GENOMIC SIGNAL PROCESSING FOR ENHANCED MICROARRAY DATA CLUSTERING

ALA M. H. SUNGOOR

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS OF KINGSTON UNIVERSITY LONDON
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

Faculty of Computing, Information Systems and Mathematics
Kingston University London

2009

THE FOLLOWING FIGURES HAVE BEEN EXCLUDED ON INSTRUCTION FROM THE UNIVERSITY

FIGURES 1.1a, 1.2, AND 1.3

Acknowledgement

First and foremost, I would like to express my sincere and deep gratitude to my supervisor Professor Robert S. H. Istepanian. I will always be grateful for his continuous advice and diligent supervision on this work.

This work would not have been possible without Dr. Jean-Christophe Nebel for having faith in my ideas and for allowing me to go where it was sometimes hard to follow.

I would also like to thank the Faculty of Computing, Information Systems and Mathematics, Kingston University London for providing me with this opportunity to fulfil this research, especially to Professor Timothy Ellis, Dean of the faculty, for his support.

I would also like to extent my gratitude and grateful to Ms. Bee Tang for her support. I have had the pleasure of being a member of MINT Centre and MOMED group, to all my especial thanks for the useful suggestions and help. Thanks also for the friendship and encouragement from Dr. Nada Philip.

Finally, I would like to thank all of those who have helped me and showed their support. They are always in my mind and with my best wishes.

Abstract

Genomic signal processing is a new area of research that combines genomics with digital signal processing methodologies for enhanced genetic data analysis. Microarray is a well known technology for the evaluation of thousands of gene expression profiles. By considering these profiles as digital signals, the power of DSP methods can be applied to produce robust and unsupervised clustering of microarray samples. This can be achieved by transferring expression profiles into spectral components which are interpreted as a measure of profile similarity.

This thesis introduces enhanced signal processing algorithms for robust clustering of microarray gene expression samples. The main aim of the research is to design and validate novel genomic signal processing methodologies for microarray data analysis based on different DSP methods. More specifically, clustering algorithms based on Linear prediction coding, Wavelet decomposition and Fractal dimension methods combined with Vector quantisation algorithm are applied and compared on a set of test microarray datasets. These techniques take as an input microarray gene expression samples and produce predictive coefficients arrays associated to the microarray data that are quantised in discrete levels, and consequently used for sample clustering.

A variety of standard microarray datasets are used in this work to validate the robustness of these methods compared to conventional methods. Two well known validation approaches, i.e. Silhouette and Davies Bouldin index methods, are applied to evaluate internally and externally the genomic signal processing clustering results.

In conclusion, thr results demonstrate that genomic signal processing based methods outperform traditional methods by providing more clustering accuracy. Moreover, the study shows that the local features of the gene expression signals are better clustered using wavelets compared to the other DSP methods.

Table of Contents

Abstract

List of Abbreviation

CHAPTER 1 *Introduction* 1

1.1- Motivation of the research 1

1.1.1- Microarray technologies 1

1.1.2- Types and functionalities of microarrays 3

1.1.3- Microarray data representation 9

1.2- Genomic Signal Processing for microarray clustering 10

1.3- Objective of the research and thesis contribution 12

1.4- Outline of the thesis 13

CHAPTER 2 *Microarray data analysis* 14

2.1- Review of microarray data analysis methods 15

2.2- Clustering analysis 22

2.2.1- Microarray data representation 23

2.2.2- Microarray data Proximity 24

2.2.3- Microarray clustering algorithms 30

2.2.3.1- Conventional microarray clustering methods 30

2.2.3.2- Comparison of clustering techniques 34

2.2.4- Microarray cluster validation measures 36

2.3- Spectral technique for microarray data analysis 38

2.4- Distortion measure based clustering 40

2.5- Distortion measure for VQ of microarray spectrum 41

2.6- Conclusion 45

CHAPTER 3 *Linear Predictive Coding for microarray clustering* 46

3.1- Characteristics of microarray gene samples 47

3.2- Background theory of LPC Analysis 49

3.3- Linear Predictor coefficients 50

3.3.1- Autocorrelation method 51

3.3.2- Covariance method 53

3.3.3- Improved LPC coefficients analysis 54

3.3.3.1- Line Spectral Frequency Transform 55

3.3.3.2- Vector Quantisation of LSF coefficients 57

3.4- Microarray LPC (miLPC) clustering analysis 58

3.5- Performance analysis of miLPC 65

3.5.1- miLPC evaluation criteria 65

3.5.2- Quantisation evaluation criteria 66

3.6- Conclusion 68

CHAPTER 4 *Wavelet for microarray clustering* 69

4.1- Properties of Wavelets 69

4.2- Application of DWD in gene expression 70

4.2.1- Wavelet background 71

4.2.2- Wavelet processing blocks 74

4.2.3- Vector Quantisation of DWD 75

4.3- Microarray DWD (miDWD) analysis 75

4.4- Performance analysis of miDWD 77

4.4.1- miDWD evaluation criteria 77

4.4.2- Quantisation evaluation criteria 79

4.5- Conclusion 81

CHAPTER 5 *Fractals for microarray clustering* 82

5.1- Application of FD in gene expression sample signal 82

5.2- Fractal dimension analysis methods 83

5.2.1- Self-similar and Fractal developments 84

5.2.2- Self-similar Fractal to Integrated dataset series 88

5.2.3- Fractal of multi-dimensional data space 89

5.2.4- Box-counting approach to Computing FD 90

5.3- miFD clustering analysis 91

5.4- Performance analysis of miFD 92

5.5- Conclusion 96

CHAPTER 6 *Performance analysis of GSP methods* 97

6.1- Microarray test data types 97

6.2- Preprocessing 101

6.3- Application of miLPC clustering 104

6.3.1- miLPC on Leukaemia dataset 104

6.3.2- Discussion 109

6.4- Application of miDWD clustering 110

6.4.1- miDWD on Leukaemia dataset 111

6.4.2- Discussion 115

6.5- Application of miFD clustering 116

6.5.1- miFD on Leukaemia dataset 116

6.5.2- Discussion 119

6.6- Cluster validation methods 120

6.6.1- Silhouette index 121

6.6.1.1-miLPC approach 121

6.6.1.2-miDWD approach 123

6.6.1.3-miFD approach 125

6.6.1.4-Discussion 127

6.6.2- Davies-Bouldin (DB) index 127

6.7- Conclusion 129

CHAPTER 7 *Conclusion and Future work* 130
7.1- Conclusions 130
7.2- Limitation and future works 131
7.2.1- Limitation of the study 131
7.2.2- Future work 134

References 135
Appendix A

List of Abbreviations

Abbreviations	**Description**
ANOVA	Analysis of Variance
AR	Autoregressive
BBS	Branch-and-Bound-Search
BN	Bayesian Network
cA(n)	Approximation coefficients (scaling coefficients)
cD(n)	Detail coefficients (wavelet coefficients)
cDNA	complementary DNA
CGH	Comparative genomic hybridization
DB	Davies-Bouldin index
db2	Daubechies wavelets type 2
DNA	Deoxyribonucleic acid
DSP	Digital Signal Processing
DWD	Discrete Wavelet Decomposition
EM	Expectation-Maximization
FD	Fractal Dimension
FDA	Fisher Discriminant Analysis
GEM	Gene Expression Matrix
GESF	Gene Expression Spectral Frequency
GSP	Genomic Signal Processing
HC	Hierarchical clustering
ICA	Independent Component Analysis
k-NN	k-nearest neighbours
KR	Kernel Regression
LDR	Local Dimensionality Reduction
LP	Linear prediction
LPC	Linear predictive coding
LR	Loess Regression
LSF	Line spectral frequency
LSP	Line spectral pair
Lv	Wavelet levels
MI	Mutual Information
miDWD	Microarray clustering based on
miFD	Microarray clustering based on
miLPC	Microarray clustering based on LPC
mRNA	Messenger RNA
MSE	Mean Square Error

NN	Nearest Neighbour
p	LPC order
PCA	Principal Component Analysis
PLR	Penalized Logistic Regression
PLS	Partial Least Squares
QDA	Quadratic Discriminant Analysis
R	Regression value
RNA	Ribonucleic acid
SBFS	Sequential Backward Floating Search
SFFS	Sequential Forward Floating Search
SNP	Single Nucleotide Polymorphism
SNR	Signal-to-Noise Ratio
SOM	Self-Organizing Map
SS	S-plines Smoothing
SVD	Singular Value Decomposition
SVM	Support Vector Machine
SVR	Support Vector Regression
SW	Silhouette width
TL	Threshold Logic
v_{gn}	Gene expression value in GEM (g=number of genes, n=Sample of expression condition
VQ	Vector quantisation
WS	Wavelets Smoothing
$\psi_{i,j}(t)$	Wavelet basis functions

CHAPTER 1

Introduction

Microarray technologies provide powerful tools for gene expression analysis that allows the study of thousands of DNA sequences simultaneously to extract information about specific gene activities. Microarrays have become essential tools for studying live biological cells in life sciences research, enabling the observation of various biological processes [1].

Recent years have witnessed numerous application methods to analyse microarray gene expression datasets [2, 3]. One of the important techniques associated with microarray analysis methods is clustering. Clustering is often used in microarray data to identify groups of samples/genes and then arranging the groups so that the closest groups are adjacent. Numerous microarray clustering techniques has been reported in the literature in recent years [4, 5]. The recent advances in genomic signal processing (GSP) techniques allow the provision of new and robust clustering techniques for microarray data analysis with potentially superior and optimized gene clustering characteristics.

The main objective of this thesis is to present, validate and provide comparative analysis of some of these GSP methods for enhanced microarray data clustering compared to existing methods.

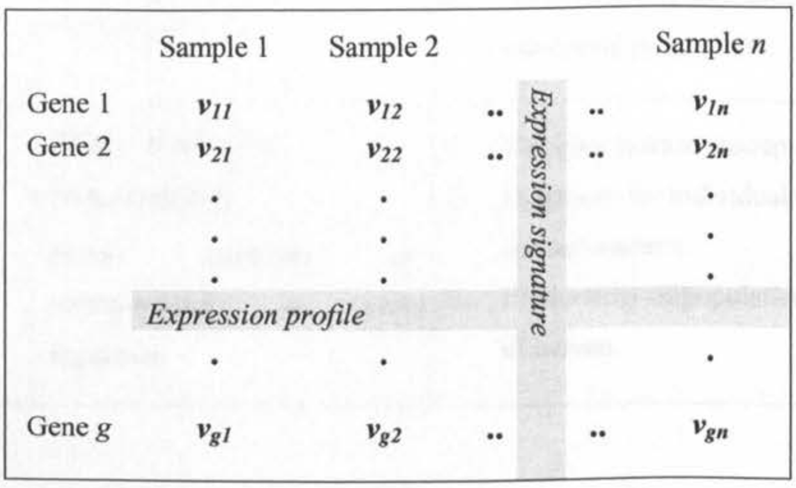
1.1- Motivation of the research

1.1.1- Microarray technologies

Recent advances in genetic information and microarray technologies enable rapid and effective analytical systems for genetic data [3]. A microarray consists of measurements of relative expression levels of mRNA species in a set of related biological samples. Parallel measurement of these expression levels result in data vectors that contain thousands of values called expression patterns. In general, microarray technologies make it possible to probe for all the genes of an entire genome using a single chip. They are powerful tools for extracting and interpreting simultaneous gene activities and relevant genomic information [6].

Analysis of microarray genetic data provides novel opportunities to broaden knowledge about various life phenomena and study many problems in biological and medical research allowing for example better understanding of genetically based diseases [7]. As shown in Figure (1.1) microarray data can be represented by an expression matrix whose rows represent the expression profiles (genes) and columns the expression signature patterns (samples) collected under a variety of conditions (e.g. different patients or time series). This makes microarray data characterized by high dimensionality of genes relative to low dimensionality of samples. Therefore, preprocessing is often applied to reduce the dimensionality and normalize the expression data prior to any further analysis. Expression microarrays usually provide two types of information. First, they can be used to catalogue genes which are expressed in a particular cell or tissue sample. Secondly, they can be used to study dynamic changes in gene expression over time.

Microarray technologies have also the potential to contribute to new healthcare diagnoses. New tools based on genes and proteins could be developed to make predictions for individual patients instead of traditional clinical practices. For example in cancer research, microarray gene expression data are used to study the molecular differences of tumours [8]. This can improve cancer treatment, where the challenge is to design specific therapies which target pathogenetically distinct tumours, by using gene expression patterns to discriminate between carcinoma cells and normal cells. Similar arrays have been designed for Diabetes and obesity to find patterns of different gene expression in adipose tissue between obese and lean mice [9, 10]. Moreover, there are numerous other disease platforms such as Cardiovascular, Colon, Ophthalmic diseases where microarrays have been applied. The analysis of microarray data usually requires specific statistical methods to perform clinical predictions, discovery of diagnostic classes, and selection of relevant genes or groups of genes that cause a given disease.



$$\begin{array}{l|cccccc}
 & \text{Sample 1} & \text{Sample 2} & & & & \text{Sample } n \\
\text{Gene 1} & v_{11} & v_{12} & \cdot\cdot & & \cdot\cdot & v_{1n} \\
\text{Gene 2} & v_{21} & v_{22} & \cdot\cdot & & \cdot\cdot & v_{2n} \\
 & \cdot & \cdot & & & & \cdot \\
 & \cdot & \cdot & & & & \cdot \\
 & \cdot & \cdot & & & & \cdot \\
 & \cdot & \cdot & & & & \cdot \\
\text{Gene } g & v_{g1} & v_{g2} & \cdot\cdot & & \cdot\cdot & v_{gn}
\end{array}$$

b- Arrays GEM

Figure 1.1 Matrix representations of microarray data

1.1.2- Types and functionalities of microarrays

There are several types of microarray technologies which have different functionality and outcomes [11]. A brief description of these types is introduced here as shown in Table (1-1) and each type discussed next. For completeness, further details on the different types can be found in [12]. In this thesis we focus on the first type (cDNA) due to their data availability.

Table 1-1 Summary of Microarray types and functionalities

Microarrays	Functions	Applications
cDNA and Olig Microarray	Analysis of gene expression levels *Monitor an entire genome on a single chip and determine the level at which gene is expressed*	1. Tracking gene expression. 2. Stages of disease progression. 3. Key molecular mechanism. 4. Case of drug therapy and stage of development response. 5. Detection of gene expression pattern and the difference.
CGH Microarray	Comparative Genomic Hybridization (CGH) *Detect rate of chromosomal aberrations*	1. For molecular disease diagnosis. 2. Implications for risk assessment of gene flow and prognostic staging. 3. Detect and map the tumour associated progression.
SNP Microarray	Single Nucleotide Polymorphism *Detect mutation or polymorphisms in gene sequences*	1. Tracking disease susceptibility. 2. Diagnosis for individual disease risk assessment. 3. Exploratory of population genetics of human.

i. cDNA and Oligonucleotide Microarrays

These are "Expression chips" of microarray which allows to determine the level at which a certain gene is expressed. Two predominant types of DNA microarray technology are designed: high density oligonucleotide (Olig) as an absolute expression level and cDNA microarrays as relative to the expression levels of a suitably defined common reference sample. Each technology has specific consideration for measuring levels of gene expression as described next. Table (1-2) shows a comparison of the two types of microarray and their functionalities. The structure and preparation of these is shown in Figure (1.2).

Table 1-2 Summary of the functionalities of the cDNA and Olig arrays

cDNA Array	Oligonucleotide Array
Long sequences of synthesis	Short sequences due to the limitation of the synthesis technology
Spot unknown sequences of nucleotide	Spot known sequences
More flexible and variability in the system	More reliable data
Easier to analyze with appropriate experiment design	More difficult to analyze
Cheap	Expensive
Low density	High density
Relative value measurement	Absolute value measurement
There is a difference between each individual channel (dye) on the same array	There is a difference between the overall mean of each individual array

Type I: A probe of desired cDNA sequence (500~5,000 bases long) material is immobilized onto glass slides using robot spotting. Small quantities of DNA are deposited on the array in the form of spots. Traditionally this method is called DNA microarray and is widely considered as development tool.

Type II: A printing sequence of oligonucleotide (20~80-mer oligos) representing each gene probe is synthesized in situ (on-chip) followed by on-chip immobilization using photolithographic techniques, and is similar to the technology used to build (VLSI) circuits used in fabrication of electronic components. Historically, this method is called DNA chips, and developed at Affymetrix Inc. under the GeneChip® trademark.

To produce microarrays, the cDNA is derived from the mRNA of known genes of normal tissue conditions (1) and diseased tissue conditions (2). The two different conditions are extracted and labelled with two different fluorescent labels, a Green dye for cells at condition (1) and a Red dye for cells at condition (2), to visualise signals from the two samples. Both extracts are washed over the microarray. Then each is hybridized on a glass slide at a known position in the array.

Figure 1.2 Schematic overview of Olig. and cDNA microarray [14]

From a fluorescent microscope and image analysis tools based on the log (green/red), the fluorescence intensities and colours for each spot of mRNA hybridizing at each site are measured. The spot indicates relatively how much mRNA with the corresponding sequences is present in the original sample of cells. If the RNA from the sample in condition (1) is in abundance, the spot will be green, if the RNA from the sample in condition (2) is in abundance, it will be red. If both are equal, the spot will be yellow, while if neither is present it will not fluoresce and appear black.

To obtain information about gene expression levels, these images should be analyzed, each spot on the array identified, its intensity measured and compared to the background. This is called image quantitation. To obtain the final gene expression matrix from spot quantitation, all the quantities related to some gene (either on the same array or on arrays measuring the same conditions in repeated experiments) have to be combined and the entire matrix has to be scaled to make different arrays comparable. If a gene is over expressed in a certain disease state, then more sample cDNA, as compared to control cDNA, will hybridize to the spot representing that expressed gene. In turn, the spot will fluoresce red with greater intensity than it will fluoresce green [13].

Microarray technology is a hybridisation-based process that has been exploited to generate a vast amount of data examining the gene expression pattern, genotyping, tissue and protein studies.

ii. Comparative Genomic Hybridization microarrays:

The Comparative Genomic Hybridization (CGH) is a process where fluorescently labelled patient and control whole-genomic DNA are hybridized to normal metaphase slides to look for either genomic gains and losses or a change in the number of copies of a particular gene involved in a disease state [12]. Figure (1.3) illustrates the CGH microarray technology. Differential hybridization signals allow the detection of unbalanced gains and losses of chromosomal material across the whole genome. In microarray CGH, large pieces of genomic DNA serve as the target DNA, and each spot of target DNA in the array has a known chromosomal location. The hybridization mixture will contain fluorescently labelled genomic DNA harvested from both normal tissue (control) and diseased tissue (sample).

Therefore, if the number of copies of a particular target gene has increased, a large amount of sample DNA will hybridize to those spots on the microarray that represent the gene involved in that disease, whereas comparatively small amounts of control DNA will hybridize to those same spots. As a result, those spots containing the disease gene will fluoresce red with greater intensity than they will fluoresce green, indicating that the number of copies of the gene involved in the disease has gone up.

Figure 1.3 Schematic overview of CGH microarray [15]

iii. SNP microarray

These microarrays can be used to detect mutations or polymorphisms in a gene sequence. In this type, the target, or immobilized DNA is usually that of a single gene. Here, the target sequence placed on any given spot within the array will differ from that of other spots in the same microarray, by only one or a few specific nucleotides. A common type of mutations studied in this type of analysis is called a Single Nucleotide Polymorphism (SNP). It corresponds to a small genetic change or variation that can occur within a person's DNA sequence [16].

1.1.3- Microarray data representation

As shown in Figure (1.1), the array is defined as a Gene Expression Matrix (GEM) and summarised by a matrix $V=(v_{gn})$ where cells v_{gn} denotes expression level of genes, rows correspond to the different *g* genes (variable), and the columns represent *n* different mRNA expression samples (observation). The samples vary according to experimental conditions and physiological states.

The GEM array can partition into rows *R*, or into columns *C* as shown:

$$V_{row}=[\ R_1\ \ R_2\ \ ....R_r.....R_g]^T \qquad\qquad V_{col}=[C_1\ \ C_2\ \ ....C_k.....C_n]$$

where,

$$R_r=[\ V_{r1}\ \ V_{r2}\ \ ....V_{rk}.....V_{rn}]^T \qquad\qquad C_k=[\ V_{1k}\ \ V_{2k}\ \ ....V_{rk}.....V_{gk}]$$

where $1\leq k \leq n$ and $1\leq r \leq g$. The row vector R_r corresponds to the expression levels of the r^{th} gene under *n* conditions. The column vector C_k corresponds to the expression levels of the *g* genes under the k^{th} condition. The row vector conditions ($1xn$) and the column vector genes ($1xg$) are defined to keep track of every condition and gene.

The aim of clustering is that given a dissimilarity measure, *n* points are grouped into *U* clusters based on their similarity. The principle of clustering technique is to share similar functions of genes having similar expression profiles or functions across a dataset.

When the expression samples belong to known classes (e.g., Leukaemia), the data for each observation consist of a *gene expression profile* $V_i=(v_{i1},v_{i2},...,v_{in})$ and a class label y_i, that is, of predictor variables v_i and response y_i. For *U* tumour classes, the class labels y_i are defined to be integers ranging from *1* to *U*, and n_u denotes the number of observations belonging to class *u*. These issues will be detailed in chapter2.

