibility, and monitor progression. A revolutionary approach in proteomic patterns analysis can offer tremendous potential for the early detection of complex human diseases like prostate cancer. Proteomic pattern analysis relies on the patterns of proteins observed and doesn't rely on the identification of a tractable biomarkers. Hundreds of clinical samples are generated each day using some popular proteomic techniques such as 2D-gel electrophoresis, mass spectrometry, and micro arrays. Such large high-throughput collections of data require powerful tools to assist data analysis. Machine learning has increasingly gained attention in bioinformatics research and its the subfield of artificial intelligence which focuses on methods to construct computer programs that learn from the experience with respect to some class of tasks and a performance measure. Because of the multi-factorial nature of MS data, it is clear that computational methods are needed to analyse the given datasets which will help in detecting the disease. In my research, I applied unsupervised clustering algorithms to serum proteomic pattern analysis for cancer early detection.

Mass spectrometry data are characterized by high dimensionality, high levels of redundancy, information irrelevant to particular disease data and measurement noise. Therefore, feature extraction techniques are crucial to extract valuable information for learning high-accuracy classifiers and gaining the full potential of mass spectrometry based disease diagnosis. Given the features obtained from the mass spectrometry datasets, using the principle of LPC, we then applied FCM and KFCM algorithms on feature extracted dataset to discriminate healthy from cancer. For every possible combination of parameters, we compute the classification rates and

then choose the best parameters results based on selectivity and sensitivity using leave-out-one crossvalidation method. In this thesis, kernel methods for clustering have been analysed by paying attention to fuzzy kernel methods and even we explored the notion of data clustering in a kernel defined feature space.

The classification results obtained using kernel (RBF kernel) based fuzzy clustering algorithms shows that KFCM is the best among the compared classifiers including the most popular applied methods like support vector machines, PCA/LDA and random forests. We also demonstrated that kernel based clustering algorithms with exponent value $m = 1.2$, can accurately classify the cancer samples based on serum proteomic patterns. Thus, we believe these approaches have significant potential in proteomic pattern analysis for early cancer detection and to identify biomarkers.

Recent studies confirm that there is no universal pattern recognition and classification model to predict molecular profiles across different datasets and medical domains. Many classification and knowledge discovery problems may require the combination of multiple techniques not only to improve the accuracy and efficiency of the analysis tasks, but also to support evaluation procedures. Therefore we can hypothesize that improvement in these components will yield the greatest increase in system reliability and that the approaches most likely to achieve those improvements will be based on explicit models of the data generation.

8.2 Directions of future work

- **Analysis of MS data:** In our research, we have implemented unsupervised kernel based clustering algorithm to analysis prostate cancer dataset (PC-H4). Therefore, it would be worth investigating this approach with other MS datasets.

- **Feature extraction:** As shown in chapter 3, we implemented LPC as a feature extraction technique to process the whole MS data, instead we might concentrate on selecting the desired peak in our future research. Efforts to identify the proteins corresponding to relevant features should follow feature selection and classification studies.

- **Parameter study on FCM:** According to my knowledge from review, there is no standard theoretical justification or empirical evidence for the choice of the exponent value m for fuzzy clustering algorithms. So future research will investigate on developing the novel technique, which could determine the optimal m value for FCM.

- **Kernel methods:** In chapter 4, we mainly discussed about the implementation of RBF kernel method into fuzzy clustering algorithms. Therefore, our future research will focus on analysing different kernel methods with FCM. In addition, we might do a comparative study to evaluate the performance of kernel methods.

- **Fuzzy hidden Markov model (FHMM):** In addition to the above specified future work, we might extend hidden Markov models to include the fuzzy-set modelling of the model parameters for robust classification.

Bibliography

[1] Aebersold, R., and Goodlett, D. Mass spectrometry in proteomics. *Chemical Reviews 101*, 2 (2001), 269–296.

[2] Aizerman, A., Braverman, E. M., and Rozoner, L. I. Theoretical foundations of the potential function method in pattern recognition learning. *Automation and Remote Control 25* (1964), 821–837.

[3] Anderle, M., Roy, S., Lin, H., Becker, C., and Joho, K. Quantifying reproducibility for differential proteomics: noise analysis for protein liquid chromatography-mass spectrometry of human serum. *Bioinformatics 20*, 18 (2004), 3575–3582.

[4] Baker, S., Kramer, B., and Srivastava, S. Markers for early detection of cancer: Statistical guidelines for nested case-control studies. *BMC Medical Research Methodology 2*, 4 (2002), 1–8.

[5] Ball, G., Mian, S., Holding, F., Allibone, R. O., Lowe, J., Ali, S., Li, G., McCardle, S., Ellis, I. O., Creaser, C., and Rees, R. C. An integrated approach utilizing artificial neural networks and SELDI mass spectrometry for the classification of human tumours and rapid identification of potential biomarkers . *Bioinformatics 18*, 3 (2002), 395–404.

[6] Bezdek, J. *Pattern recognition with fuzzy objective function algorithms.* Plenum, New York, 1981.

[7] Bouguessa, M., Wang, S., and Sun, H. An objective approach to cluster validation. *Pattern Recogn. Lett. 27*, 13 (2006), 1419–1430.

[8] Breiman, L. Bagging predictors. *Mach. Learn. 24*, 2 (1996), 123–140.

[9] C, D. J. A fuzzy relative to the isodata process and its use in detecting compact, well-separated clusters. *J.cybernet 3* (1974), 32–57.

[10] Chace, D. H., Petricoin, E. F., and Liotta, L. A. Mass spectrometry-based diagnostics: The upcoming revolution in disease detection has already arrived. *Clin.Chem 49*, 7 (2003), 1227–1229.

[11] Chi, Z., Yan, H., and Pham, T. *Fuzzy algorithms: with applications to image processing and pattern recognition.* Word Scientific, New Jersey, 1996.

[12] Choe, H., and , Jordan, J. On the optimal choice of parameters in a fuzzy c-means algorithm. In *IEEE International conference on Fuzzy systems* (1992), pp. 349–354.

[13] Christopher, J. C. B. A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery 2* (1998), 121–167.

[14] C.M., M. Genomics and proteomics: application of novel technology to early detection and prevention of cancer. *Cancer Detection and Prevention 26* (October 2002), 249–255(7).

[15] Dao-Qiang, Z., and Song-Can, C. Clustering incomplete data using kernel-based fuzzy c-means algorithm. *Neural Process. Lett. 18*, 3 (2003), 155–162.

[16] Datta, S., and DePadilla, L. M. Feature selection and machine learning with mass spectrometry data for distinguishing cancer and non-cancer samples. *Statistical methodology 3* (2006), 79–92.

[17] Davis, J. C. *Statistics and Data Analysis in geology.* Wiley, Singapore, 1986.

[18] Deer, P. J., and Eklund, P. A study of parameter values for a mahalanobis distance fuzzy classifier. *Fuzzy Sets Syst. 137*, 2 (2003), 191–213.

[19] Dembele, D., and Kastner, P. Fuzzy c-means method for clustering microarray data. *Bioinformatics 19*, 8 (2003), 973–980.

[20] Deng, X., Geng, H., Bastola, D. R., and Ali, H. H. Link test-a statistical method for finding prostate cancer biomarkers. *Comput. Biol. Chem. 30*, 6 (2006), 425–433.

[21] Donald, D., Hancock, T., Coomans, D., and Everingham, Y. Bagged super wavelets reduction for boosted prostate cancer classification of seldi-tof mass spectral serum profiles. *Chemometrics and intellegent laboratory systems 82* (2006), 2–7.

[22] Duda, O. R., E.Peter, H., and G.David, S. *Pattern recognition.* Wiley, 2001.

[23] Filippone, M., Camastra, F., Masulli, F., and Rovetta, S. A survey of kernel and spectral methods for clustering. *Pattern Recogn. 41*, 1 (2008), 176–190.

[24] Fung, G. M., and Mangasarian, O. L. A feature selection newton method for support vector machine classification. *Comput. Optim. Appl. 28*, 2 (2004), 185–202.