There are two approaches associated with the clustering analysis of the GEM. The first is to compare expression profiles of genes by comparing the rows of the expression matrix, whereas the second approach is to compare expression profiles of samples by comparing the columns of the expression matrix. The comparison of either rows or columns can be used to determine the similarities or dissimilarities between the data pairs. If two rows (genes) are found to be similar then it can be said that the respective genes are co-regulated and have similar functions. By comparing columns (samples), one can determine which genes are differentially expressed and then study the affects of various compounds on this expression. In this work, we focus on *sample clustering* i.e. enhanced clustering for determining which genes are differentially expressed for specific diseases and target selected for this work.

1.2- Genomic Signal Processing for microarray clustering

Genomic Signal Processing (GSP) is an emerging engineering discipline that aims to analyse the profiles of genomic information in order to understand the structural and functional genomics using Digital Signal Processing (DSP) methods. It is concerned with the processing of genomic signals to gain biological knowledge and translate into system-based application [17, 18]. In general, application of GSP is directed towards the simultaneous analysis of interaction among groups of gene samples and provides expression analysis system based clustering.

As an emerging discipline, GSP integrates numerous DSP mathematical and computational methods with the global understanding of genomics through the construction of new genomic functional models. For microarray clustering, GSP methods have the potential to provide enhanced clustering methods compared to existing methods. This is due to the fact that the expression profiles can be transferred to a spectral component – that can be interpreted more easily as a measure of the similarity of expression profiles.

Figure (1.4) shows the processing steps of GSP based microarray data clustering. These are summarized in the following steps:

1. Dimension reduction. Since gene expression data are highly dimensional and contain short multivariate time series, it requires pre-processing to reduce the dimensionality of the gene expression variables. This can be achieved either statistically by selecting the most expressed genes or by specifying the number of genes in the profile.

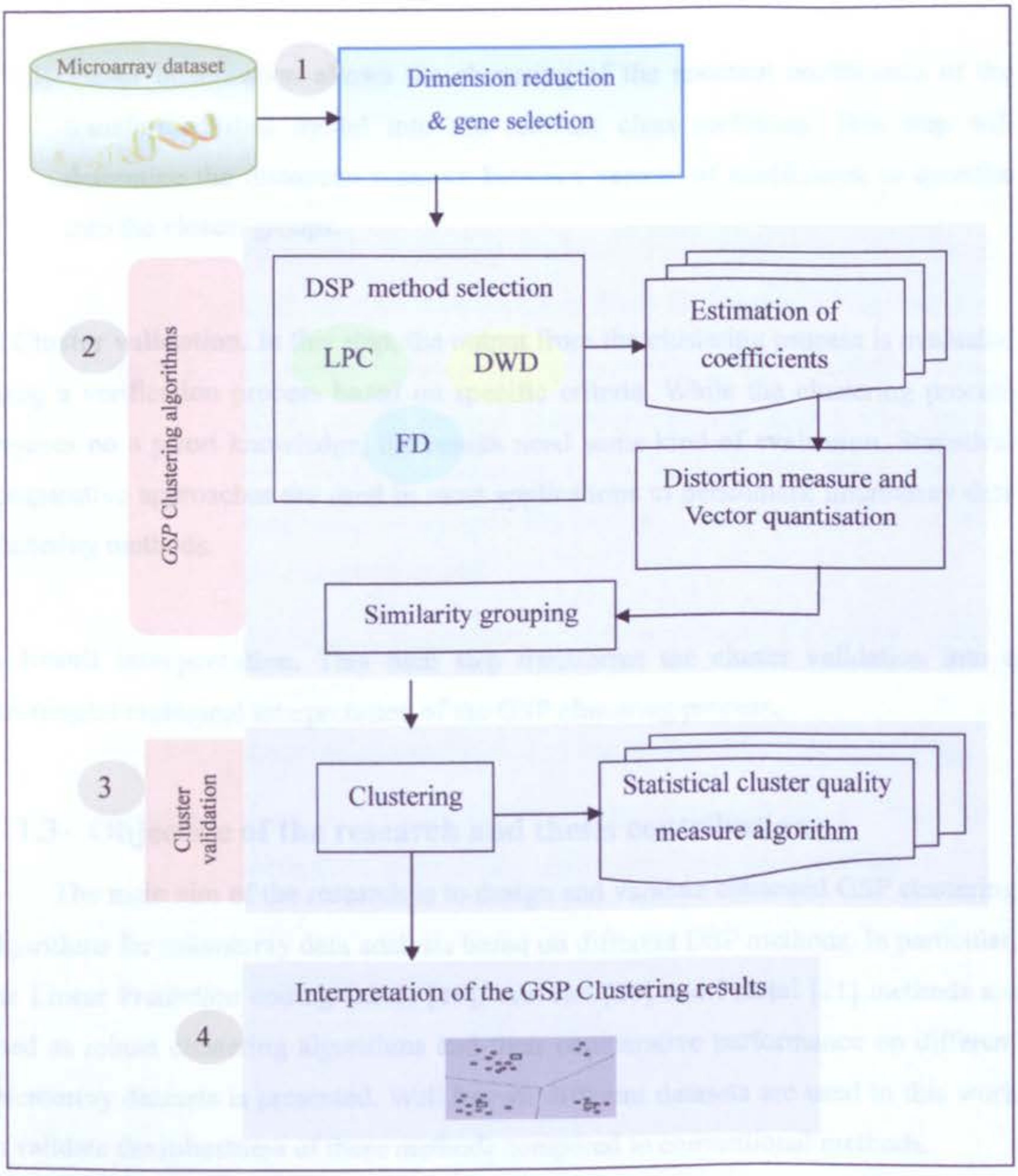


Figure 1.4 GSP based microarray clustering

2. Clustering algorithm. It refers to the selection of relevant clustering algorithms that produce the best data clustering. Specifically for this work and for GSP clustering, the approach is divided into the following steps:

(i)- DSP selection: In this, the specified DSP method is selected to translate the signal into a representation relevant to the vector of expression profile and to find the best predictive coefficients for the microarray model. This step will also determine the proximity measure relative to the similarity-quantified measurement between two vectors of the coefficients.

(ii)- Vector quantisation allows the clustering of the resultant coefficients of the transformed data model into the relevant class partitions. This step will determine the distortion measure between vectors of coefficients to quantise into the closest groups.

3. **Cluster validation.** In this step, the output from the clustering process is evaluated using a verification process based on specific criteria. While the clustering process requires no a priori knowledge, the results need some kind of evaluation. Statistical comparative approaches are used in most applications to benchmark microarray data clustering methods.

4. **Result interpretation.** This final step transforms the cluster validation into a meaningful biological interpretation of the GSP clustering process.

1.3- Objective of the research and thesis contribution

The main aim of the research is to design and validate enhanced GSP clustering algorithms for microarray data analysis based on different DSP methods. In particular, the Linear Prediction coding (LPC) [19], Wavelet [20] and Fractal [21] methods are used as robust clustering algorithms and their comparative performance on different microarray datasets is presented. Well known different datasets are used in this work to validate the robustness of these methods compared to conventional methods.

The specific aims of this research project and major contributions are summarized as follows:

1. Extensive literature review and studies on different microarray clustering and analysis methods.
2. Improvement, application and comparative performance analysis of three advanced DSP methods to microarray clustering.
3. Comparative validation of these methods on different cDNA microarray dataset samples and performance analysis with existing clustering methods.
4. Develop a new and specific MATLAB® tool programs that interpret the DSP methods for enhanced microarray clustering.

1.4- Outline of the thesis

The reminder of the thesis is outlined as shown below:

Chapter 2 includes an extensive literature review of relevant work on microarray clustering methods. The chapter also outlines the GSP and spectral analysis methods for microarray clustering.

Chapter 3 presents a detailed description of the Linear Predictive coding (LPC) and the general clustering design method. It also explains the Vector Quantisation (VQ) method used to predict the coefficient to estimate the similarity between samples to build the clustering model.

Chapter 4 describes the application of discrete wavelet for microarray GSP clustering.

Chapter 5 describes the application of fractals in microarray clustering.

Chapter 6 presents comprehensive comparative analysis of these GSP methods to number of well-known disease test platforms.

Chapter 7 concludes the work and addresses the possible future research directions in this area.

CHAPTER 2

Microarray data analysis

In general, data mining is the process of discovering knowledge or hidden patterns in large datasets that have a meaningful and interesting view from a particular point. In the context of DNA microarray data, the result from extracting information can be to group together the genes that are tightly co-expressed over a range of different experiments, that is, to cluster samples with similar functionality [22]. It is well known that two main approaches of data mining have been used to analyse gene expression data either in a **supervised** or an **unsupervised** approach. Supervised methods are used when there is a prior knowledge of the number of groups and the primary characteristics of each group. In this case, a new aspect is classified depending on its characteristics in one or more predefined groups. The unsupervised method is about the organisation of a collection of unlabeled patterns (data vector) into clusters based on similarity, it assumes no prior knowledge of the data; the idea is to discover the intrinsic structure from data itself. The capability of hybridising these two methods is possible in which an unsupervised approach is followed by a supervised method.

In this chapter we introduce these clustering methods as used in microarray data analysis. The detailed description of the signal processing methods for microarray gene expression relevant to this work with comprehensive literature review in this area is also presented.

2.1- Review of microarray data analysis methods

In recent years, numerous methods are applied to analyse microarray data from different perspectives and the methodology such as statistical, computational and machine-learning approaches. In this section we present an overview of the existing microarray data analysis methods with a brief description of the techniques used in microarray data analysis as shown in Figure (2.1).

1- Normalisation:

Generally the raw expression data after image analysis and quantification steps may be carried out with a preprocessing step involved with the data normalization, by which expression levels are made comparable. There are two basic normalization methods: *Global* per-chip normalization is a scaling that enforce the averages of the expression distributions (expression levels for all genes within an array) to have equal mean, and *Local* per-gene normalization which compares the results for a single gene across all the samples. The objective behind the normalization methods is that the amount of transcription is mainly similar across the samples, when the differentially expressed of the genes could be occurred.

Recent work [23] compares three different normalization approaches, namely: Loess Regression (LR), Splines Smoothing (SS) and Wavelets Smoothing (WS). In addition, two other methods are also proposed, called Kernel Regression (KR) and Support Vector Regression (SVR). The results obtained from this work indicate that the SVR is the most robust and that the Kernel is the least effective normalization technique, while no practical differences were observed between Loess, Splines and Wavelets. Other similar work on normalisation methods are cited in [24].

2- Feature extraction:

Microarray data are high dimensional complex data structures which consist of a large number of features (*g* Genes) and a small number of instances (*n* Samples), typically, *g* and *n* are in the order 10,000 and 100 respectively. Therefore advanced analysis methods are required to emphasize features hidden in the data array.

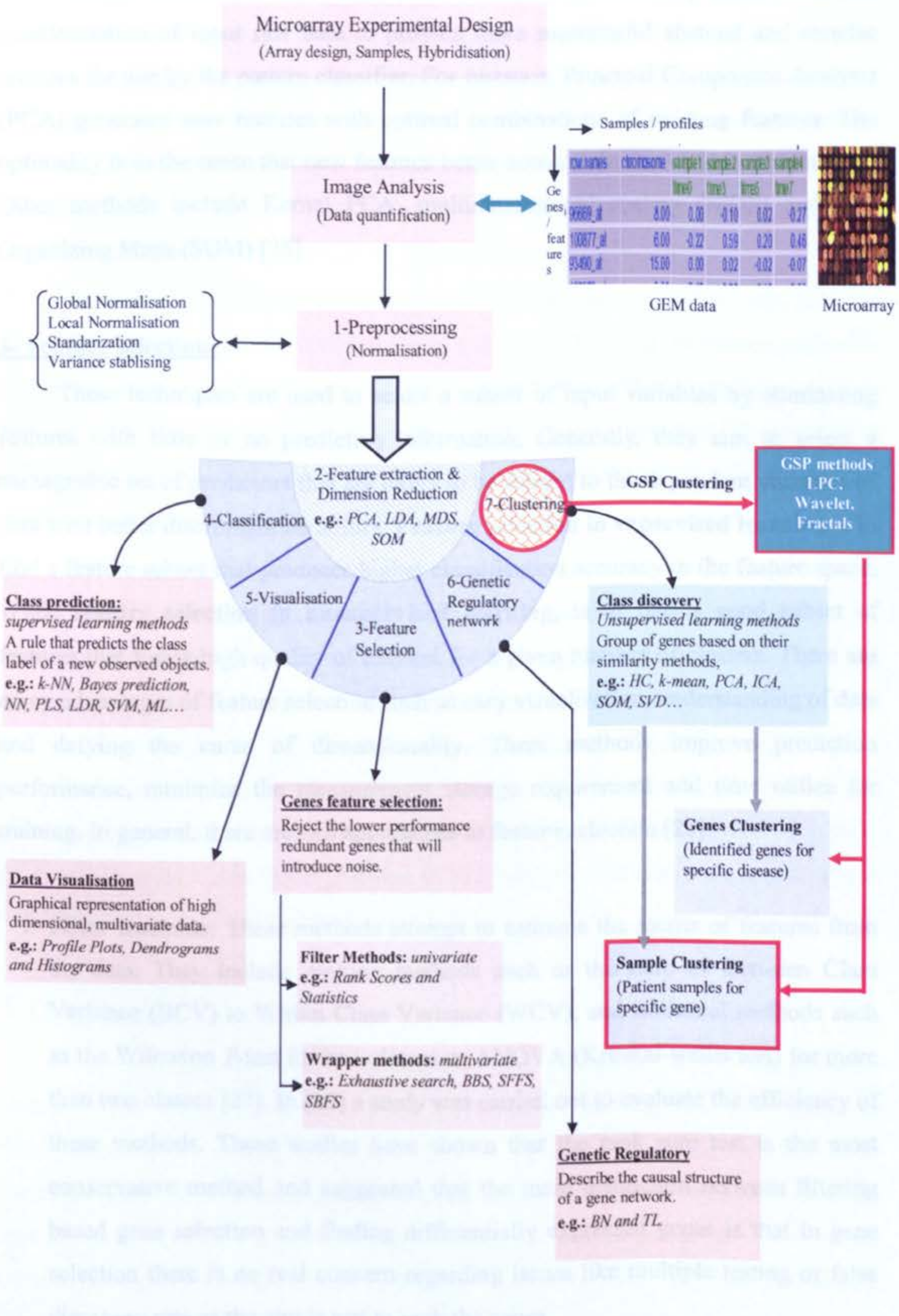

Microarray Experimental Design
(Array design, Samples, Hybridisation)
Samples / profiles
Image Analysis
(Data quantification)
Genes / features
row.names chromosome sample1 sample2 sample3 sample4
time0 time3 time5 time7
96669_at 8.00 0.00 -0.10 0.02 -0.27
100877_at 6.00 -0.22 0.59 0.20 0.46
93490_at 15.00 0.00 0.02 -0.02 -0.07
GEM data
Microarray
Global Normalisation
Local Normalisation
Standarization
Variance stablising
1-Preprocessing
(Normalisation)
2-Feature extraction &
Dimension Reduction
e.g.: PCA, LDA, MDS, SOM
4-Classification
7-Clustering
GSP Clustering
GSP methods
LPC,
Wavelet,
Fractals
5-Visualisation
3-Feature Selection
6-Genetic Regulatory network
Class prediction:
supervised learning methods
A rule that predicts the class label of a new observed objects.
e.g.: k-NN, Bayes prediction, NN, PLS, LDR, SVM, ML…
Class discovery
Unsupervised learning methods
Group of genes based on their similarity methods,
e.g.: HC, k-mean, PCA, ICA, SOM, SVD…
Genes feature selection:
Reject the lower performance redundant genes that will introduce noise.
Genes Clustering
(Identified genes for specific disease)
Data Visualisation
Graphical representation of high dimensional, multivariate data.
e.g.: Profile Plots, Dendrograms and Histograms
Filter Methods: univariate
e.g.: Rank Scores and Statistics
Sample Clustering
(Patient samples for specific gene)
Wrapper methods: multivariate
e.g.: Exhaustive search, BBS, SFFS, SBFS
Genetic Regulatory
Describe the causal structure of a gene network.
e.g.: BN and TL.

Figure 2.1 Microarray data analysis methods (red blocks correspond to the focus of this work)

Feature extraction, also known as dimension reduction, deals with the transformation of input raw data to provide more meaningful abstract and concise features for use by the pattern classifier. For instance, Principal Component Analysis (PCA) generates new features with optimal combinations of existing features. The optimality is in the sense that new features better account for the variation in the data. Other methods include Kernel PCA, multidimensional scaling (MDS) and Self Organizing Maps (SOM) [25].

3- Feature selection:

These techniques are used to select a subset of input variables by eliminating features with little or no predictive information. Generally, they aim to select a manageable set of predictors that are likely to be related to the dependent variables of data with better discrimination ability. **Feature selection in supervised learning** is to find a feature subset that produces higher classification accuracy in the feature space. While **feature selection in unsupervised learning**, is to find a good subset of features that forms high quality of clusters for a given number of clusters. There are many advantages of feature selection such as easy visualization, understanding of data and defying the curse of dimensionality. There methods improve prediction performance, minimize the measurement storage requirement and time utilize for training. In general, there are two approaches to feature selection [26]:

i. **Filter methods**: These methods attempt to estimate the merits of features from the data. They include ranking methods such as the ratio of Between Class Variance (BCV) to Within Class Variance (WCV), and statistical methods such as the Wilcoxon *T*-test for two classes or ANOVA (Kruskal-wallis test) for more than two classes [27]. In [28] a study was carried out to evaluate the efficiency of these methods. These studies have shown that the rank sum test is the most conservative method and suggested that the main distinction between filtering based gene selection and finding differentially expressed genes is that in gene selection there is no real concern regarding issues like multiple testing or false discovery rate as the aim is just to rank the genes.

ii. **Wrapper methods**: These methods estimate subsets of variables according to their predictive power. The method conducts a search for a good subset using the learning algorithm itself as part of the evaluation function. Therefore many smart algorithms for searching the gene subset space have been proposed. Among these are Branch-and-Bound-Search (BBS), Sequential Forward/Backward Selection (SFS/SBS), and Sequential Forward/Backward Floating Search (SFFS/SBFS)[26-29].

Since filter methods are based on processing the whole signal at once, they are faster than wrapper methods that depend on search and learning. Another argument is that some filters (e.g. those based on mutual information criteria) provide a generic selection of variables, not tuned by a given learning machine. Another compelling justification is that filtering can be used as a preprocessing step to reduce space dimensionality and overcome over fitting.

Many feature Selection methods were implemented for microarray analysis to select individual genes as single variant analysis by applied statistical methods t-test and Wilcoxon rank-sum test that compute the correlation between individual genes and a class variation [8, 30]. These studies show that the predictions based on an informative subset of genes are more accurate than those that are based on all genes. In another approach groups of genes were selected as multivariate analysis by applied PCA and Singular Value Decomposition (SVD) methods that merge the most relevant combination of gene expressions in a group [31].

4- Classification:

Generally, the aim is to assigning data to a predefined set of categories or classes. These methods rely on a set of objects called training data. The classes to which these objects belong to are identified as dependent variables, and the set of variables describing different features of these objects is called independent variables that are used to build a predictive model. It can be used to predict the class of the objects for which class information is not known a priori. Since the classification is a supervised learning method that requires an explicit knowledge of the classes the different objects belong to, these classification methods can perform an effective feature selection that leads to better prediction accuracy. Common classification methods used in gene expression data analyses originate from statistical and probability methods [32]. Among them, there are the Logistic Regression (LR) and Fisher Discriminant Analysis (FDA) methods that deal with the high dimensional nature of microarray data by initially reducing their dimension. Since these methods rely on linear functions, they are incapable to express nonlinear relationships in microarray data. Therefore, these difficulties cause inaccurate prediction result. Alternatively, classification algorithms are based on machine learning techniques, such as Support Vector Machine (SVM), Neural Network (NN), Naive Bayesian Network (BN) and *k*-Nearest Neighbour (*k*-NN), these are capable of accurate analysis of microarray data [33]. Other methods attempt to reduce the microarray dimension using a Partial Least Squares (PLS) followed by classification of the data using Quadratic Discriminant Analysis (QDA) [34].

A comprehensive evaluation of classification methods for cancer diagnosis based on microarray gene expression is presented in [35]. This study concludes that MultiCategory (MC-SVM) methods are the most effective classifiers in performing accurate cancer diagnosis in comparison with other machine learning methods like *k*-NN or NN. The gene selection techniques in these methods could significantly improve the classification performance of both MC-SVMs and other non-SVM learning algorithms, and that ensemble classifiers generally did not improve performance of the best non-ensemble models. An alternative method called Penalized Logistic Regression (PLR) was proposed [36] to deal with the common weakness of SVM: given a tumour sample, SVM only predicts a cancer class label but does not provide any estimate of the underlying probability for the microarray cancer diagnosis cases.

Moreover, a combined method based on Independent Component Analysis (ICA) and Regularized Regression models for analysing gene expression data was presented in [37]. The gain of the approach could make full use of the high order statistical information contained in the gene expression data, then implement regularized regression models to handle the situation of large numbers of correlated predictor variables. The experimental results showed that the method is efficient and practical in comparison with other methods.

Other classification methods based on spectral component analysis were also investigated. An autoregressive technique was used to evaluate the potential regulatory relationship between genes with dominant spectral components [38]. This technique summarizes the essential features of an expression pattern by means of an estimated frequency spectrum. Specifically, the pattern is decomposed into a set of damped sinusoids of different frequencies so that each sinusoid can be considered separately during the analysis. Hence, this method allows the flexibility of ignoring irrelevant frequency components that may otherwise be too overwhelming.

5- Visualization:

The visualization module includes the ability to visualize and drill down interactively into the details of the resultant data. A graphical representation refers to the visual interpretation of complex relationships in multidimensional data [39]. In [39], two main components were used: "intrinsic visualisation" (graphical drawing) to combine object relationships to spatial distances by sighting similarity object closed together, and "extrinsic visualisation" (visual form). In [40] visualization techniques were applied in the time series modelling framework together with graphical models to built a three dimensional prototype to demonstrate the visualization effect of the modelling results.

6- Genetic Regulatory network:

The goal of these techniques is to generate gene networks from microarray data using network modelling method. It is used for observing the interrelationship mechanisms between genes within a genetic regulatory system, which activates specific group of genes by particular signals and regulates a common biological process. For example the group members may regulate each others transcriptions. Such groups are called genetic regulatory systems [41]. Network modelling is used for observing the interrelationship between genes within a genetic regulatory system. Particularly, a Bayesian Network (BN) is a probabilistic model which illustrates the multivariate probability distribution for a set of variables [41]. It is used to find the network structure and the corresponding model parameters which describe best the probability distribution for which the dataset is drawn in a graphical representation.

Implementation of BN in microarray is to reveal relationships between various genes, by extracting the information about their dependencies and independencies of the encoded set of variables of the dataset and visualizing this relationship by a network structure, which is interpreted as a genetic network. Combined with other experiments such a resulting network may be a basis to reveal new functions of genes and their proteins.

Other researchers proposed a quantitative method for determining functional interactions in cellular signalling and gene networks [42]. In this method a mechanical level is applied within a "modular" framework, which dramatically decreases the number of variables to be assumed. The method was based on a mathematical derivation that demonstrates how the topology and strength of network connections can be retrieved from experimentally measured network responses to successive perturbations of all modules. The network analysis could reveal functional interactions even when the components of the system were not all known. Under these circumstances, some connections retrieved by the analysis were not direct but correspond to the interaction routes through unidentified elements. The method was tested and illustrated using computer-generated responses of a modelled Mitogen-Activated protein kinas cascade and gene network.

Recently, a gene regulatory model was proposed based on generated Threshold Logic (TL) rules when a given gene interaction graph is presented [43]. The rule generation method is fairly simple and depends on the given gene interaction data and any additional biological data. An important feature of this model is that it is adaptable and consistent with biological data. Threshold logic has long been known as an alternative to Boolean Logic which had been used to model Gene Regulatory Networks (GRNs). It was demonstrated that the new TL-GRN could model inter-cellular and intra-cellular gene regulation. The advantage of this model was that it could be used to generate accurate rules with limited data. These rules required few parameters to estimate and were simple to determine. Another advantage was that the resulting gene networks can be simulated in hardware efficiently by using Differential Current-Mode Logic (DCML) gates.

7- Clustering:

This is the most relevant part to the work of this thesis. Commonly, cluster analysis is a well known approach to discover meaningful set of subgroups that are more similar to one another than to members of other clusters. A detailed presentation of the clustering analysis is given next. In general, clustering methods depend on altered definition of best possible criteria and type of similarity measure to a critical diagnosis. Clustering is also closely related to the techniques used to create the codebooks used in vector quantisation, a technique to be discussed next.

2.2- Clustering analysis

In this section we describe the general principle of clustering and the relevant processing blocks to the work in this thesis. Basically there are two parameters to define a cluster, the *similarity measure* between objects and the *subsequent grouping* of objects into clusters associated to the outcome similarities. *Cluster tightness* is defined by the minimum distance of objects within group variance, whereas, well separated cluster separation is represented by the maximum distance between group variance.

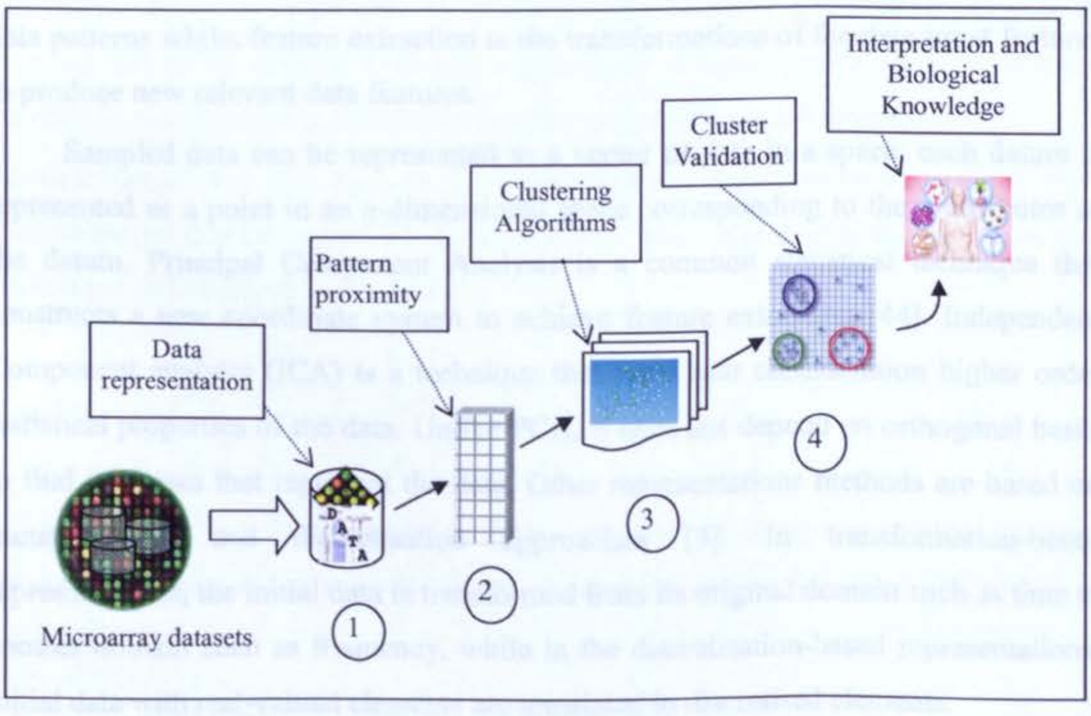


Figure 2.2 General microarray clustering process

Pattern clustering methods used in microarray analysis consist of four steps as illustrated in Figure (2.2) [3]:

1. **Data Representation**. It refers to number of patterns, features, and desired clustering methods used.
2. **Pattern Proximity**. It defines a distance and similarity measurement method suitable with the selected representation or model utilized.
3. **Clustering Algorithm**. It aims to group the represented or modelled data according to the similarity previously defined. It is referred as pattern grouping.
4. **Cluster Validation**. It either validates or scores clustering results.

Next we describe each of these clustering steps.

2.2.1- Microarray data representation

The modelling of data patterns may involve two feature based steps, i.e. "selection" and "extraction", which describe or signify most elements about a datum. Clustering methods used either or both of those features in their analysis. Feature selection is used to identify the most significant elements of the original features in data patterns whilst feature extraction is the transformations of the data input features to produce new relevant data features.

Sampled data can be represented as a vector of data in a space: each datum is represented as a point in an *n*-dimensional space corresponding to the *n*-attributes of the datum. Principal Component Analysis is a common statistical technique that constructs a new coordinate system to achieve feature extraction [44]. Independent Component analysis (ICA) is a technique that takes into consideration higher order statistical properties of the data. Unlike PCA, it does not depend on orthogonal bases to find the bases that represent the data. Other representations methods are based on transformation and discretisation approaches [3]. In transformation-based representations, the initial data is transformed from its original domain such as time to another domain such as frequency, while in the discretisation-based representations, initial data with real-valued elements are translated to discretised elements.

Moreover, mathematical model can describe the data model by relations that determine how the system varies from one state forwards to next state depending on the current or other relevant values of the same variable. Statistical models allow the characterisation of a system or variable based upon its statistical parameters such as mode, median, mean, variance, regression coefficients, least-squares fit to some mathematical equation. Examples of statistical models are autoregressive and generative models [45]. In the autoregressive model, the built model depends on the statistical properties of the past behaviour of variables, whilst the generative model supposes that the data is generated by some primary probability distribution.

2.2.2- Microarray data Proximity

Similarity measurement is referred as the likeness, or identicality of two objects. Referring to the GEM represented in Figure (1.1), assuming that n is the total number of samples V prepared for clustering process, a $v_i \in V$; $i = 1, \dots, n$ is represented by a feature vector of samples in g dimensions as $v_{gi} = \{v_{g1}, v_{g2}, \dots, v_{gi}\}$; $i = 1, \dots, n$ The samples are represented conventionally as multidimensional vectors, with a dimension presenting a single feature either quantitative or qualitative description of the object. In order to be able to assess the similarity, a quantitative measure of likeness has to be utilised to create a proximity distance matrix. It is a common practice to use distance or correlation metrics to quantify such likeness.

Given a dataset $X = \{x_1, x_2, \dots, x_n\}$ representing the object which is described by a d-dimensional feature vector, the distance matrix M_{dist} $(d_{i,j})$ represents how close two objects are. It is defined in Eq. (2.1):

$$M_{dist}(d_{i,j}) = \begin{bmatrix} 0 & d_{12} & \dots & d_{1n} \\ d_{21} & 0 & \dots & d_{2n} \\ \vdots & \vdots & 0 & \vdots \\ d_{n1} & d_{n2} & \dots & 0 \end{bmatrix} \qquad (2.1)$$

Where $d_{ij}=d(x_i, x_j)$ with respect to some distance function, while the similarity measure is how similar are the objects. Similarity matrix M_{sim} $(s_{i,j})$ is defined in Eq.(2.2) below:

$$M_{sim}(s_{i,j}) = \begin{bmatrix} 1 & s_{12} & \dots & s_{1n} \\ s_{21} & 1 & \dots & s_{2n} \\ \vdots & \vdots & 1 & \vdots \\ s_{n1} & s_{n2} & \dots & 1 \end{bmatrix} \qquad (2.2)$$

Where $s_{ij}=S(x_i, x_j)$ with respect to similarity measure between two objects and by definition $S_{ii}=1$

The suitability and performance of a similarity measure for a specific comparison depends on the nature of the objects to be compared. When using model-based clustering techniques the complete data is fitted to a model and there is no direct comparison between data points. Therefore, the clusters are implicit in the model structure. For microarray gene expression analysis, there are different methods used to measure the distance and quantify the likeness between samples. They are summarised next.

i. Distance functions

In general, clustering techniques require some measure of similarity or distance between *(x)* vector and *(y)* vector. A scalar function, *d(x,y)*, is a distance function of the pairs of vector if it satisfies the following axioms requirements of a distance measure[3]:

Nonnegativity	$d(x, y) \geq 0$
Reflexivity	$d(x, y) = 0$ if x=y,
Symmetry	$d(x, y) = d(y, x)$
Triangle in equality	$d(x, y) \leq d(x, z) + d(z, y)$ *for any z*

There is a variety of different measures of inter-observation distances and inter-cluster distances which can be used as criteria when merging nearest clusters into broader groups. It is useful to summarize several commonly used distances [3].

1- Euclidean distance

It is the most common distance function. It is defined as the distance $d_e(x,y)$ measured along a straight line between two points *(x,y)* in the data-space as shown in Eq (2.3):

$$d_e(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \qquad (2.3)$$

2- City block or Manhattan distance

It is defined as the rectilinear route measured parallel to the axes as shown in Eq. (2.4):

$$d_c(x, y) = \sum_{i=1}^{n} |x_i - y_i| \qquad (2.4)$$

3- Minkowski distance

Obviously, both Euclidean distance and City block distance are special cases of the Minkowski measure [3], Where *m=2* for Euclidean and *m=1* for City block to. The value of m depends on the amount of emphasis placed on the larger differences $|xi- yi|$

$$d_m(x, y) = \left(\sum_{i=1}^{n} |x_i - y_i|^m\right)^{1/m} \qquad (2.5)$$

While other distance methods are:

Mahalanobis distance is used to normalise based on a covariance matrix to make the distance metric scale-invariant. Chebyshev distance is used to measure distance assuming only the most significant dimension is relevant.

ii. Similarity functions

In general, the similarity measures how similar two objects are, whilst objects which are similar share a low distance. Similarities also have some properties when representing two points (data objects x and *y*):

Nonnegativity	$0 \leq S(x, y) \leq 1$
Max. Similarity	$S(x, y) = 1$ if $x=y$,
Symmetry	$S(x, y)= S(y, x)$

General Approach for Combining Similarities is as follows [25]:

1. For K^{th} object attribute, compute a similarity, S_k in the range *[0,1]*
2. Define an indicator variable, δ_k, for the k_{th} attribute as follows:

$\delta_k = 1$, if the K^{th} attribute is a binary asymmetric attribute and both objects have a value of 0, or if one of the objects has a missing values for the K^{th} attribute.

Otherwise, $\delta_k = 0$,

3. Compute the overall similarity between the two objects using the following formula:

$$S(x,y) = \frac{\sum_{k=1}^{n} \delta_k s_k}{\sum_{k=1}^{n} \delta_k} \tag{2.6}$$

Correlation and covariance measures are statistical concepts used to measure either the association of the relationship between two random variables or measure the mutual relation of the outcomes of the process at time instants .The most usual measure is expressed by Pearson's correlation coefficient $\rho(x,y)$, defined in Eq 2.7, it reflects the difference between the elements x and y to be compared relative to the standard deviation:

$$\rho(x,y) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{2.7}$$

$$d(x,y) = 1 - \rho(x,y) \tag{2.8}$$

The Pearson's correlation coefficient has been suggested as a metric as shown in Eq. 2.6 which focuses on whether the coordinate of two vectors change in the same way. Moreover it can identify statistically significant gene expression clusters and helps identifying genes that regulate each other or have similar cellular function [45]. Furthermore, the not centred correlation coefficient was found to conform well to the intuitive biological notion of what it means for two genes to have similar expression.

The covariance sequence is the mean-removed cross-correlation sequence, it is a measure of the scatter, or the dispersion, of the random process around the mean value, i.e. how much the deviation of two variables match. It is defined as:

$$C_{xy}(m) = E\left\{(x_{n+m} - \mu_x)(y_n - \mu_y)\right\} \tag{2.9}$$

Where x_n and y_n are stationary random processes, μ represent means, and E is the expected value operator representing the sum.

Pearson's correlation coefficient has been used to measure the similarity of a potential regulatory relationship of the two genes expressional [46, 50].

iii. Probability

Similarity can be obtained by the highest probability of sampled data fits in the available models. Bayes theorem is fundamental to engineering pattern classification solutions to data mining problems [32]. It relates the conditional probabilities between two stochastic variables. The posterior probability, which is the probability of a given element belonging to a particular model, can be calculated from the prior probability and the likelihood.

Let events $\{C_1, C_2, \ldots . C_k\}$ from a partition of the space S such that the Prior Probability distribution $P(C_i)>0$ for all i and let D be any event such that $P(D)>0$ represents evidence of prior probability for all i. Bayes theorem is a rule for computing and updating the posterior probability of events C_i given D from prior probability $P(C_i)$ and the conditional probability $P(D/C_i)$ of D given each event C_i. The computation of the likelihood of event D given C_i, is given as:

$$P(C_i/D) = \frac{P(D/C_i)P(C_i)}{P(D)} \tag{2.10}$$

The updated probability can then be used for comparison to compute the similarity measurement obtaining how close the data fits the model.

The procedure of the probability method is naturally divided into two phases. Firstly the algorithm learns from samples with known class membership representing as training session or defines as a prior knowledge. Secondly a predict rule is established to classify new samples in a test session. Several papers have used this probability to assess the similarity of genes in a variety of statistical models [47, 48].

iv. Mutual Information (MI)

It is an information theory measure. It provides a general measurement for dependencies in the data with such properties as positive, negative, and nonlinear correlations, in order to identify genes that share inputs to which they respond differently [49]. The dependence criteria perform better than metric methods, and are a more generalized measure of correlation, which provides advantages in gene expression analysis [50]. Other information theory measures, such as entropy-based measure have also been used in the clustering of microarray gene expression [51].

The information present in microarray gene expression can be quantified using Shannon entropy H. It can be calculated from the probabilities P of occurrences of individual or combined events. Given two random variables x, y, then H is calculated as follows:

$$H(x) = -\sum P_x \log P_x$$
$$H(y) = -\sum P_y \log P_y \qquad (2.11)$$
$$H(x,y) = -\sum P_{x,y} \log P_{x,y}$$

Mutual information M is defined as the sum of individual entropies subtracted from the entropy of the co-occurrence as follows,

$$M(x,y) = H(x) + H(y) - H(x,y) \qquad (2.12)$$

Application of MI measure requires the expression patterns to be represented by discrete random variables; therefore M between two expression patterns can be expressed as follows:

$$M(x,y) = \sum_i \sum_j P_{i,j} \log \frac{P_{i,j}}{P_i P_j} \qquad (2.13)$$

Where $P_i=P(X=x_i)$ and $P_j=P(Y=y_j)$ are the probability distribution functions. Usually M is non-negative and equal zero if and only if X and Y are statistically independent; this signifies that X contains no information about Y and vice versa. It means that the patterns do not follow any kind of dependence, which is impracticable in correlation or distance measure as represent by Eq. (2.14). Further advantages are: it is a generalized measure of statistical dependence in the data, and reasonably immune against missing data and outliers [52].

$$d(x,y) = \sum_i w_i \, M\{u_i \neq v_i\} \qquad (2.14)$$

v. Spectral Distortion

It is a signal processing approach used to calculate the distortion between the pair of spectral vector of data in array. The concept of pattern comparison is measured based on the similarity distance. Spectral distortion measure method provides a similarity distortion measure between vectors of gene expression data. This effectively works by finding the similarity between waveform shapes. Further details on the spectral distortion methods are described in the next section.

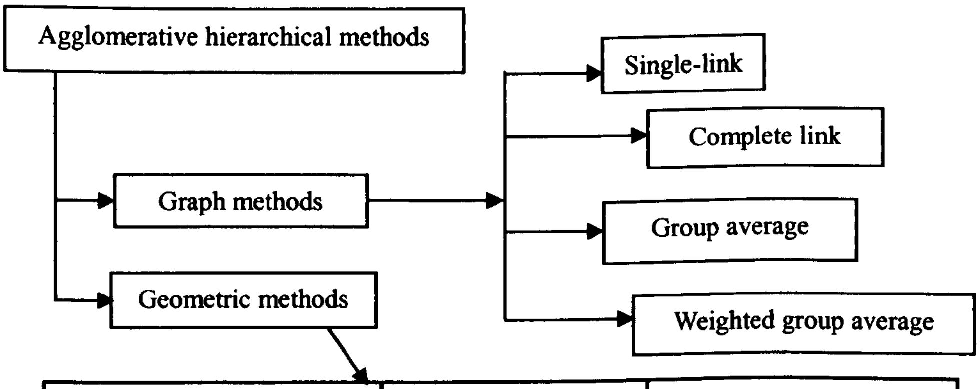
2.2.3- Microarray clustering algorithms

2.2.3.1- Conventional microarray clustering methods

In recent years extensive work has been carried out in this area. A brief summary of the most efficient methods are given in [33]. For completeness a brief description of these methods is presented next

i. Hierarchical techniques

Hierarchical clustering (HC) constructs appropriate tree structures between samples, typically accomplished with smaller data samples size. These iterative methods are based on degree of similarity by either forwards merging smaller clusters into larger ones based on bottom-up approach called agglomerative, or backwards by splitting larger clusters based on top-down approach which is called divisive.



Hierarchical methods	$\mu(C_i \cup C_j)$	Similarity between C_i and C_j
Centroid, uses the average distance between all pairs of objects in cluster *i* and clusters	$\frac{\lvert C_i\rvert\mu(C_i)+\lvert C_j\rvert\mu(C_j)}{\lvert C_i\rvert+\lvert C_j\rvert}$	$\lvert\mu(C_i)-\mu(C_j)\rvert^2$
Median, uses the Euclidean distance between the centroids of the two clusters	$\frac{\mu(C_i)+\mu(C_j)}{2}$	$\lvert\mu(C_i)-\mu(C_j)\rvert^2$
Ward, uses the incremental sum of squares	$\frac{\lvert C_i\rvert\mu(C_i)+\lvert C_j\rvert\mu(C_j)}{\lvert C_i\rvert+\lvert C_j\rvert}$	$\frac{\lvert C_i\rvert\lvert C_j\rvert}{\lvert C_i\rvert+\lvert C_j\rvert}\lvert\mu(C_i)-\mu(C_j)\rvert^2$

*where $\mu(C)$ denotes the centre of cluster C

Figure 2.3 Hierarchical clustering methods [52-54]

The final result of the algorithm is a tree of clusters called a *dendrogram*, which shows how the clusters are interrelated. By cutting the dendrogram at a desired level a clustering of the data items, disjoint groups are obtained. Hierarchical agglomerative cluster analysis is the strategy that is the most commonly used for microarray data analysis [52-54]. Figure (2.3) describes the methods required to determine which groups should be combined or separated. The main reasons for this approach are its use of a simple technique with a small number of samples to produce high accuracy and the visual results which have a clear structure instead of divisive hierarchical approach.

ii. Partitioning-Optimisation techniques

These techniques are different from hierarchical techniques in that they allow relocation of the elements; they allow poor initial partitions to be corrected at a later stage. A centroid or a cluster representative can represent each cluster; this is some sort of summary description of all the objects involved in a cluster.