[25] Garzotto, M., Beer, T. M., Hudson, R. G., Peters, L., Hsieh, Y.-C., Barrera, E., Klein, T., and Mori, M. Improved Detection of Prostate Cancer Using Classification and Regression Tree Analysis. *J Clin Oncol 23*, 19 (2005), 4322–4329.

[26] Gordon, A. *Classification*. Chapman / Hall/ CRC, New York, 1999.

[27] Guzzi, P. H., Mazza, T., and Tradigo, G. Preprocessing of mass spectrometry proteomics data on the grid. In *CBMS '05: Proceedings of the 18th IEEE Symposium on Computer-Based Medical Systems* (Washington, DC, USA, 2005), IEEE Computer Society, pp. 549–554.

[28] Halkidi, M., Batistakis, Y., and Vazirgiannis, M. Clustering algorithms and validity measures, 2001.

[29] Hastie, T., Tibshirani, R., and Friedman, J. *The elements of statistical learning*. Springer–verlag, New York, 1997.

[30] Hilari, M., Kalousis, A., Muller, M., and Pellegrini, C. Machine learning approaches to lung cancer prediction from mass spectra. *Proteomics*.

[31] Hilari, M., Kalousis, A., Muller, M., and Pellegrini, C. Processing and classification of protein mass spectra. *Mass Spectrometry Reviews 25*, 3 (2006), 409–449.

[32] Hoffman, E.-D., and Stroobant, V. *Mass spectrometry: Principles and Applications*. John Wiley and Sons, New York, 2003.

[33] Hood, L., Heath, J. R., Phelps, M. E., and Lin, B. Systems Biology and New Technologies Enable Predictive and Preventative Medicine. *Science 306*, 5696 (2004), 640–643.

[34] Isobel, C. *Practical Geostatistics*. Applied science, London, 1979.

[35] Jain, A., Duin, R., and Mao, J. Statistical pattern recognition: a review. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 22*, 1 (Jan 2000), 4–37.

[36] Jain, A., Duin, R., and Mao, J. Statistical pattern recognition: a review. *IEEE Transactions on Pattern Analysis and Machine Intelligence 22*, 1 (Jan 2000), 4–37.

[37] Jain, K. J., and Richard, D. C. *Algorithms for clustering data*. Prentice Hall, New Jerssy, 1988.

[38] Jeffries, N. Performance of a genetic algorithm for mass spectrometry proteomics. *BMC Bioinformatics 5*, 1 (2004), 180.

[39] Jeffries, N. Algorithms for alignment of mass spectrometry proteomic data. *Bioinformatics 21*, 14 (2005), 3066–3073.

[40] Kohavi, R. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *The International Joint Conference on Artificial Intelligence (IJCAI)* (1995), pp. 1–77.

[41] Kohonen, T. The self-organizing map. In *Proceedings of the IEEE* (1990), vol. 78, pp. 1464–1480.

[42] Kohonen, T. *Self-organizing maps*. Springer, New York, 2001.

[43] Lane, C. S. Mass spectrometry-based proteomics in the life sciences. *Cellular and Molecular Life Sciences (CMLS) 62* (2005), 848–869.

[44] Lawerence, R., and Juang, B.-H. *Fundamentals of speech recognition*. Prentice Hall PTR, New Jerssy, 1943.

[45] Levner, I. Feature selection and nearest centroid classification for protein mass spectrometry. *BMC Bioinformatics 6* (2005), 1–14.

[46] Li, J., Liu, H., Ng, S.-K., and Wong, L. Discovery of significant rules for classifying cancer diagnosis data. *Bioinformatics 19*, 2 (2003), 93–102.

[47] Li, L., Tang, H., Wu, Z., Gong, J., Gruidl, M., Zou, J., Tockman, M., and Clark, R. A. Data mining techniques for cancer detection using serum proteomic profiling. *Artificial intelligence in medicine 32* (2004), 71–83.

[48] Li, L., Umbach, D. M., Terry, P., and Taylor, J. A. Application of the GA/KNN method to SELDI proteomics data. *Bioinformatics 20*, 10 (2004), 1638–1640.