These techniques can be considered as attempts to partition the dataset in a way that optimises some predefined criterion. In this category, *K*-mean is a commonly used algorithm; its criterion function is described as follows:

$$E = \sum_{i=1}^{c} \sum_{x \in c_i} d(x, m_i) \tag{2.15}$$

where m_i is the centre of cluster C_i, while $d(x,m_i)$ is the Euclidean distance between a point x and m_i. Thus, the criterion function E attempts to minimize the distance of each point from the centre of the cluster to which the point belongs. Most of the techniques have three distinctive steps: initialisation of clusters, allocation of elements to initialised clusters and reallocation of some or all of the elements to other clusters once the initial segmentation has been completed. Fuzzy c–means and k-means are other common partitioning optimisation techniques which have been used for gene expressions [55]. Large datasets can be processed with K-means clustering, unlike hierarchical, because K-means does not require prior computation of a proximity matrix of distances and similarities, but it is sensitive to outliers.

iii. Model-Based techniques

These clustering algorithms can be developed based on a statistical probability model, such as the finite mixture model for probability densities. A likelihood (or posterior probability) derived from this model is used as the criterion to be optimised. The model is usually used to represent the type of constraints and geometric properties of the covariance matrices [56]. The type of model has to be specified according to the objectives of the cluster analysis and the properties of the dataset. The structure of the chosen model can usually be selected by model selection techniques. The parameters are estimated based on modelling the distribution of samples in dataset. There has been a considerable amount of research involving the model based technique. They include models that have been used for gene expression using Hidden Markov Models [57], and mixed-effects models with B-splines [58]. Another approach based on modelling the distribution of the gene expression profile of test sample as a finite mixture of an unknown number of distributions, with each mixture component characterizing the gene expression levels within a class, assumes that each class has a multivariate normal density with diagonal variance-covariance matrix [58]

iv. Grid-Based techniques

The approach of these techniques begins with dividing the space into a finite number of cells. Cells that have more than a predefined number of elements are treated as dense and the dense cells are connected to form the clusters. In general, a typical algorithm for this method consists of the following steps:

1. Creating the grid structure, this means partitioning the data space into a finite number of cells.
2. Calculating the cell density for each cell.
3. Sorting of the cells according to their densities.
4. Identifying cluster centres.
5. Traversal of neighbour cells.

The most representative techniques are: STatistical INformation Grid-based method STING [59] and WaveCluster [60]. The WaveCluster is based on signal processing techniques (wavelet transformation) to convert the spatial data into frequency domain efficiently to discover clusters with arbitrary shape. It handles outliers by being insensitive to order of input. It initially summarizes the data by imposing a multidimensional grid structure onto the data space. Each grid cell summarizes the information of a group of points that map into the cell. Then it uses a wavelet transformation to transform the original feature space: convolution with an appropriate function results in a transformed space where the natural clusters in the data become distinguishable. Thus, the clusters can be identified by finding the dense regions in the transformed domain. A-priori knowledge about the exact number of clusters is not required.

v. Density-Based techniques

These methods are capable of finding arbitrarily shaped clusters, where clusters are defined as dense regions separated by low-density regions. Density is usually defined as the number of objects in a particular neighbourhood of data objects. The main idea is to classify a data object as one of the "cores" of a cluster if it has more neighbours than a predefined threshold within a predefined neighbourhood. Clusters are formed by connecting neighbouring "core" objects and those "non-core" objects which are either within the threshold boundaries of clusters or become outliers. Common representatives of these techniques are Density Based Special Clustering of Applications with Noise (DBSCAN) [61] and Ordering Points to Identify the Clustering Structure (OPTICS) [62]. The difficulty of these approaches is that it is hard to choose the parameter values, such as the density threshold. The works [63,64] present some density- based clustering techniques for gene expression.

vi. Graph-based techniques

The graph-based approach first constructs a graph and then applies a clustering algorithm to partition the graph. Each element of the collection to be clustered is associated to a vertex on a graph. An edge from each element to every other is built, and a weight representing the extent to which the elements are similar is associated to this edge. Finally edges in the graph are cut to form a good set of connected components. Each of these will be a cluster. Graph-based techniques are widely used in microarray gene expressions. CLuster Identification via Connectivity Kernels CLICK [65]. These approaches include Enhanced Cluster Affinity Search Technique (E-CAST) where a dynamic threshold is computed at the beginning value of each new cluster [66].

vii. Topographic-based techniques

The topographic clustering algorithms simultaneously identify subgroups of similar data and preserve information about the relationships between the subgroups [67]. These are associated with positions in the output space such that their relative closeness reflects the similarity of the data they contain. The most representative algorithm is the self-organising map (SOM). In the classical SOM, a set of nodes is arranged in a geometric pattern, typically 2-dimensional lattice. Each node is associated with a weight vector with the same dimension as the input space. The purpose of SOM is to find a good mapping from the high dimensional input space to the 2-D representation of the nodes. One way to use SOM for clustering is to regard the objects in the input space represented by the same node as grouped into a cluster. Because of the topographic properties of the clustering, the SOM is commonly used in gene expression analysis [68]. Hybrid approaches have been proposed where the different techniques are combined. For example, grid-based or density-based techniques can be used for cluster initialisation in partitioning-optimisation techniques, and topographic-based and model-based techniques can follow a hierarchical clustering scheme.

2.2.3.2- Comparison of clustering techniques

In this section we present a comparative summary of the clustering methods used in microarray analysis. Table (2-1) shows a comprehensive comparison of unsupervised clustering analysis techniques used for microarray gene expression and the respective advantages and disadvantages of each method. From this analysis, it can be seen that the choice of method depends on the specific application domain.

In general, the following issues should be considered when selecting any clustering algorithm:

1. All clustering methods usually return a clustering result no matter how much information the data actually contains, therefore there is only need to obtain list of units in the selected clusters.
2. Clustering alone is only for exploratory, visualization and hypothesis generating tool and not a biological proof.
3. Accuracy and precision that reflect how the results are close to the true values besides the detection of the sensitivity.

Table 2-1 Comparison of different of microarray clustering methods

Method/ type	Characteristic	Advantage	Disadvantage
Hierarchical Clustering/	Cluster using agglomerative or divisive approach, uses a dissimilarity matrix to merge data successively to construct the tree or dendogram	➢ Intuitive algorithm and Straight forward. ➢ No explicit criterion. ➢ Do not need to estimate number of clusters. ➢ The number of clusters can be found from the dendrogram. ➢ Deterministic partitions.	➢ Common tree is susceptible to outliers. ➢ Unreadable when tree is large ➢ Doesn't give discrete clusters, need to define clusters with cutoffs. ➢ Get different clustering for different experiment sets. ➢ Computationally slow.
K-mean/	It is non-Hierarchical that identifies clusters by minimizing the overall within cluster variance. It partitions the data points into k disjoint subsets based on a distance measure between instances.	➢ Minimize sum of squares, as simplified Gaussian mixture model. ➢ Computationally fast. ➢ It can identify coregulated genes where some prior knowledge can be used to predict the appropriate number of clusters *K*.	➢ Unstructured nature of it tends to proceed in a Local minimum. ➢ Does not allow scattered genes. ➢ Random partitions. *K*-means may produce different partitions depending on the initialization. ➢ The number of clusters has to be given in advance
SOM/	It provides a mapping from multidimensional input space to a two-dimensional output space while processing topological relations closely.	➢ Clusters are interpreted on 2D geometry (more interpretable). ➢ Can preserve the topology. ➢ Local order and local clustering structures shown on the map display are as dependable a possible. ➢ Useful for large data amount. ➢ Strong visualization capability.	➢ The algorithm is based on heuristic. ➢ Solutions are sub-optimal due to 2D geometry restriction. ➢ The size of the map needs to be carefully decided. ➢ Do not provide an accurate layout when displaying global structure.
PCA/	It reduces the data with high dimensionality by performing a covariance analysis between factors, It can also allow a visual inspection of the relationship between them.	➢ Reduce the dimensionality of the data to summarise the most important parts whilst simultaneously filtering out noise. ➢ It based procedure in prediction accuracy. ➢ Orthogonal transform, second order statistics.	➢ Difficult to visualise nonlinear structures consisting of arbitrarily shaped clusters or curved manifolds. ➢ Improper for large datasets.
ICA/	It is a statistical method, to decompose given multivariate data into a linear sum of statistically independent components.	➢ Non orthogonal transform, high order statistics, related to the projection pursuit. ➢ The reduced model is based on the occurrence of common motifs in the genes promoter sequences.	➢ Sign magnitude problem and hard to select proper ICA. ➢ Sensitive to modes whose influences on the genes follow 'superGaussian' distribution with large tails and a pronounced peak in the middle.
EM/	They detect clusters in observations and assign those observations to the clusters	➢ Maximize the overall probability or likelihood of the data, given the (final) clusters ➢ Can be applied to both continuous and categorical variables	➢ Does not compute actual assignments of observations to clusters

2.2.4- Microarray cluster validation measures

In order to evaluate the spectral clustering approaches, validation methods assess whether the clustering algorithm with the specified parameters such as number of clusters, similarity measure, model, etc., can identify the underlying patterns of the analysed dataset. Several cluster quality or validity measures have been proposed in the literature [69]. Cluster validity measures quality of a clustering relative to clusters created either by other clustering algorithms, or by the same algorithms using different parameter values. The validity measure should reflect the quality of the clusters based on the objectives of the clustering algorithm [69,70].

Table 2-2 Cluster Validation Measures

Relation	Type	Description
$D_{min}(C_i, C_j) = \min\lvert p_i - p_j \rvert$ *Where* $p_i \in C_i$ *and* $p_j \in C_j$	Minimum based Distance	Min distance between all pair of objects drawn from the two clusters
$D_{mean}(C_i, C_j) = \lvert m_i - m_j \rvert$ *Where* m_i *and* m_j *are centriods of* C_i *and* C_j	Mean based Distance	Distance between the means of the two clusters
$D_{avg}(C_i, C_j) = \frac{1}{n_i n_j}\Sigma\Sigma\lvert p_i - p_j \rvert$ *Where* $p_i \in C_i$ *and* $p_j \in C_j$, *and* n_i *and* n_j *are numbers of samples in clusters* C_i *and* C_j	Group-average Based Distance	The average of the distance between all pairs of individuals that are made up of one individual from each cluster

In general, based on a distance measure *d* between samples, it is possible to define a distance measure *D* between clusters (set of samples). These measures are an essential part in estimating the quality of a clustering process, and therefore they are part of clustering algorithms. The validation method is based on cluster compactness (in term of intra cluster variance) and density between clusters (in term of inter cluster density). Inter cluster density evaluates the average density in the region among clusters in relation to the internal density of the clusters, while intra cluster variance measures the average scattering of clusters. Therefore, a good clustering method will have an intra density which is much higher than its inter density. Table (2-2) illustrates the widely used measures definitions for minimum distance between clusters (C_i , C_j). However, for clustering measure the well established method depends on the basic measure definitions.

Typically, a cluster quality measure is a statistical measure that quantifies the performance quality of the clustering results. In this thesis we used two methods to measure the clustering quality performance of the implemented algorithm which are: Davies-Bouldin index and Silhouette width [71] as it has been widely recognised as the best validation method. In that work [71] both methods use data samples and cluster centroids determined from GSP cluster algorithm in the quality measure. These indices are described next:

1. Davies-Bouldin (DB) index,

In this index, the clustering result obtained from GSP clustering algorithm divides the dataset into N clusters is represented as: $C=\{C_1, C_2, \ldots\ldots, C_N\}$, the DB method calculates an index value to each cluster C_i as follows:

$$DB_i = \max_{j=1,..N,i\neq j} DB_{ij} \quad (2.16)$$

$$DB_{ij} = \frac{\{sc(C_i)+sc(C_j)\}}{cd(C_i,C_j)} \quad (2.17)$$

where, $sc(C_i)$ represents average distance of samples (belonging to C_i cluster) to centre of C_i cluster and $cd(C_i, C_j)$ represents the distance between the centres of the C_i and C_j clusters. The value of DB index and clustering quality are considered as directly proportional. The DB index is the average value of all clusters as follows:

$$DB = \frac{1}{N}\sum_{i=1}^{N} DB_i \quad (2.18)$$

2. Silhouette Width (SW) index,

The index exploits the inherent features of clusters to assess the validity of results and select the optimal partitioning for the data under concern. The definition of the (SW) index is based on compactness and separation of the clusters taking also into account density.

To determine the average of *SW*, the *SW* value of each samples in microarray data as SW_i using Eq. (2.19). Then the average *SW* for each cluster is computed. Finally, the overall average *SW* for all samples is calculated as shown in Eq. (2.19)

$$SW_i = \frac{sc(i)-sd(i)}{\max\{sc(i),sd(i)\}} \qquad (2.19)$$

where *sc(i)* is the average distance between the sample *i* to other samples in the same cluster (intracluster distance) and *sd(i)* is the average distance between the sample *i* and other samples which are nearest neighbour cluster to the i^{th} samples cluster (intercluster distance).

$$ASW(c) = \frac{1}{N}\sum_{i=1}^{N} SW_i \qquad (2.20)$$

The average of Silhouette score for SW_i class *C* across all genes reflects the overall quality of the clustering result as illustrated in Eq. (2.19). A larger averaged silhouette width indicates a better overall quality of the clustering result. If the value of ASW is close to 1 it means the sample is in an appropriate cluster. If it is close to 0 it means the sample could also belong to the nearest cluster to the i^{th} samples cluster, if it is close to -1 it means this sample is not in an appropriate cluster.

2.3- Spectral technique for microarray data analysis

The previous section has outlined some of the advantages and disadvantages of existing statistical based clustering methods. However, in recent years there has been a new focus on the application of digital signal processing methods for enhanced analysis of microarray gene expression data. It is well known that microarray data have the capability to be represented by samples sequence of spectral vectors, with the spectral difference or spectral distortion between the pair of the spectra measured for the purpose of pattern comparisons and speech recognition [72]. As we have already discussed in the previous sections, there are two ways for the analysis of a microarray gene expression matrix: either the analysis of expression profiles of genes by comparing the rows of the expression matrix; or the analysis of expression profiles of samples by comparing the columns of the expression matrix. Either comparison can be used to determine the similarities or dissimilarities between data pairs. If two rows (genes) are found to be similar then it can be said that the respective genes are co-regulated and have similar functions. By comparing columns (samples), one can determine which genes are differentially expressed and then study the effects of various traits on this expression. As discussed earlier, we focus on the latter approach. Spectral clustering is a relatively recent approach which is applied to find similarity of spectral data vectors in a matrix through clustering analysis. This becomes, in effect, a dimensional reduction of the space, selected parts of which may be clustered thereafter [73]. A popular application is image segmentation where different regions of the image may be treated separately. In data mining, the technique has also been applied to the division of data available on the datasets. Spectral methods are attractive because they make less severe assumptions on the shape of the clusters than partitioning algorithms and can be very fast, depending on the similarity matrix [74].

Spectral analysis studies have been used in several bioinformatics applications, among them, a spectral component analysis of time-series microarray data has been implemented for the identification of genes that are subjected to common transcriptional regulation [38]. Based on the motivation that the most commonly used approach to determine if the two genes have a potential regulatory relationship is to measure their expressional similarity using the Pearson correlation coefficient, but recognizing that this approach has many limitations. The authors instead proposed an Autoregressive (AR)-based technique. They used the well-known AR modelling technique to characterize temporal gene expression data from the Spellman's a-synchronized yeast cell-cycle experiment. Time-series expression profiles were decomposed into spectral components and correlations between profiles computed. They reported log ratios of the test sample expression over control sample expression level measurement.

A microarray dataset contains a set of gene expression values. These values can be represented as a vector, where the indices identify the spatial location of the dataset in the gene expression scene [75]. A set of gene expression samples may then be considered as a set of spectral reflectance vectors, one for each spatial location. The objective of the spectral clustering analysis is to group together spectral reflectance vectors with similar spectral pattern independent of the vector value. The spectral clustering analysis is used in this work as an unsupervised vector quantisation (VQ) algorithm to reduce the large set of spectral features to small number of feature prototypes based on the measurement of the difference between pair of sample vectors in terms of spectral distortion amount [76].

In particular, we use a combined approach of unsupervised VQ as part of the different enhanced genomic signal processing methods used in this work. The details of these methods will be further explained in the next section.

2.4- Distortion measure based clustering

For spectral clustering analysis of microarray, we use spectral distortion measure to compute the similarity or dissimilarity measure between two data vectors based on various types of distances and distortion measures. Spectral distortion or distance measures are non-negative numerical quantities that map two variables, possibly valued vectors, to a scalar that indicates the degree of difference or dissimilarity between two variables of vectors.

In digital signal processing and communication system, the measurement is evaluated by how *good* the reconstructed signal is compared to the original signal. *Good* can be interpreted differently including computational, complexity and memory allocated size. Hence, there are a number of distortion measures developed for various purposes to extract the objective difference between the two signals. Several distortion measure methods have been proposed in pattern recognitions and speech recognition. Details on these methods are detailed in [77]

Spectral distortion is a distortion measure used widely in speech coding application. This measure has the following advantages: Firstly, the distortion measure may select a model from a codebook that is good in term of its distortion measure where the codebook design is toward a clustering method. Secondly, the distortion measure may reject models with a high distortion which are subjectively good. Consequently, in some instances the matrix of pairwise similarities or distances between the objects to be clustered is replaced by a distortion measure between a data point and a class centroid as in vector quantisation methods, where the aim is to find a relatively small number of classes with high interclass similarity or low interclass distortion and good interclass separation. Consequently, spectral distortion will be used in this work

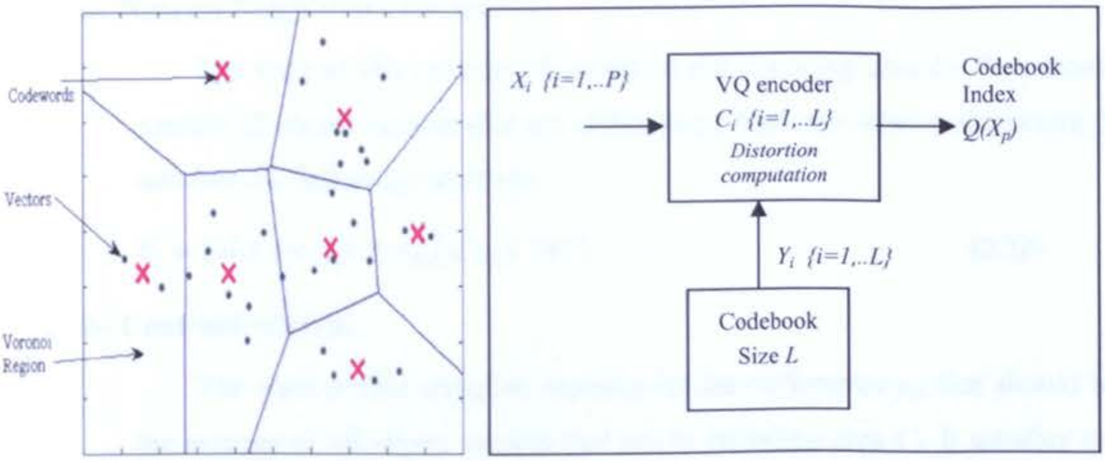
In microarray gene expression data clustering, there are various measures of similarity such as Euclidean distance and correlation between the vectors of expression levels. The advantage associated with correlation method is it captures similarity in shape without emphasis on the altitude of the two series of measurements and being sensitive to outliers. For example, when measuring two different gene expression samples that are fluctuating around the same average value, these samples may be very similar in terms of Euclidean distance (distance close to 0), however dissimilar in terms of correlation (correlation close to 0)

However, Genome signal are characterized with distinguished features such as patterns in the time-frequency domain. In the GSP clustering the parameters prediction analysis is used to represent the spectrum of the genome signal. It has the ability to quantify fragile spectral structure information in the dataset and to provide efficient approximation to the exact spectrum.

In order to cluster the microarray gene expression data samples into several groups, the Vector quantisation based distortion measure with LBG algorithm was used to produces clustering process due to the ability to consider many spectral factors as dimension of gene expression samples vector.

2.5- Distortion measure for VQ of microarray spectrum

Vector quantisation (VQ) is the most straightforward method based on block coding to cluster a set of data in a space using parameter vectors, providing the class directory with labelling of segments [78]. The application of VQ in the microarray clustering process has two main advantages: first it allows capturing meaningful classes in the microarray gene expression data samples, represented by their sample centres, and also it makes subsequent classification decisions robust to the inherent noise within the gene data samples. In VQ a number of gene data samples are grouped together into a target vector and this entire vector is coded. This means that there is a set of code vectors or representation vectors, which form a codebook. The target vector is compared with all code vectors in the codebook by means of a certain distortion measure. The code vector which has the smallest distortion with respect to the target vector is the winning code vector. The VQ algorithm can be stated as follows: given a vector of data source x_p with its statistical characteristics and a set number of codewords y_l which correspond to the centroids (average vector) of the clusters, estimate the distortion measure and then find a codebook index and a partition segment of quantised data as shown in Figure (2.4).



a- VQ Segmentation b- VQ encoder

Figure 2.4 VQ schematic

The principle of VQ is to map *P*-dimensional input vectors $x=[x_1,\dots,x_p]$ by finite set of *L* code words called codebook: $Y = \{y_i, \ 1 \leq i \leq L\}$. To design a codebook, the *P*-dimensional space is partitioned into *L* cells $\{C_i, \ 1 \leq i \leq L\}$, and all cells are quantised, which is the process of assignment one of the code-vectors y_i to each *x* belonging to cell C_i, $q(x_p)=y_i$, if $x_p \in C_i$:

The average quantisation error between input data source and their reproduction codeword is called the distortion of the vector quantiser. The computation procedure of codebook involves allocating a collection of vectors into centroids. The major concern for a vector quantiser codebook design is the trade-off between distortion and rate. Once the number of quantisation levels is defined, the rate is set. Then the focus is on data quantisation as a means of removing noise from data. The centres of the groups of data corresponding to different quantisation levels should be selected so that distortion is minimized. Depending on a squared-error distortion measure, the mean distortion d_m can be given by Eq (2.21).

$$d_m = \frac{1}{PL}\sum_{p=1}^{P}\left|x_p - q(x_p)\right|^2 \qquad (2.21)$$

There are two criteria which satisfy the distortion measure processes used in quantisation process:

a- Nearest Neighbour criteria

The state of this criterion depends on the encoding area C_i that should contain all object vectors that are closer to y_i; than any other codevectors. It satisfies the following relation:

$$C_i = \{x | d_m(x, y_i) \leq d_m(x, y_j) \ ; \forall j\} \tag{2.22}$$

b- Centroid criteria

The state of this criterion depends on the codevector y_i; that should be the average of all object vectors that are in encoding area C_i. It satisfies the following relation:

$$y_i = cent(c_i) = arg \ \ \min_y E\ \{d_m(x, y) | x \in c_i\} \tag{2.23}$$

Figure (2.5) shows the procedure of the VQ algorithm. The design of codebooks is usually accomplished by an iterative algorithm called the Lloyd algorithm. The algorithm was designed as a clustering technique to generate a set of representative vectors of the source data and optimizes the codebook using the distortion measured method [78].

In this work, we use VQ in microarray data space, i.e. a vector z which represents a vector of gene expressions sample is mapped to a code vector q_m of expressions in microarray. The algorithm starts with the preliminary codebook and refines it iteratively. The process continues until no significant further improvement is possible. Implementation of VQ in clustering microarray gene expression samples is as follows:

1- Select the expression vector q_m that is nearest to a vector z, with distortion measure d_m, if the distortion is small enough the algorithm terminates, as defined in the following

$$d_m(z, q_m) = arg \ min_i \ (d_m(z, q_i)) \tag{2.24}$$

2- Assign the resultant microarray codebook C_q as cluster label to the data grouped in q.

Finally, once the codebook has been defined, model coefficient vectors of x are extracted, compared to all codewords of C and mapped to a single codeword that represents the different genes mapped on the tested microarray data.

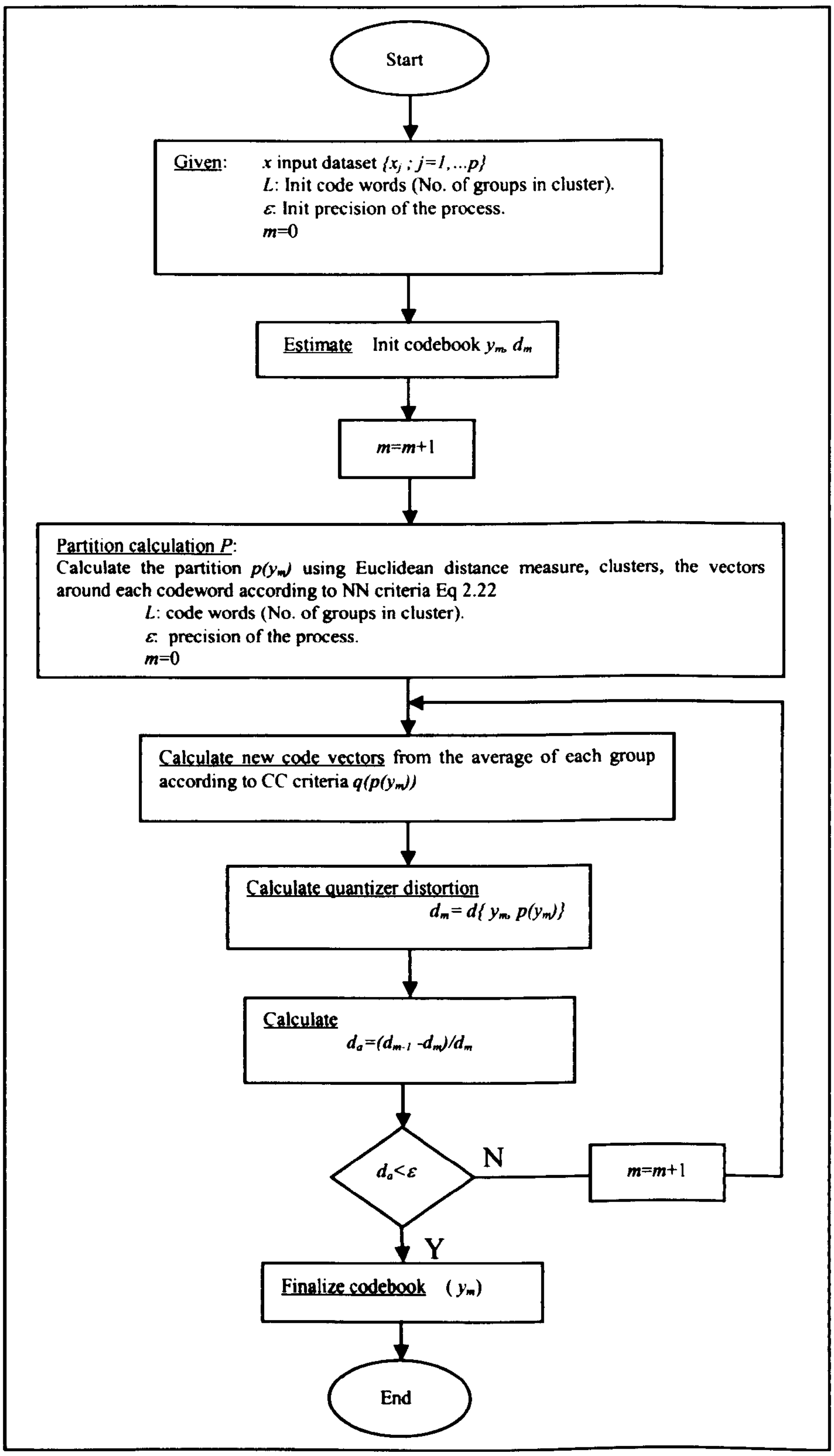

Start
Given: x input dataset $\{x_j ; j=1,...p\}$
L: Init code words (No. of groups in cluster).
ε: Init precision of the process.
m=0
Estimate Init codebook y_m, d_m
m=m+1
Partition calculation P:
Calculate the partition $p(y_m)$ using Euclidean distance measure, clusters, the vectors around each codeword according to NN criteria Eq 2.22
L: code words (No. of groups in cluster).
ε: precision of the process.
m=0
Calculate new code vectors from the average of each group according to CC criteria $q(p(y_m))$
Calculate quantizer distortion
$d_m = d\{ y_m, p(y_m)\}$
Calculate
$d_a=(d_{m-1} -d_m)/d_m$
$d_a<\varepsilon$
N
m=m+1
Y
Finalize codebook (y_m)
End

Figure 2.5 procedure of the VQ algorithm

2.6- Conclusion

In this chapter we have explained the clustering methods used in microarray data analysis, together with extensive literature review and studies on different microarray clustering and analysis methods. The description of the DSP methods and spectral technique for microarray gene expression with comprehensive literature review is presented. From these, it is concluded that GSP methods have the ability to present meaningful information in time and frequency domain from the data especially for unsupervised clustering. The selected DSP methods are selected to translate the spectrum of the genome signal to a vector of prediction parameters, and then vector quantisation is used to produce clustering process. In the next chapter we introduce one of DSP spectral method which is LPC to predict the vector of prediction parameters and discuss applications of the method to microarray dataset.

CHAPTER 3

Linear Predictive Coding for microarray clustering

The origin of Linear predictive coding (LPC) comes from the field of speech processing where a particular value in a signal can be predicted by a linear function of the past values of the signal. LPC is a time series algorithm, which is also known as autoregressive analysis. It has had significant applications in many areas other than the speech analysis field.

The underlying motivation for using Linear predictive analysis for microarray clustering is that it provides a decomposition of the gene expression data samples and predicts future values of the input sample based on past samples. The main objective is to represent a gene expression data sample with a set of coefficients to obtain a predictive gene expression signal with better computational efficiency.

In this chapter the LPC approach for extracting a spectral feature of microarray gene expression coefficients is presented. The chapter outlines the characteristics of the microarray gene sample signals and describes the details of the LPC coefficients. Transformation from the predictive LPC coefficients to the line spectral frequencies has been implemented and then the VQ approach is applied to measure the spectral distortion and compute the dissimilarity or similarity between spectral analysis vectors of the gene samples to produce the relevant index of quantisation for clustering purpose.

3.1- Characteristics of microarray gene samples

In order to investigate the consistency of microarray clustering response based on LPC method, we first examine the microarray gene expression pattern locally as a signal to understand the time series correlation of the gene expression data.

Periodicity is a common phenomenon in biology where periodic processes occur at all levels of biological organization with a cyclic series such as Hormones, Proteins[79]. Figure (3.1) shows typical profile of the gene expression sample signal of the Leukaemia dataset [8] used in this work. Practically, the gene expression data contains rich information based on a set of a finite number of expression value at a time t that can be represented as a vector ($v_{d,t}$) with d-dimensions. Therefore, the gene expression of a set of samples can be represented by the following expression data vector: $G_{exp}=\{v_{d,t}\}_{(d=1,\dots,g)(t=1,\dots,n)}$. Gene expression signal levels show that the variation of the gene expression signal profile involves excitation signals at specific samples. In many circumstances processing gene expression in time series will produce a range of frequencies that will allow finding targets that are expressed periodically with specific correlations both between genes and samples. These data characteristics can be analysed further in the frequency domain using different DSP methods.

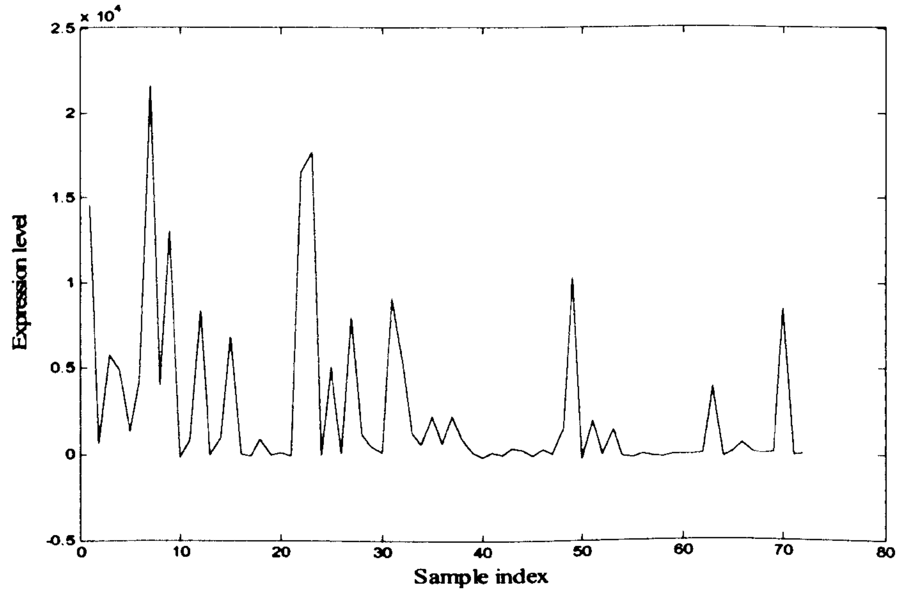
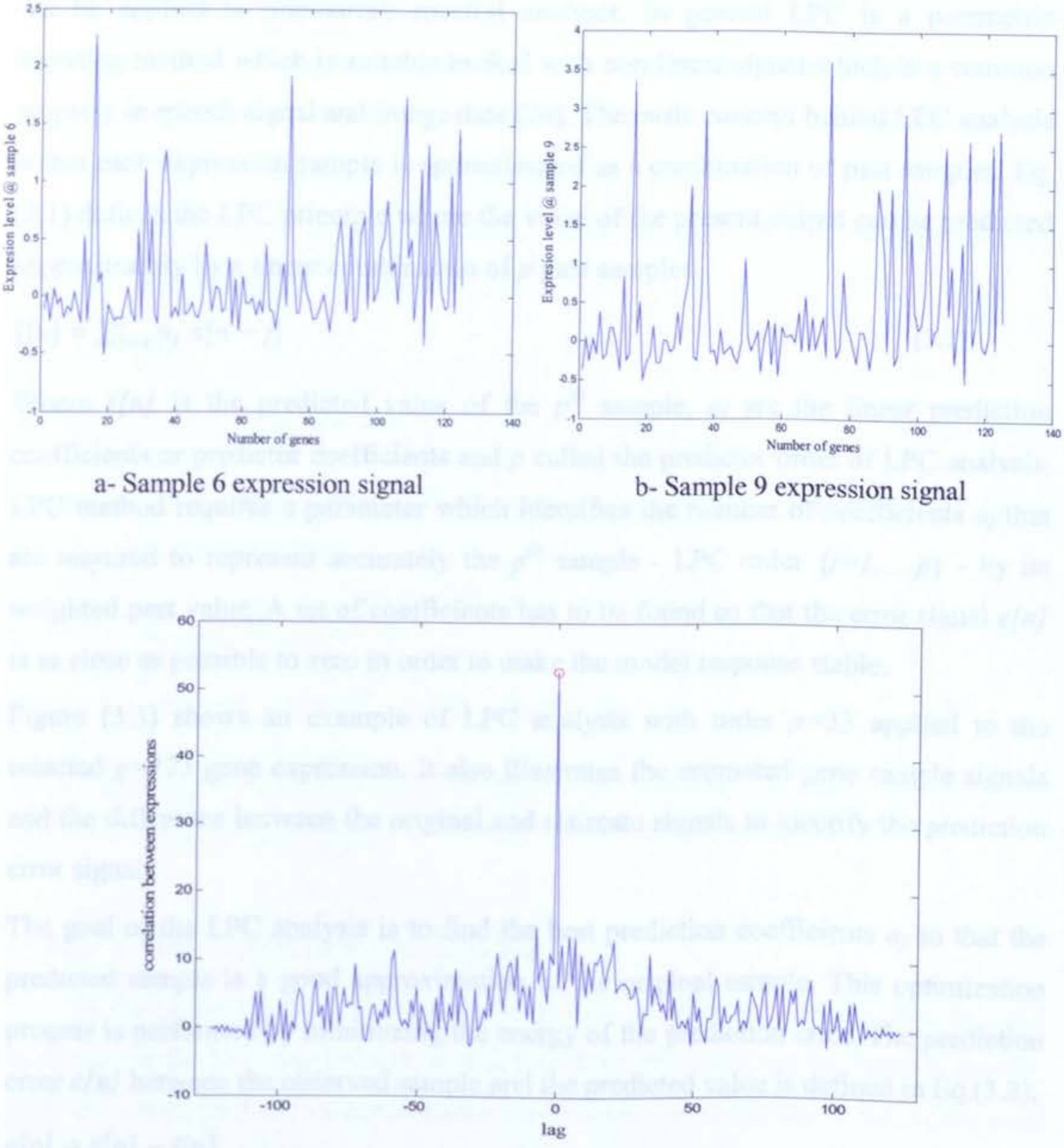


Figure 3.1 Gene Expression levels $G_{exp}=\{v_{d,t}\}_{(d=39)\,(t=1,\dots,72)}$ for Leukaemia dataset, where d is number of genes and t is sample index

Figure (3.2-a,b) shows the results of analysis applied to the gene expression signal *s[n]* between two samples (sample 6 & 9), each with a selection of *g=125*genes. This test shows that the returned vector shown in Figure (3.2-c) indicates the cross correlation of that specific range of oscillation in expression value. The cross-correlation indicates similarity between two samples sequences. The large peak of amplitude highlights this similarity and indicates a good match over the full length of the sample.



a- Sample 6 expression signal

b- Sample 9 expression signal

c- Correlation between sample 6 &9

Figure 3.2 Correlation analyses between expressions signal on Leukaemia dataset using *g=125* genes

In this thesis, we apply the principle of LPC to extract the spectral features of microarray data for enhanced clustering. The computation is based on the principle that the estimated value of a particular microarray value $s[n]$ at position or time n, denoted as $\hat{s}[n]$, can be calculated as a linear combination of the past p samples.

3.2- Background theory of LPC Analysis

The following section describes the concept of the LPC algorithm and how it can be applied to microarray spectral analysis. In general LPC is a parametric encoding method which is suitable to deal with non linear signal which is a common property in speech signal and image data [76]. The basic concept behind LPC analysis is that each expression sample is approximated as a combination of past samples. Eq. (3.1) defines the LPC principle where the value of the present output can be predicted approximately by a linear combination of p past samples.

$$\hat{s}[n] = \sum_{j=1}^{p} a_j \; s[n-j] \qquad (3.1)$$

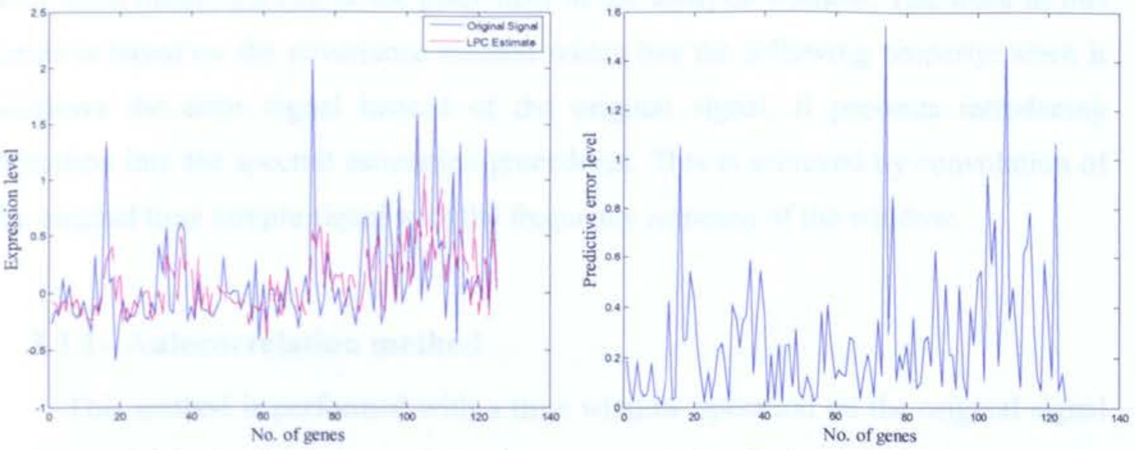
Where $\hat{s}[n]$ is the predicted value of the p^{th} sample, a_j are the linear prediction coefficients or predictor coefficients and p called the predictor order of LPC analysis. LPC method requires a parameter which identifies the number of coefficients a_j that are required to represent accurately the p^{th} sample - LPC order $\{j=1,\ldots,p\}$ - by its weighted past value. A set of coefficients has to be found so that the error signal $e[n]$ is as close as possible to zero in order to make the model response stable.

Figure (3.3) shows an example of LPC analysis with order p=33 applied to the selected $g=125$ gene expression. It also illustrates the estimated gene sample signals and the difference between the original and estimate signals to identify the prediction error signal.

The goal of the LPC analysis is to find the best prediction coefficients a_j so that the predicted sample is a good approximation of the original sample. This optimization process is performed by minimizing the energy of the prediction error. The prediction error $e[n]$ between the observed sample and the predicted value is defined in Eq.(3.2).

$$e[n] = s[n] - \hat{s}[n]$$

$$e[n] = s[n] - \sum_{j=1}^{p} a_j \; s[n-j] \qquad (3.2)$$



a- Original and reconstructed signals b- Predictive error signal

Figure 3.23 LPC test analysis to sample number 23 with *p=33, g=125* and *MSE=0.145*

This involves choosing a_j to minimize the mean energy, *E*, in the error signal over a window of data samples, as follows:

$$E = \sum_n e^2[n]$$

$$E = \left\{\sum_n \left[s[n] - \sum_{j=1}^{p} a_j \; s[n-j]\right]^2\right\} \tag{3.3}$$

The values of a_j that minimize *E* are found by setting all derivatives equal to zero for $(j=1,2,\ldots,p)$ as follows [76]:

$$\frac{\delta}{\delta a_j}(s[n] - \sum_{j=1}^{p} a_j \, s[n-j])^2 = 0$$

Thus, $\sum_{i=1}^{p} s[n] \, s[n-i] = \sum_{j=1}^{p} a_j \, s[n-j] s[n-i]$ (3.4)

Eq. (3.4) is the required formula for estimating the predictor coefficients a_j. $j \, \{1,\ldots p\}$ applied to obtain the predictive coefficients a_j of the predictive model.

3.3- Linear Predictor coefficients

In the following sections we describe briefly the different methods that are used to determine the LPC coefficients, further details on this topic can be found in modern DSP text books [80]. The autocorrelation and covariance methods are two of the most common and efficient linear predictive spectral estimation techniques. Both methods choose the LP coefficients a_j in such a way that the residual energy is minimized. In both methods the classical least square technique is used for such purpose. However, their main difference lies in the placement of the analysis window. The work in this thesis is based on the covariance method which has the following property: since it windows the error signal instead of the original signal, it prevents introducing distortion into the spectral estimation procedures. This is achieved by convolution of the original time sample signal with the frequency response of the window.