[49] Li, X., Li, J., and Yao, X. A wavelet-based data pre-processing analysis approach in mass spectrometry. *Comput. Biol. Med. 37*, 4 (2007), 509–516.

[50] Liu, Y. Serum proteomic pattern analysis for early cancer detection. *Technology in Cancer Research and Treatment 5*, 1 (2006), 61–66.

[51] Malmstrm, J., Malmstrm, L., and Marko-Varga, G. Proteomics: A new research area for the biomedical field. *J. of organ dysfunction 1* (2005), 83–94.

[52] McBratney, A., and Moore, A. Application of fuzzy sets to climate classification. *Agriculture and Forest Meterology 35* (1985), 165–185.

[53] Muller, K.-R., Mika, S., Ratsch, G., Tsuda, K., and Scholkopf, B. An introduction to kernel–based learning algorithms. *Neural Networks, IEEE Transactions on 12*, 2 (Mar 2001), 181–201.

[54] Okeke, F., and Karnieli, A. Linear mixture model approach for selecting fuzzy exponent value in fuzzy c-means algorithm. *Ecological Informatics* (2006), 117–124.

[55] Pal, N. R., and Bezdek, J. On cluster validity for the fuzzy c-means model. *IEEE Trans. on Fuzzy systems 3*, 3 (1995), 370–379.

[56] Petricoin, E., Wulfkuhle, J., Espina, V., and Liotta, L. Clinical proteomics: Revolutionizing disease detection and patient tailoring therapy. *Journal of Proteome Research 3*, 2 (2004), 209–217.

[57] Petricoin III, E., Ardekani, A., Hitt, B., Levine, P., Fusaro, V., Steinberg, S., Mills, G., Simone, C., Fishman, D., and Kohn, E. Use of proteomic patterns in serum to identify ovarian cancer. *The Lancet 359*, 9306 (2002), 572–577.

[58] PetricoinIII, E. F., Ornstein, D. K., Paweletz, C. P., Ardekani, A., Hackett, P. S., Hitt, B. A., Velassco, A., Trucco, C., Wiegand, L., Wood, K., Simone, C. B., Levine, P. J., Linehan, W. M., Emmert-Buck, M. R., Steinberg, S. M., Kohn, E. C., and Liotta, L. A. Serum proteomic patterns for detection of prostate cancer. *J. Natl. Cancer Inst. 94*, 20 (2002), 1576–1578.

[59] Pham, T. D. Spectral distortion measures for biological sequence comparisons and database searching. *Pattern Recogn. 40*, 2 (2007), 516–529.

[60] Pham, T. D., Chandramohan, V., Zhou, X., and Wong, S. T. C. Robust feature extraction and reduction of mass spectrometry data for cancer classification. In *ICDMW '06: Proceedings of the Sixth IEEE International Conference on Data Mining - Workshops* (Washington, DC, USA, 2006), IEEE Computer Society, pp. 202–206.

[61] Pham, T. D., and Wagner, M. A geostatistical model for linear prediction analysis of speech. *Pattern Recognition 31* (1998), 1981–1991.

[62] Pham, T. D., Wang, H., Zhou, X., Beck, D., Brandl, M., Hoehn, G., Azok, J., Brennan, M.-L., Hazen, S. L., Li, K., and Wong, S. T. C. Linear predictive coding and its decision logic for early prediction of major adverse cardiac events using mass spectrometry data. In *WISB '06: Proceedings of the 2006 workshop on Intelligent systems for bioinformatics* (Darlinghurst, Australia, Australia, 2006), Australian Computer Society, Inc., pp. 61–66.

[63] Posadas, E. M., Simpkins, F., Liotta, L. A., MacDonald, C., and Kohn, E. C. Proteomic analysis for the early detection and rational treatment of cancer–realistic hope? *Ann Oncol 16*, 1 (2005), 16–22.