3.3.1- Autocorrelation method

This method is performed with a time window operation on the original signal as follows:

$$x[n] = s[n]w[n] \qquad (3.5)$$

Generally, when a window is employed it is assumed the signal sequence *s[n]* is zero outside the analysis frame. Therefore, it limits the input data signal to finite interval $0 \leq n \leq N-1$.The energy in the residual signal becomes:

$$E = \sum_n e^2[n] = \left\{\sum_n \left[x[n] - \sum_{j=1}^{p} a_j\; x[n-j]\right]^2\right\} \qquad (3.6)$$

Eq. (3.6) is solved by differentiating the energy with respect to a_j , *j=1, 2, ...p,* and then equalling the result to zero, $\frac{\delta E}{\delta a_j} = 0$, The resulting equation becomes;

$$\sum_n x[n]\, x[n-i] = \sum_{j=1}^{p} a_j\, \sum_n x[n-j]x[n-i] \qquad i=1,2,...,p \qquad (3.7)$$

The autocorrelation function of the time limited signals *x[n]* is defined as:

$$R(i) = \sum_{n=i}^{N-1} x[n]\, x[n-i] \qquad i=1,2,...,p \qquad (3.8)$$

By substituting the autocorrelation function Eq. (3.8) into Eq. (3.7), the following system of equations is obtained:

$$R(i) = \sum_{j=1}^{p} a_j\, R(i-j) \qquad (3.9)$$

Where *R(i)* is the autocorrelation of *s[n]*. a_j is a predictive coefficient with a vector of length *p*, *R(i-j)* is a matrix of size *p x p*. These *p* equations are known as the Yule-Walker (Y-W) equations for Autoregressive (AR) models [80]. It can be explicitly stated as:

$$R\, a = r \qquad (3.10)$$

While the expressed of the system equation sets could be represented in normal matrix form as follows:

$$\begin{bmatrix} R(0) & R(1) & R(2) & \dots & R(p-1) \\ R(1) & R(0) & R(1) & \dots & R(p-2) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ R(p-1) & R(p-2) & R(p-3) & \dots & R(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} r(1) \\ r(2) \\ \vdots \\ r(p) \end{bmatrix} \qquad (3.11)$$

$a=[a_1, a_2, \dots. a_p]^T$ *where T indicates the transpose of a vector (or matrix).*

The matrix R in Eq. (3.11), which is often called the autocorrelation matrix, has the following properties: it is symmetrical, has a Toeplitz structure, and all the elements along a given diagonal are equal. Therefore, Eq. (3.7) can be solved using computationally efficient recursive procedures, such as the Levinson-Durbin's (LD) algorithm [81] which is the most widely used. Figure (3.4) shows the structure algorithm of the recursive process of the Levinson Durbin recursion method. The coefficients ki, $1 \leq i \leq p$ are computed as a by-product of the LD algorithm. They are known as Reflection Coefficients. They can alternatively represent predictor coefficients.

```
Input: Predictor order P, Autocorrelation coefficients $R(0),\dots.,R(p)$
Output: LP coefficients $a_i=a_1,\dots..a_p$
$E_0=R(0)$
For $i=1$ to $p$ do
    $k_i = \frac{R(i)-\sum_{k=1}^{i-1} a_{i-1}(k)R(i-k)}{E_{i-1}}$
    $a_i=k_i$
    For $j=1$ to $i-1$ do
        $a_i(k) = a_{i-1}(k) - k_i \, a_{i-1}(i-k)$
    End
    $E_i = (1-k_i^2)E_{i-1}$
End
```

Figure 3.4 Levinson Durbin recursion algorithm [81]

3.3.2- Covariance method

This method determines the predictor coefficients by windowing the error signal *e[n]* rather than windowing the original signal, in order to consider all-pole signal modelling. The method is based on minimizing the forward prediction error in the least squares sense. As a result the error in the residual signal becomes:

$$E = \sum_n e^2\,[n]\,w[n]$$

$$= \sum_n\ [\,s[n] - \sum_{k=1}^{m} a_k\, s[n-k]]^2 w[n] \qquad (3.12)$$

Where the error is minimized over a finite interval of size *N* as defined by the rectangular window function *w[n]*.

After minimizing and differentiating Eq. (3.12) with respect to a_k, we obtain:

$$\sum_{n=0}^{N-1} s[n]\,s[n-i] = \sum_{j=1}^{p} a_k \sum_{n=0}^{N-1} s[n-k]\,s[n-i] \qquad i=1,2....p \qquad (3.13)$$

It can be noticed that the terms of the form $\sum_{n=0}^{N-1} s[n-k]\,s[n-i]$ are those of the short term covariance of *s[n]*.

Finally, the covariance function of *s[n]* is defined by:

$$C(i,0) = \sum_{k=1}^{p} a_k C(k,i) \qquad (3.14)$$

Where, $C(k,i) = \sum_{n=0}^{N-1} s[n-k]\,s[n-i]$ *and* $C(i,0) = \sum_{n=0}^{N-1} s[n-i]\,s[n]$

Substituting the covariance function Eq. (3.14), into Eq. (3.13), the obtained system of equations can be explicitly stated by enumerating the equations for each value of *j* as:

$$C\,a = c \qquad (3.15)$$

While the expanded structure of the system of equations can be represented in normal matrix form as follows:

$$\begin{bmatrix} C(1,1) & C(1,2) & C(1,3) & \dots & C(1,p) \\ C(2,1) & C(2,2) & C(2,3) & \dots & C(2,p) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ C(p,1) & C(p,2) & C(p,3) & \dots & C(p,p) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} c(1) \\ c(2) \\ \vdots \\ c(p) \end{bmatrix} \qquad (3.16)$$

Evidently the covariance matrix in Eq. (3.16) is also symmetrical about the main diagonal since Eq (3.14) illustrates that *C(k,i)=C(i,k)*. However, it does not have a Toeplitz structure. Though, it can be solved by using the well known Cholesky decomposition method [80]. Figure (3.5) shows the structure algorithm of the Cholesky decomposition method [81]. Amongst the characteristic of the covariance method, it has improved resolution for short data records (more accurate estimates), and is able to extract frequencies from data showing periodicity.

```
Input: Predictor order $P$, Covariance coefficients $C(1)...C(p)$
Output: LP coefficients $a_i=a_1....a_p$
For $k=1$ to $p-1$ do
    $C(k,k)=sqrt(C(k,k))$
    $C(k+1:p, k)=C(k+1:p,k)/C(k,k)$
    For $j=K+1$ to $p$ do
        $C(j,k+1)=C(j,k+1)- C(j,1:k)^T * C(k+1,1:k)$
    end
end
$a(p,p)=sqrt(C(p,p))$
```

Figure 3.5 Cholesky decomposition algorithm [81]

The methodology for obtaining LPC coefficients of microarray gene data samples using covariance method involves calculating the following quantities in reference to Eq. 3.12 and 3.15:

- a_{ij} A vector of coefficients in 2-Dimension: *row* of length equal to the length of gene expression data samples, for example Luke microarray dataset has dimension of 7192 gene by 72 samples and therefore the length of coefficient vector is equal 72, and *column* of length equal to $p+1$ which depends on the order of LPC model.
- E The error signal associated to the predictive samples as shown in Eq.(3.12).

3.3.3- Improved LPC coefficients analysis

It is well known that the resultant LPC coefficients a_i are not suitable for coding and have sensitive quantisation properties that are not compatible for microarray clustering prosess. Furthermore, stability checks are complicated. Direct quantisation of the a_i coefficients is not advisable because small changes due to the quantisation error can cause the filter on the synthesis side to become unstable and produce large spectral errors. Thus other better quality methods of quantising the prediction coefficients with high efficiency have been formulated [82]. These include the reflection coefficients (RC) or partial autocorrelation coefficients, arc-sine reflection coefficients (ASRC) and log area ratios (LAR). Since it has been shown in the literature [77, 83] the line spectral frequency (LSF) representation is one of the best methods, this approach is used in this work and is detailed in the next section.

3.3.3.1- Line Spectral Frequency Transform

Line spectral frequency (LSF) is derived from LPC and has been introduced by Itakura [84] as an alternative representation to the LPC parameters in the frequency domain; it is also known as Line Spectrum Pair (LSP) representation. Whereas LPC parameters have a large dynamic range of values that causes inaccurate quantisation, line spectral frequencies have a well behaved dynamic range. Therefore, if interpolation is done in the LSF domain, it is easier to guarantee the stability of the resulting synthesis filter.

The LSF representation has a number of properties making it desirable for quantisation, such as a bounded range, a sequential ordering of the parameters and a simple check for the filter stability [83]. The LSF parameters also exhibit the distortion independence property, which means that any change in an LSF parameter will not produce any global effect, it will only affect the frequency spectrum close to it. Thus, the LSF parameters at higher frequencies can be represented with fewer quantise levels. Besides that, clustering of LSFs based on characterizing a frequency of a given spectrum of data depends on the closeness of the corresponding LSFs. Consequently, due to the spectrum sensitivity of LSFs, which are localized, the individual LSFs can be quantised independently without significant loss of quantisation distortion from one spectral region to another. Therefore, LSF parameters are more practical for quantisation than LPC coefficients.

In order to define the LSF, let's assume the transfer function *H(z)* of LP model is:

$$H(z) = \frac{1}{A(z)}$$

Where the *H(z)* is referred to as an all-pole model and filter *A(z)* is known as the inverse filter of *H(z)*, defined as

$$A(z) = 1 - \sum_{k=1}^{p} a_k\ z^{-k} \tag{3.17}$$

And $A(z)=1+a_1\ z^{-1}\ +.....+\ a_p\ z^{-p}$ is the inverse filter polynomial where p is the predictor order and a_i is the i^{th} predictor coefficient of the filter. It is used to construct two polynomials formed from $A(z)$ and its time reversed system function $A(z^{-1})$:

$$P(z) = A(z) + z^{-(p+1)}\, A(\,z^{-1})$$

$$Q(z) = A(z) - z^{-(p+1)}\, A(\,z^{-1}) \tag{3.18}$$

Factoring the above equation, we get:

$$P(z) = (1 - z^{-1}) \prod_{k=2,4,..p}(1 - 2\cos w_k\ z^{-1} +\ z^{-2})$$

$$Q(z) = (1 + z^{-1}) \prod_{k=1,3,..p-1}(1 - 2\cos w_k\ z^{-1} +\ z^{-2}) \tag{3.19}$$

where $\{w_k\}_{k=1}^{p}$, are called the LSP parameters. By using these two polynomials in Eq. (3.18), the zeros of $A(z)$ are mapped onto the unit circle. If $A(z)$ is a minimum-phase system, it means that all roots of $P(z)$ and $Q(z)$ lie inside the unit circle and interlace with each other. Therefore, the transformation from LPC coefficients to the LSF parameters is reversible and $A(z)$ can be obtained from Eq. (3.19) as follows:

$$A(z) = \frac{1}{2}\,[P(z) + Q(z)] \tag{3.20}$$

The roots of $P(z)$ and $Q(z)$ can be expressed in terms of angular frequencies ω which are located between 0 and π known as LSF. The equivalent to the frequency response function form is:

$$A(e^{jw}) = \frac{1}{2}\,[P(e^{jw}) + Q(e^{jw})]$$

It is shown that the LSFs are the phases of the zeros of $P(z)$ and $Q(z)$, i.e. the LSFs are the zeros of $P(e^{jw}\ )$ and $Q(e^{jw})$. Therefore, if a pair of LSFs is very close at w_0, $P(e^{jw})+Q(e^{jw}\ )$ will be very close to zero resulting in a peak around w_0 in the amplitude frequency response curve. On the contrary, if two LSFs are far from each other, the amplitude frequency response curve will be located around the two LSFs[84].

LSFs can be denoted by: $(l_1,\ l_2,\ l_3,\ ....l_p)^T$ and its parameters satisfy the following ordering property: $0=w_0<w_1<w_2<...<w_{p+1}\ =\pi$. Thus, the stability of LPC can be ensured by quantising the LPC information in LSF domain.

Since the LSF representation is a frequency domain representation, it can be used to exploit certain properties of the gene expression data samples. The magnitude of the power spectrum depends on the spacing of the LSF parameters. Closely positioned LSF parameters correspond to the peaks of the spectrum, and widely positioned LSF parameters correspond to the spectrum valleys. Since the power spectrum information is more important to the gene expression samples, finer quantisation of the LSF parameters in these regions is desired. This can be achieved by finer quantisation of closely positioned LSF parameters.

3.3.3.2- Vector Quantisation of LSF coefficients

In this section we describe the vector quantisation issues of the LSF transformation used in the clustering approach. The different components of LSF parameter vectors have different spectral significances. Hence the vector quantisation VQ method is used to allocate different spectral values to individual components.

In principle, the VQ process uses a 'nearest neighbour' approach in the computational process, i.e. the vector z under consideration is mapped to the code vector q_m , C is the cluster to which the vector is classified, and d is a suitable minimum distortion measure between the vectors, if:

$$C = arg\ min_i\ (d(z, q_i)) \qquad (3.21)$$

The distance df between consecutive LSF vectors can be calculated according to the following expression:

$$df(LF_i, LF_k) = \sum_{j=1}^{p}[w_j\ (lf_{ij} - lf_{kj})]^2 \qquad \text{and} \qquad w_j = P(lf_j) \qquad (3.22)$$

where LF_i and LF_k are vectors of LSFs, lf_{ij} is the j^{th} frequency of LF_i and w_j is the power spectral distortion measure. However, generally the gain-normalized log spectral distortion is used since it is widely accepted as a quality measure of coded speech spectra. It evaluates the similarity of two auto-regressive envelopes. It is expressed in the frequency domain by the following equation [76]:

$$d(z, q_i) = \int_{-\pi}^{\pi}(log\ P_z(w) - log\ P_{qi}(w))^2 \frac{dw}{2\pi} \qquad (3.23)$$

where $P(w)$ is the auto-regressive envelope that is defined as:

$$P(w) = \frac{1}{\left|1+\sum_{k=1}^{p} a_k e^{-jwk}\right|^2} \tag{3.24}$$

The design of codebooks is usually accomplished by an iterative algorithm called the Lloyd algorithm as described earlier in chapter 2 (Section 2.5). This algorithm generates a set of representative vectors of the source data and optimizes the codebook using the distortion measure method. Finally, once the codebook has been defined, LPC coefficient vectors of *s* are extracted, compared to all codewords of *C* and mapped to a single codeword that represents the different genes mapped on the tested microarray data. The structure of the encoder LPC-VQ is shown in Figure (3.6). The input data is x, the p^{th} order predictor parameters are a_p and the VQ parameters are â. The distortion inherent in the model is $d(x, a_p)$, and the resulting distortion from quantising a_p is $d(a_p$, â). The overall distortion is $d(x, \hat{a})=d(x,a_p)+d(a_p+ \hat{a})$

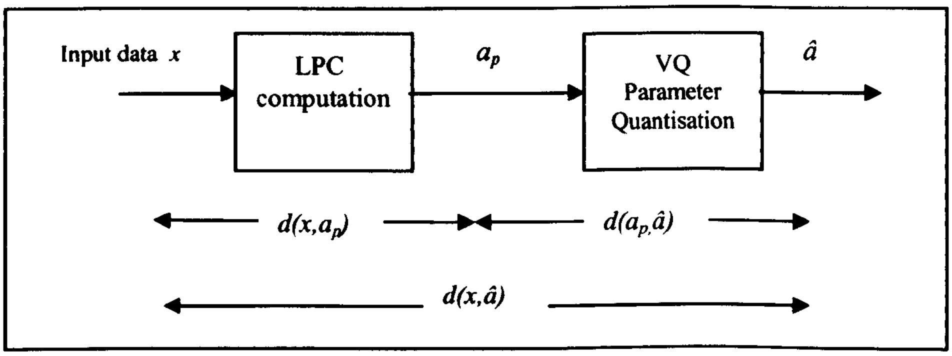


Figure 3.6 Scheme of distortion measure used in the LPC-VQ analysis

3.4- Microarray LPC (miLPC) clustering analysis

In this section, we introduce the LPC microarray clustering method (miLPC) for clustering microarray gene expression samples. A block diagram of miLPC clustering is shown in Figure (3.7). Microarray gene expression samples are the input to the system and a clustering decision is obtained from the system output, miLPC has the following processing blocks:

a- Microarray Normalization

If one considers a gene expression profile, denoted by the vector, $V=[v^1,v^2,v^3,\ldots v^n]$, measured for *n* samples, rescaling is an essential preprocessing step. It is commonly done by replacing every expression level v^i in *V* by:

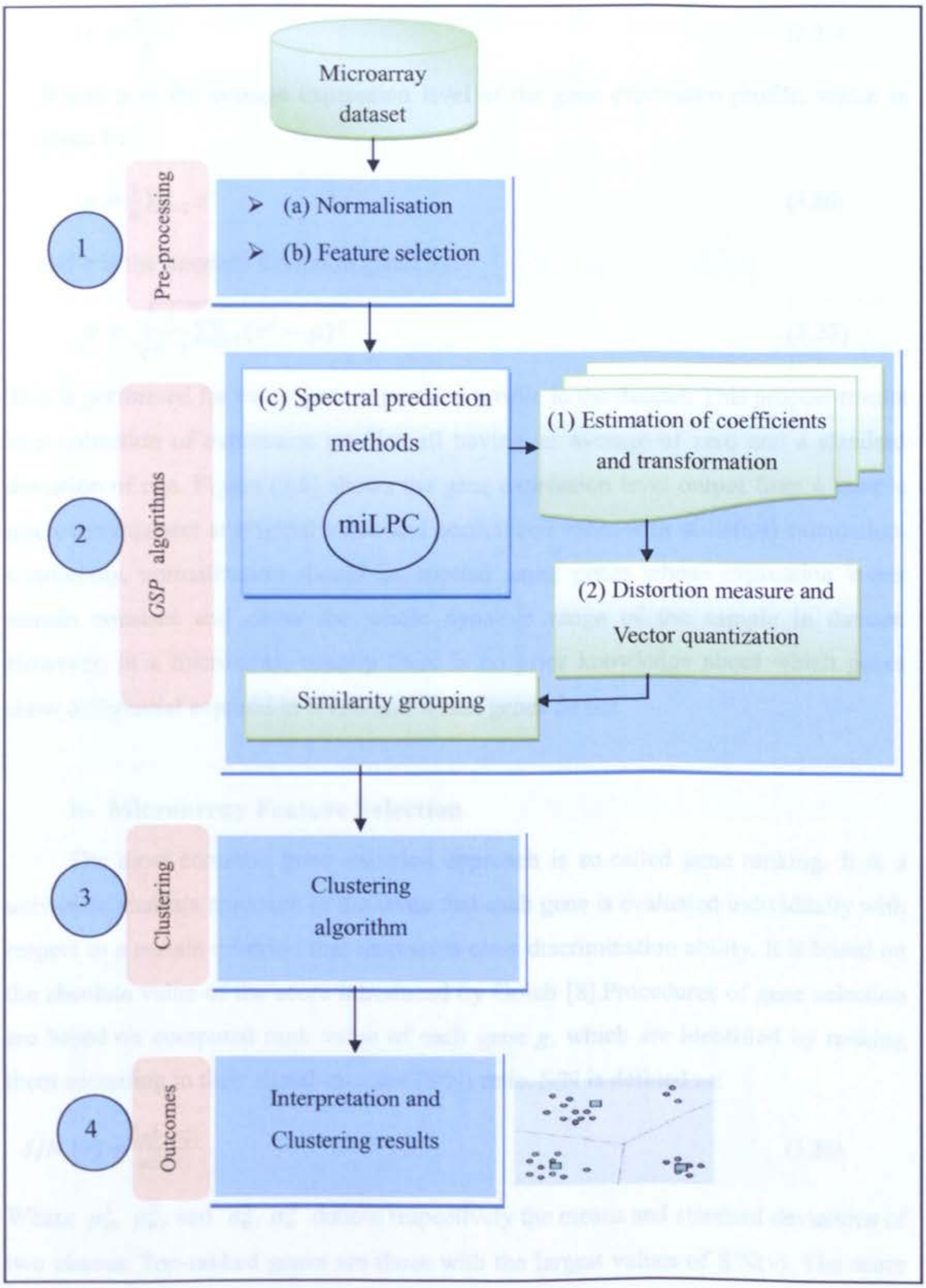

Microarray dataset
1
Pre-processing
(a) Normalisation
(b) Feature selection
2
GSP algorithms
(c) Spectral prediction methods
miLPC
(1) Estimation of coefficients and transformation
(2) Distortion measure and Vector quantization
Similarity grouping
3
Clustering
Clustering algorithm
4
Outcomes
Interpretation and Clustering results

Figure 3.7 miLPC clustering method

$$v = \frac{v^i - \mu}{\sigma} \tag{3.25}$$

Where μ is the average expression level of the gene expression profile, which is given by:

$$\mu = \frac{1}{n}\sum_{i=1}^{n} v^i \tag{3.26}$$

and σ is the standard deviation given by:

$$\sigma = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(v^i - \mu)^2} \tag{3.27}$$

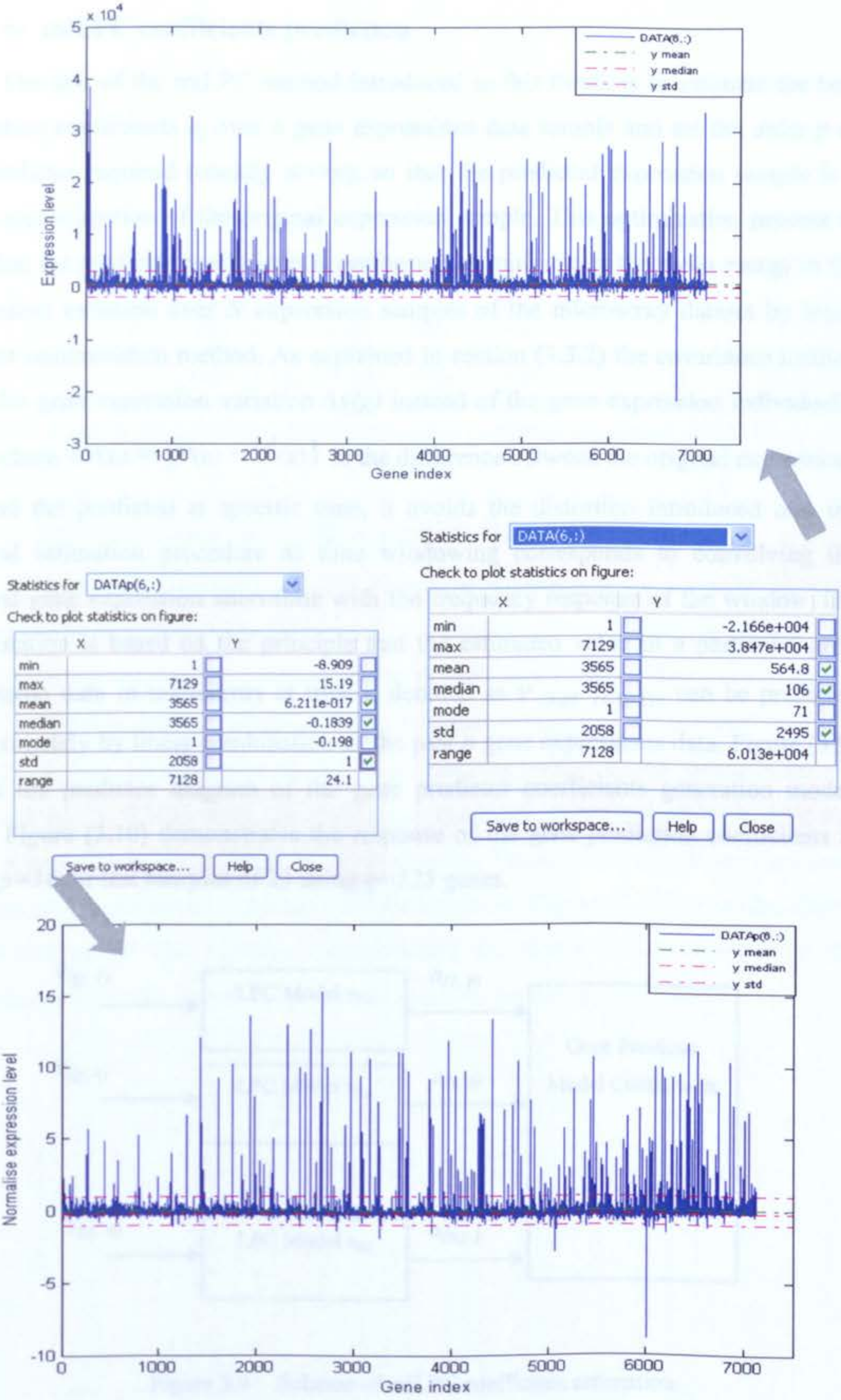
This is performed for every gene expression profile in the dataset. This process results in a collection of expression profiles all having an average of zero and a standard deviation of one. Figure (3.8) shows the gene expression level output from a sample microarray dataset at original value and normalized value with statistical calculation. Commonly, normalization should be applied using genes whose expression levels remain constant and cover the whole dynamic range of the sample in dataset. However, in a microarray, usually there is no prior knowledge about which genes show differential expression levels and which genes do not.

b- Microarray Feature Selection

The most common gene selection approach is so-called gene ranking. It is a univariate analysis approach in the sense that each gene is evaluated individually with respect to a certain criterion that represents class discrimination ability. It is based on the absolute value of the score introduced by Golub [8].Procedures of gene selection are based on computed rank value of each gene g, which are identified by ranking them according to their signal-to-noise (S/N) ratio. S/N is defined as:

$$S/N\,(v) = \frac{|\mu_v^1 - \mu_v^2|}{\sigma_v^1 + \sigma_v^2} \tag{3.28}$$

Where μ_v^1, μ_v^2, and σ_v^1, σ_v^2 denote respectively the means and standard deviations of two classes. Top-ranked genes are those with the largest values of S/N(ν). The score which is calculated for all genes individually and genes with the best scores are selected after sorting them is explained in chapter 6.



Statistics for DATAp(6,:)

Check to plot statistics on figure:

	X		Y	
min	1	□	-8.909	□
max	7129	□	15.19	□
mean	3565	□	6.211e-017	☑
median	3565	□	-0.1839	☑
mode	1	□	-0.198	□
std	2058	□	1	☑
range	7128		24.1	

Statistics for DATA(6,:)

Check to plot statistics on figure:

	X		Y	
min	1	□	-2.166e+004	□
max	7129	□	3.847e+004	□
mean	3565	□	564.8	☑
median	3565	□	106	☑
mode	1	□	71	□
std	2058	□	2495	☑
range	7128		6.013e+004	

Figure 3.8 Normalization of gene expression sample 6 of leukaemia dataset.

c- miLPC coefficients prediction

The aim of the miLPC method introduced in this thesis is to estimate the best prediction coefficients a_j over n gene expressions data sample and set the order p of the predictor required (usually n>>p), so that the predicted expression sample is a good approximation of the original expression sample. This optimization process to calculate the predictor coefficients is performed by minimizing the mean energy in the expression variation over N expression samples of the microarray dataset by least-squares minimization method. As explained in section (3.3.2) the covariance method used the gene expression variation $\Delta v(g)$ instead of the gene expression individually $v(g)$, where $\Delta v(g) = \{v(g) - v'(g)\}$ is the difference between the original expressions $v_{(g)}$ and the predicted at specific time, it avoids the distortion introduced into the spectral estimation procedure as time windowing corresponds to convolving the original gene expression short-time with the frequency response of the window The computation is based on the principle that the estimated value of a particular gene expression data in microarray at time t, denoted as $v`_{(x,g)}$ $\{x \in t\}$, can be predicted approximately by linear combination of the past p gene expressions data. Figure (3.9) shows the predictor diagram of the gene predictor coefficients generation model, while Figure (3.10) demonstrates the response of the gene prediction coefficients at order $p=34$ for test samples of 23 using $g=125$ genes.

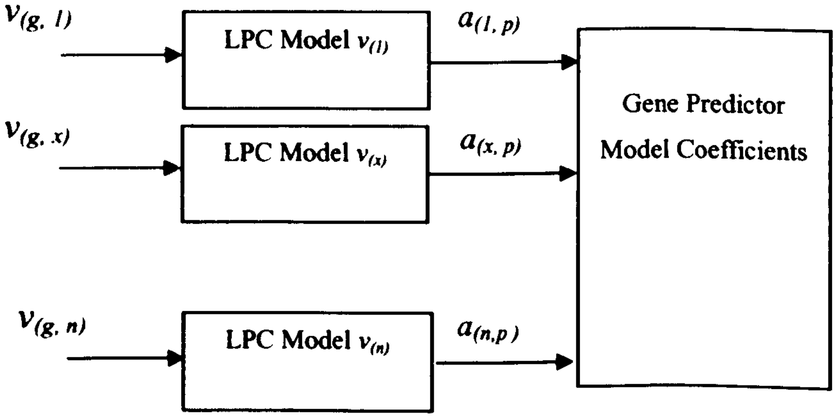


Figure 3.9 Scheme of miLPC coefficient estimation

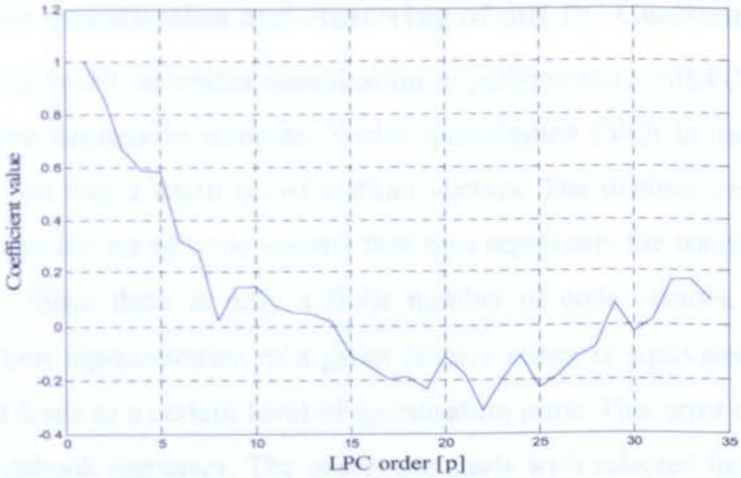


Figure 3.10 miLPC coefficient estimation (*p=34*) to expression sample 23

1. Transformation of Coefficient prediction

The process of direct quantisation of the coefficients a_j is not advisable because of their relatively large dynamic range and possible filter instability problem as explained earlier in Section 3.3.3. In this work we chose the Line Spectral Frequency (LSF) representation to produce Gene Expression Spectral Frequency (GESF), because it has been shown to be a particularly efficient for quantisation of information[83]. It also does not distort the spectrum, varies smoothly in time and offers a better coding in relation to spectral peaks. These GESF coefficients are used subsequently to determine distortion between samples. Figure (3.11) shows the GESF transformation of the response demonstrated in Figure (3.10) concerning the prediction expression sample 23 estimated based on prediction order p=34

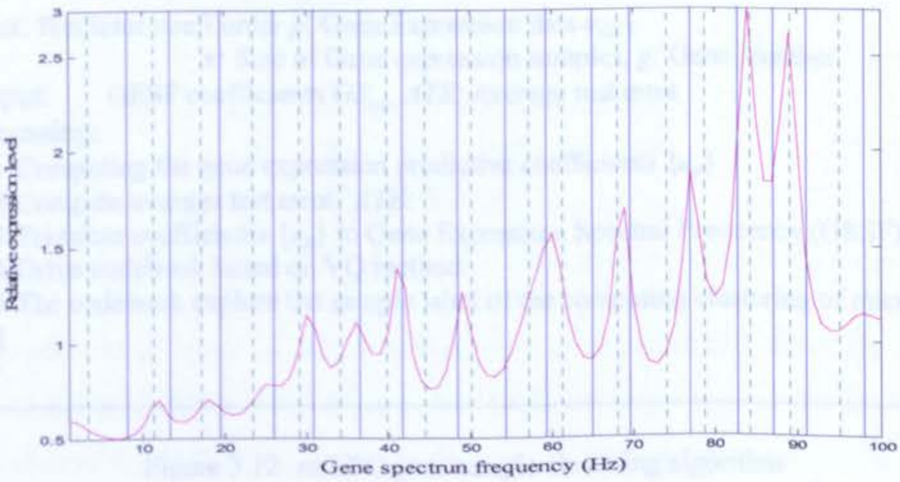


Figure 3.11 GESF concerning the prediction expression sample 23 with respect to prediction order p=34

2. Vector quantisation and clustering of miLPC Coefficients

Clustering based on vector quantisation is performed to miLPC coefficients in microarray gene expression samples. Vector quantisation (VQ) is used to convert a feature vector set into a small set of distinct vectors. The distinct vectors are called code vectors and the set of code vectors that best represents the training set is called the codebook. Since there is only a finite number of code vectors, the process of choosing the best representation of a given feature vector is equivalent to quantising the vector and leads to a certain level of quantisation error. This error decreases as the size of the codebook increases. The procedure starts with selected initial centres and next, all the objects are classified into the appropriate clusters using a minimum distance function. A distortion measure for the current cluster arrangement (e.g., Mean Square Error) is calculated and each cluster centre is updated to be the average value of the feature vectors corresponding to objects within the cluster. At this stage, the objects are regrouped, new centres are calculated, and the distortion measure is updated. The process of clustering and updating centres and distortion is repeated until the normalized change in distortion is below some threshold in the iteration.

Figure (3.12) describes the algorithm of miLPC gene sample clustering. The VQ codebook can be used as a model in pattern recognition as explained earlier. The key point of VQ modelling is to derive an optimal codebook which is commonly achieved by using a clustering technique.

```
Input: Predictor coeff order $p$, Gene Expression data $v_{x,g}$ ,
                    $x$: Size of Gene expression samples, $g$: Gene Number
Output:    GESF coefficients $GS_{x,p}$. $ATE$: Average test error
Processing:
    1-Computing the gene expression predictive coefficients $\{a_p\}$
    2-Compute average test error, $ATE$:
    3-Translate coefficients $\{a_p\}$ to Gene Expression Spectral Frequency (GESF)
    4-Drive codebook based on VQ method
    5-The codebook explore the sample label of the computing clustering of micrarray
End
```

Figure 3.12 miLPC gene sample clustering algorithm

3.5- Performance analysis of miLPC

In this section we describe the performance analysis of the miLPC method and application on test data.

3.5.1- miLPC evaluation criteria

In order to apply the miLPC method on sample microarray data, some basic parameters must be chosen. Variation of these parameters causes varying performance. To obtain useful results with linear prediction and to apply it successfully, it is necessary to understand the relationship and the effect of the changes in parameters on the clustering process. The main influencing parameter effects on LPC analysis performance is based on the chosen order *p* of the linear predictive which attempts to be able to achieve reasonable model. The goal of this experiment is to study the prediction order with reasonable computation that gives minimum error between the original gene sample signal and the prediction signal. Figure (3.13) shows the comparative results achieved by several order selections implemented to calculate the Mean Square Error (MSE) of the predictive coefficients with respect to genes involved in the process.

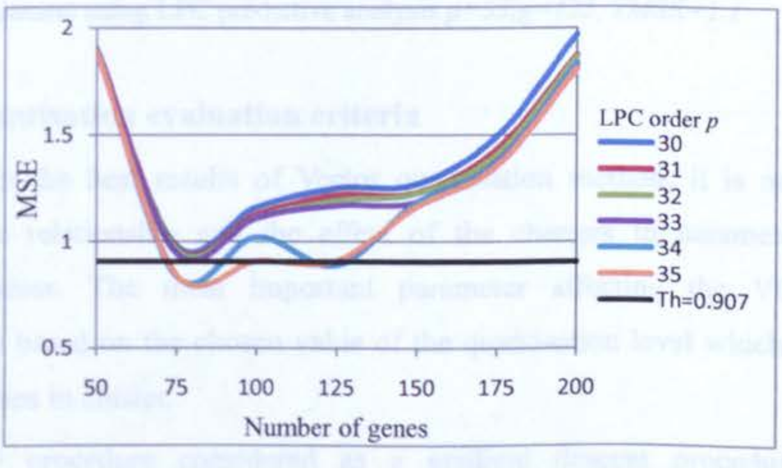


Figure 3.13 MSE of miLPC analyses with different order (*p*) and selected genes

The amount of error by which the predictive signal differs from the original sample variations is illustrated in Figure (3.14). We will refer here to the *TMSE* as the average value of the MSE for a given set of samples. For example, the *TMSE=1.1* for the results shown in the same figure.

The preliminary tests show that the best order p selected is dependent on the Minimum residual error of the LPC predictive analysis, from Figure (3.13), it shows that a threshold level of error which is equal $Th=0.907$ gives accurate clustering analysis. The total error estimated pertaining to each sample in the microarray is shown in Figure (3.14), while the overall average error is equal to $TMSE=1.1$. Further experimental result and discussion present in chapter 6.

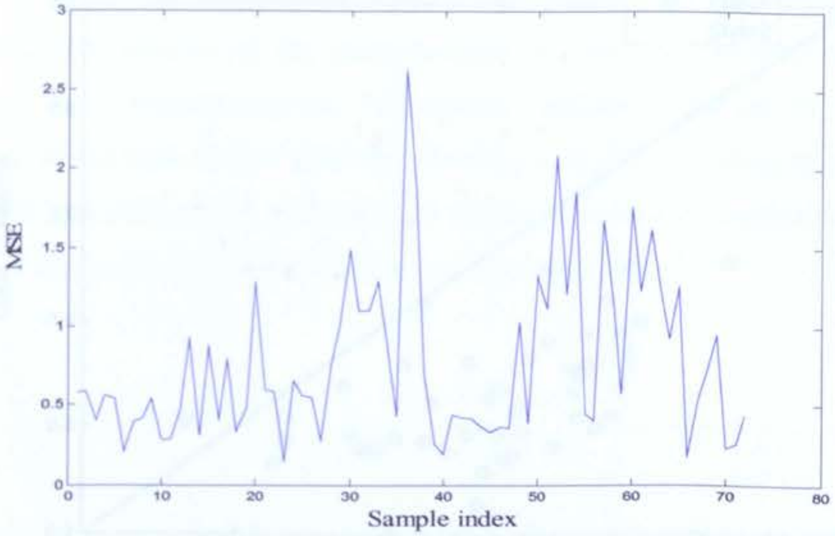


Figure3.14 Total MSE estimated pertaining to each sample in the leukaemia microarray dataset using LPC predictive analysis $p=33, g=125$, $TMSE=1.1$

3.5.2- Quantisation evaluation criteria

To obtain the best results of Vector quantisation method, it is necessary to understand the relationship and the effect of the changes in parameters on the clustering process. The most important parameter affecting the VQ analysis performance is based on the chosen value of the quantisation level which represents number of classes in cluster.

The VQ procedure considered as a gradient descent procedure for the approximation of a best quantised level relative to the partition region which is determined by minimizing the distortion measure. The number of iterations required to achieve reasonable model is dependent on the dimensionality of the microarray feature, and the attributes of the coefficients distribution estimated.

In this work, we set this quantisation level equal to two, which represent the number of clusters in microarray. Figure (3.15) shows the clustering result between two classes in Leukaemia microarray. It shows there is an error in sample 36 where it should be lie in class 1 segment when we used *g=125* genes and LPC order *p=33* that will obtain *MSE=2.6*. VQ allows different cell shapes, like hexagons, to fill the region of expression samples, the set of Voronoi regions partition the entire space of clustering is shown in Figure (3.16). It illustrates the two classes clustering of microarray samples as represented in Figure (3.15).

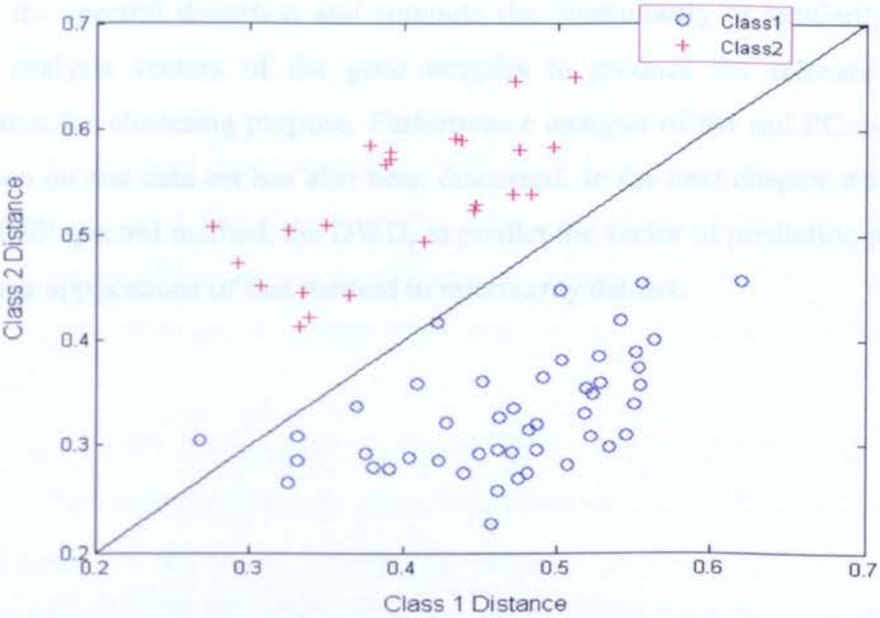


Figure 3.15 miLPC Clustering Leukaemia dataset

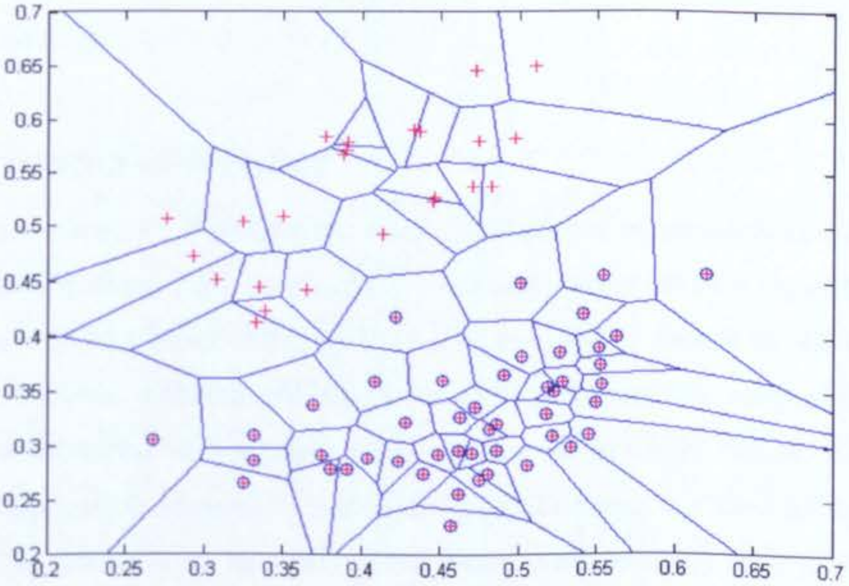


Figure 3.16 Voronoi clustering of Leukaemia dataset

3.6- Conclusion

In this chapter we have explained the LPC approach in data clustering and described the characteristic of the microarray gene expression pattern as a signal. We introduced a new method, the miLPC, that modifies the standard LPC approach with VQ tailored for enhanced microarray clustering. miLPC implies a transformation from the predictive LPC coefficients to the GESF and then the VQ approach is applied to measure the spectral distortion and compute the dissimilarity or similarity between spectral analysis vectors of the gene samples to produce the relevant index of quantisation for clustering purpose. Performance analysis of the miLPC method and application on test data set has also been discussed. In the next chapter we introduce another DSP spectral method, the DWD, to predict the vector of prediction parameters and discuss applications of that method to microarray dataset.