[64] Qu, Y., Adam, B.-L., Thornquist, M., Potter, J. D., Thompson, M. L., Yasui, Y., Davis, J., Schellhammer, P. F., Cazares, L., Clements, M. A., Wright, George L., J., and Feng, Z. Data reduction using a discrete wavelet transform in discriminant analysis of very high dimensionality data. *Biometric 59* (2003), 143–151.

[65] Ramze, R. M., Lelieveldt, B., and Reiber, J. A new cluster validity index for the fuzzy c-mean. *Pattern Recognition Letters 19* (March 1998), 237–246.

[66] Saeys, Y., Inza, I., and Larranaga, P. A review of feature selection techniques in bioinformatics. *Bioinformatics* (2007), 1–10.

[67] Sajda, P. Machine learning for detection and diagnosis of disease. *Annual Review of Biomedical Engineering 8*, 1 (2006), 537–565.

[68] Schapire, R. A brief introduction to boosting. In *Proceedings of the sixteenth international joint conference on artificial intelligence* (1999), pp. 1401–1406.

[69] Schokopf, B., and Smola, J. *Learning with kernels*. MIT press, 2002.

[70] Schölkopf, B., Smola, A., and Müller, K.-R. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Comput. 10*, 5 (1998), 1299–1319.

[71] Shen, H., Yang, J., Wang, S., and Liu, X. Attribute weighted mercer kernel based fuzzy clustering algorithm for general non-spherical datasets. *Soft Comput. 10*, 11 (2006), 1061–1073.

[72] Shin, H., and Markey, M. K. A machine learning perspective on the development of clinical decision support systems utilizing mass spectra of blood samples. *J. of Biomedical Informatics 39*, 2 (2006), 227–248.

[73] Srinivas, P. R., Srivastava, S., Hanash, S., and Wright, George L., J. Proteomics in Early Detection of Cancer. *Clin Chem 47*, 10 (2001), 1901–1911.

[74] Sun, H., Wang, S., and Jiang, Q. Fcm-based model selection algorithms for determining the number of clusters. *Pattern recognition 37* (2004), 2027–2037.

[75] Titulaer, M. K., Siccama, I., Dekker, L. J., and Luider, T. A database application for pre-processing, storage and comparison of mass spectra derived from patients and controls. *BMC Bioinformatics 7*, 403 (2006).

[76] Wanger, M., Naik, D., and Pothen, A. Protocols for disease classification from mass spectrometry data. *Proteomics 3*, 9 (2003), 1692–1698.

[77] Wu, B., Abbott, T., Fishman, D., McMurray, W., Mor, G., Stone, K., Ward, D., Williams, K., and Zhao, H. Comparison of statistical methods for classification of ovarian cancer using mass spectrometry data. *Bioinformatics 19*, 13 (2003), 1636–1643.

[78] Wulfkuhle, J., Liotta, L., and Petricoin, E. Proteomic applications for the early detection of cancer. *Nature reviews 3* (2003), 267–275.

[79] Xie, X. L., and Beni, G. A validity measure for fuzzy clustering. *IEEE Trans. Pattern Anal. Mach. Intell. 13*, 8 (1991), 841–847.

[80] Yang, X., Cao, A., and Song, Q. A new cluster validity for data clustering. *Neural Process. Lett. 23*, 3 (2006), 325–344.

[81] Yasui, Y., Pepe, M., Thompson, M. L., Adam, B.-L., Wright, George L., J., Qu, Y., Potter, J. D., Winget, M., Thornquist, M., and Feng, Z. A data-analytic strategy for protein biomarker discovery: profiling of high-dimensional proteomic data for cancer detection. *Biostat 4*, 3 (2003), 449–463.

[82] Yu, J., Cheng, Q., and Huang, H. Analysis of the weighting exponent in fcm. *IEEE Transactions on Systems, Man and Cybernetics 34* (2004), 634–639.

[83] Zahid, N., Limouri, M., and Essaid, A. A new cluster-validity for fuzzy clustering. *Pattern recognition 32*, 7 (1999), 1089–1097.

[84] Zhang, D.-Q., and Chen, S.-C. A novel kernelized fuzzy c-means algorithm with application to medical image segmentation. *Artificial Intelligence in Medicine 32*, 1 (2004), 37–50.