CHAPTER 4

Wavelet for microarray clustering

Discrete Wavelet Decomposition (DWD) is a well-known technique in the digital signal processing area and is used extensively in biomedical signal processing. In general, the wavelet technique divides up data, functions, or operators into different frequency components and then deals with each component with a resolution matched to its scale.

In this chapter the application of DWD to microarray sample clustering is explained. The chapter outlines the basic DWD characteristics of the microarray gene sample signals and illustrates different wavelet families and their application in the clustering process. The VQ approach is applied to measure spectral distortion and to compute the dissimilarity or similarity between spectral analysis vectors of gene samples to produce the index of quantisation in respect to the clustering of the microarray samples.

4.1- Properties of Wavelets

In this section, we highlight the relevant properties of wavelets in microarray gene expression analysis [20]. In general, a wavelet allows to obtain a view of a signal at different resolutions which differ by a factor of two, and to encode the difference of information between different resolutions as orthogonal wavelet coefficients. Each coefficient is computed with a single scalar product of the signal and the wavelet. A wavelet transformation converts data from an original domain to a wavelet domain by expanding the raw data in an orthonormal basis. Each wavelet basis contains an infinite number of wavelets that are generated by dilation and translation of a scaling function (father wavelet) and the wavelet function (mother wavelet). An inverse wavelet transformation converts data back from the wavelet domain to the original domain. Wavelets have significant properties for genomic expression data analysis such as [85]:

- **Minimize computation complexity** of transformation with linear time and space complexity in addition to the symmetry of scaling, in concerning the variation involved with expression data samples
- **Vanishing moments** reflect the oscillatory nature of wavelets which could characterize differences or details in the genome data samples profile. They can lead to de-noising and dimensionality reduction.
- **The multi resolution** decomposition structure of scaling and wide variety of basis functions, leads to hierarchical representations and manipulates expression samples as objects.
- **De-correlated Coefficients** of the expression model that gave their ability to reduce temporal correlation smaller than other in a process

4.2- Application of DWD in gene expression

The underlying motivation for the DWD analysis of microarray clustering is that a set of wavelet bases can represent accurately the localized features contained in microarray data without losing other features.

In this thesis, multilevel wavelet decomposition is performed to represent gene profile into approximations and details to extract the spectral features of microarray data for enhanced clustering. Furthermore, the method is used to characterise multiple expression sample positions and their length scale.

Wavelets are groups of mathematical equations which can be applied on data that have variable frequency components, allowing the study of each of these components into their scales fields [86]. In this application, DWD allows the decomposition of an input signal as expression samples onto a set of basis functions and its analysis it by transforming it to time-frequency domain.

4.2.1- Wavelet background

Wavelet analysis has the capability of extracting many useful aspects in data like trends, breakdown points, discontinuities in higher derivatives, and self-similarity. It allows a sample series to be viewed in different multiple resolutions by decomposing data at different frequencies of decompose data without significant degradation unlike traditional statistical techniques [87].

The breakdown process of a samples signal onto a set of basis functions is achieved by dilations, contractions, scaling, and shifting. This partition provides resolution optimality in both time and frequency domains and does not require a stationary signal. It is based on two major sub operations: scaling which captures the gene expression samples information by successive low pass/ high pass filtering and down sampling, whilst the translation sub operation captures the information at different locations. It decomposes expression samples data into several groups of coefficients which contain information regarding the sampled signal at different scales. Coarse scale coefficients capture gross and global features of the signal while fine scale coefficients contain local detail.

Since microarray data represent the activity of genes across different samples, expression of a gene may display a specific range of frequencies [88]. Therefore, wavelet decomposition is a method that can be applied to convert spatial expression samples data into the frequency domain. The method has high degree of spatial localization, but the degree of concentration depends on the frequency content of the wavelet function [60]. Since high frequency wavelets are narrower than lower frequency ones, wavelets can be seen as a set of adaptive base functions.

The basic DWD algorithm is shown in Figure (4.1). The method starts by applying recursively two convolution functions, a low and high pass filters on the given data signal *S*. Each function produces an output stream that is half the length of the original input in a specific resolution level. As a result, two sets of coefficients are calculated: the *cA(n)* coefficients are generated by the low pass filter and the *cD(n)* coefficients are produced by the high pass filter.

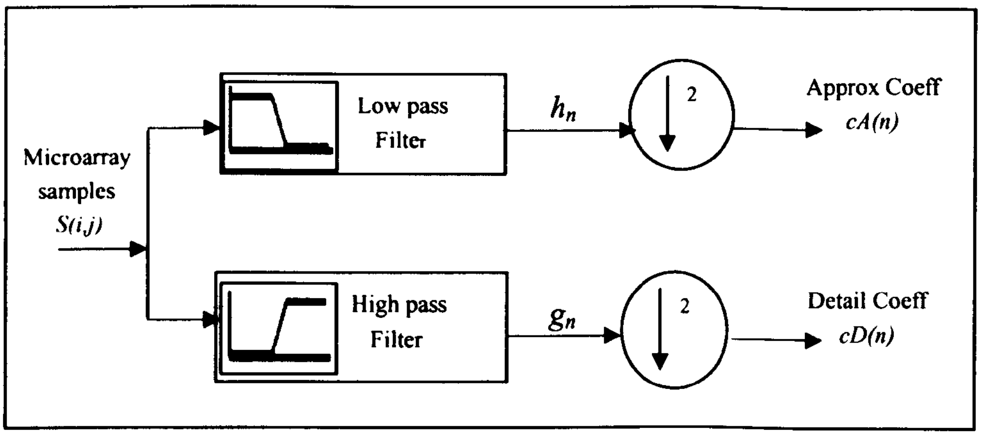


Figure 4.1 Wavelet decomposition analysis

Concerning wavelet analysis for gene expression data, a gene expression profile can be represented as a sum of wavelets at different time shifts and scales using DWD. The DWD is capable of extracting the local features by separating the components of gene expression profiles in both time and scale. In DWD, a time varying function $f(t) \in L^2(R)$ can be expressed in terms of $s(t)$ and $\psi(t)$. The mathematical formulation can be summarized as follows:

$$f_{i,j}(t) = \sum_n a_0(n)s(t-n) + \sum_n \sum_{j=1} d_j(n)\,\psi_{n,j} \qquad (4.1)$$

Where $L^2(R)$ is the function space and $s(t)$, $\psi_{i,j}(t)$, a_0 and d_j represent the scaling function, wavelet function, scaling coefficients (approximation coefficients) at scale 0, and detail coefficients at scale j respectively. The variable n is the translation coefficient for the localization of gene expression data. The scales denote the different (low to high) scale bands. $\psi_{i,j}(t)$ is the wavelet basis functions defined by i and j parameters. They are derived from (contracted) and shifted versions of a function $\psi_{i,j}(t)$, called mother wavelet, defined as

$$\psi_{i,j}(t) = 2^{m/2}\psi(2^m t - n) \qquad (4.2)$$

Eq.(4.2) is used to obtain an orthonormal wavelet basis. Parameter m stretches the mother wavelet leading to either a narrower or broader new function. Parameter n translates the mother wavelet along t space. Therefore, all the basis functions $\psi_{i,j}(t)$ have the same profile, but dilated and translated according to parameters m and n respectively [87].

The first step of the wavelet decomposition procedure, produces two sets of coefficients: approximation coefficients (scaling coefficients) a_1, and detail coefficients (wavelet coefficients) d_1. These coefficients are computed by convolving the signal with the low-pass filter for approximation, and with the high-pass filter for detail. The convolved coefficients are down sampled by keeping the even indexed elements. Then the approximation coefficients a_1 are split into two parts by using the same algorithm and are replaced by a_2 and d_2, and so on. This decomposition process is repeated until the required level is reached.

As shown in Figure (4.1) a coarser approximation of microarray samples S can be calculated by iteratively convoluting with the low pass filter h_j and down sampling the signal by two. Therefore, a set of discrete approximations S_j, $1<j<z$ (where z is the maximum possible scale) is produced. g_j denotes the difference between S_j and S_{j-1} and is called the detail signal at the scale j. The wavelet representation of discrete microarray gene expression samples S can therefore be computed by successively decomposing S_o into a_j and d_j. This representation provides information about microarray gene expression sample approximation coefficients and detail coefficients at different scales. Detail and approximation at level j are expressed respectively by Eq. (4.3) and Eq.(4.4) as follows:

$$D_{j+1}(n) = \sum_t a_j(t)\, g(2n-t) \tag{4.3}$$

$$A_{j+1}(n) = \sum_t a_j(t)\, h(2n-t) \tag{4.4}$$

where $h(2n-t)$ and $g(2n-t)$ are the low-pass filters and high-pass filters. The coefficient vectors are produced by down sampling and are only half the length of signal or the coefficient vector at the previous level. Conversely, approximations and details are constructed inverting the decomposition step by inserting zeros and convolving the approximation and detail coefficients with the reconstruction filters.

The inverse discrete wavelet transform is given by the reconstruction formula:

$$s(t) = \sum_i \sum_j f_{i,j}(t)\ \psi_{i,j}(t) \tag{4.5}$$

And similarly for the recombination steps:

$$a_{j-1}(n) = \sum_t a_j(t)\, h(2n-t) + \sum_t d_j(t)\, g(2n-t) \tag{4.6}$$

Then $a_{j-1} = Ha_j + Gd_j$

The main concept of this decomposition is to start from a scale-oriented decomposition, and then to analyse the obtained signals on frequency subbands. Using these decomposition coefficients, microarray data clustering can be achieved by measuring similarities between datasets using the vector quantisation method.

4.2.2- Wavelet processing blocks

The basic DWD comprises four processing steps:

1. **Wavelet decomposition**: The samples series from the original dataset are decomposed using a series of wavelets. The convolution between the samples signal (the starting segment) and the filters *G* and *H* are calculated as shown in Eqs. (4.3 & 4.4). Then, the samples series is shifted one data point to the left and the previous calculation is repeated. This is done until the whole signal is covered. From this shifting we obtain a set of coefficients that represent how the wavelet function matches the signal in time.
2. **Coefficient selection**: Both sets of coefficients are obtained by the convolution of the signal with the filters *G* and *H*. In this work we have chosen Daubechies wavelets, which are, according to investigation [87], the best wavelets for this application.
3. **Signal reconstruction**: The individual series are reconstructed using the estimated coefficients. Reconstruction is done with convolution of the detail signals and the last approximation with the inverse filters. The reconstruction procedure is started from the last approximation, where the signal is shifted one data point to the left and the previous calculation is repeated until reaching the last coefficient choose the detail level from which we do the reconstruction.
4. **Evaluation**: Error between each predictive reconstruction and the original signal is calculated. The closest reconstruction is selected. Individual reconstructed time series create the 'filtered' dataset.

In this case the signal *S* can be described in terms of the wavelet it was transformed with using the *cDlevel* and *cAlevel* coefficients. The inverse transform (reconstruction) can take place by using those coefficients and the original wavelet. Figure (4.2) shows the plot of the original signal, the coefficients and the reconstruction, after a level three transform (and the inverse transform for the reconstruction)

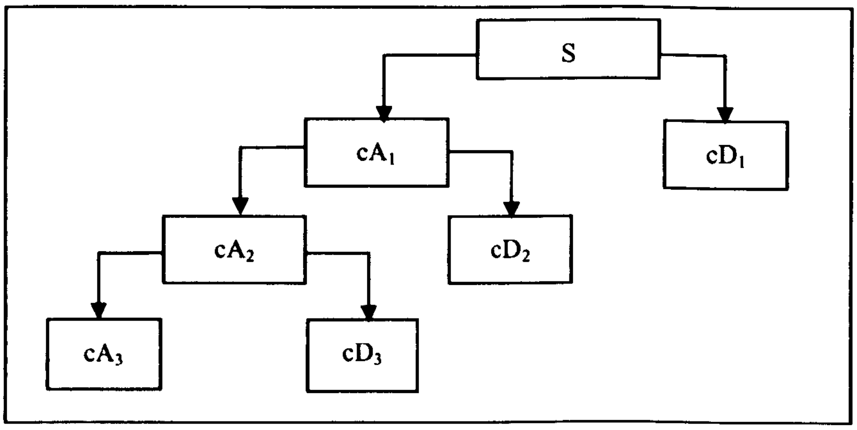


Figure 4.2 Representation of DWD levels

4.2.3- Vector Quantisation of DWD

Similar to miLPC method, the DWD is combined with the VQ for the clustering purpose. The use of vector quantisation in this process is two-fold: first it captures meaningful classes in the data, represented by their centers, and second it makes subsequent cluster decisions more robust to the inherent noise within the data[76]. The detail of the VQ method is described earlier in chapters (2 & 3).

4.3- Microarray DWD (miDWD) analysis

In this section, we briefly present some earlier work on the use of the DWD microarray clustering method (miDWD) for clustering microarray gene expression samples. The potential of wavelets is described in the work [20]. They introduced analysis based on wavelet transform for identification of microarray features and exploration of their relationship with phenotypic outcomes. The method allows decomposing gene signal into components on different length scales, even when the genome is severely distorted, providing a convenient basis for exploring their behaviour. The expression signal given by genes in clustered order could be implemented with wavelet transformed. In the work reported in [90], a hybrid analysis method to find significant genes based on wavelet analysis and Genetic algorithm (GA) was introduced. Multilevel wavelet decomposition was performed to reduce the dimensionality of microarray features by breaking gene profile into approximations and details coefficients. Approximation coefficients were reconstructed to build the approximation. Genetic algorithm is further implemented to select the optimal features from approximation coefficients. The method achieved accurate results in comparison with other statistical methods only when 15 GA at 2^{nd} level of wavelet decomposition is used. The work [86] presents a wavelet-based approach to perform cluster analysis on multidimensional datasets in comparison with other statistical methods such as, classical K-means, hierarchical clustering and the aforementioned Similarity based Clustering, SCM. Furthermore, systematic determination of cluster boundaries based on the ratio of with-in class variance and between-class variance is introduced in [91]. Moreover, in order to reduce the noise content in the expression data, they used discrete wavelet transform with a threshold value before the clustering procedure to smooth the noise. They tested three different types of mother wavelet functions: Daubechies wavelets, Haar mother wavelet and Symlet mother wavelet. They showed that Daubechies wavelets are the most appropriate and the data enhancement by wavelet transforms yielded better results for time series data which has periodicity.

The multi-resolution property of wavelet transforms inspires researchers to consider algorithms that could identify clusters at different scales. WaveCluster is a multi-resolution clustering approach for very large spatial databases that provides stable and efficient clustering [60]. Recent work [89] applied wavelet feature extraction based on multilevel wavelet decomposition analysis for microarray dataset.

In this thesis we present the miDWD method for gene sample clustering. This method is a powerful tool in the data clustering since it outperforms any unsupervised method. The main concept of miDWD is to represent the expression signal as a set of wavelet bases, which would allow to detecting the localized features, which could not be detected by statistical methods. Wavelets tend to be irregular, asymmetric and are capable of revealing aspects of data that other analysis techniques disregard. They include aspects like trends (approximation coefficients) and discontinuities in higher derivatives (detail coefficients). The analysis of the higher frequency coefficients allows detecting localized features. The higher is the number of correlated coefficients between the localized sections of two samples, the more similarity the sections have. The difference between cancer tissue samples and normal tissue samples can be measured using wavelet basis based on compactness and characteristic of wavelet function. The wavelet detail coefficients at different levels disclose the fully statistical information contained in the gene expression vector's derivatives. The preliminary results suggest that the detail coefficients at the second and third levels are perfect to characterize the features of microarray data for some microarrays.

The goal of the miDWD method is to start from scale-oriented decomposition, and then to analyses the obtained signals on frequency subbands. Using these decomposition coefficients, microarray data clustering can be achieved by measuring similarities between datasets using the vector quantisation method in order to obtain precise discrimination between features of microarray samples and perform robust clustering.

Like the miLPC method, miDWD followed the same procedure. First, we applied standard statistical normalization method to normalize microarray data. Then, we selected specific number of genes with the highest expression values. Then these data are processed using DWD algorithmic. Finally, clustering was achieved by performing vector quantisation.

4.4- Performance analysis of miDWD

In this section we describe the performance analysis of the miDWD method and its application on test data.

4.4.1- miDWD evaluation criteria

In order to apply the miDWD method on sample microarray data, some basic parameters must be chosen. However, the variation of these parameters causes variation in performance. Therefore, to obtain useful results with DWD and to apply it successfully, it is necessary to understand the relationship and the effect of the changes in parameters on the clustering process. The main influencing parameter is the chosen level of decomposition which attempts to produce a reasonable model. The goal of the following experiment is to study the relationship between the prediction order and the error either between the original gene sample signal and the reverse prediction signal or in clustering. Figure (4.3) shows wavelet based on db2 clustering errors achieved by different levels (*Lv*) with respect to the number of selected genes (*g*). Results in chapter 6 show that proper clustering of leukaemia dataset can be achieved if the minimum average error, which is estimated using MSE, is below 0.98. Therefore, DWD level2 may be suitable for robust clustering if the number of genes is high enough.

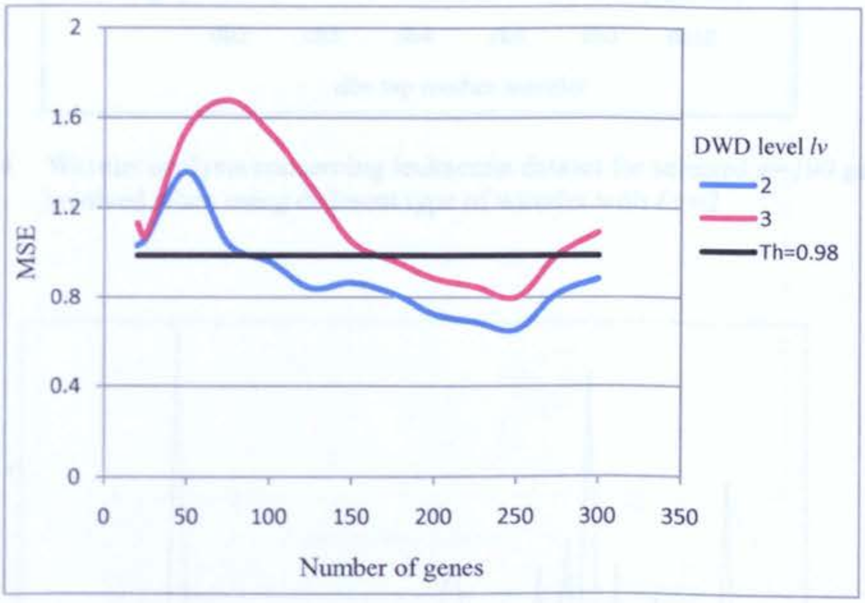


Figure 4.3 Wavelet clustering

Further analyses illustrated in Figure (4.4) show the effectiveness of wavelet type and number of taps in the mother wavelet on the MSE of the reverse prediction signal. It shows that the best frequency resolution of the wavelet filter is obtained with the db2 decomposition level that performs accurate clustering using $g=100$ and $Lv=2$.

The preliminary test as demonstrated in Figure (4.5) introduces the amount of error by which the wavelet reconstruction signal differs from the original signal concerning leukaemia dataset for selected $g=75$ genes involved when using db2 with decomposition level $lv=2$. Since the estimated error $MSE= 1.03$ is greater than the 0.98 threshold, that causes clustering errors. Here, one sample, i.e. sample 35, is incorrectly classified.

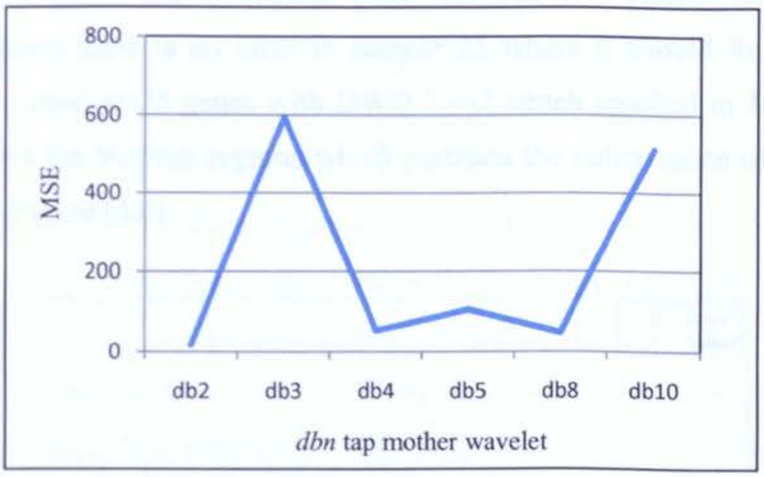


Figure 4.4 Wavelet analysis concerning leukaemia dataset for selected *g=100* genes involved when using different type of wavelet with *Lv=2*

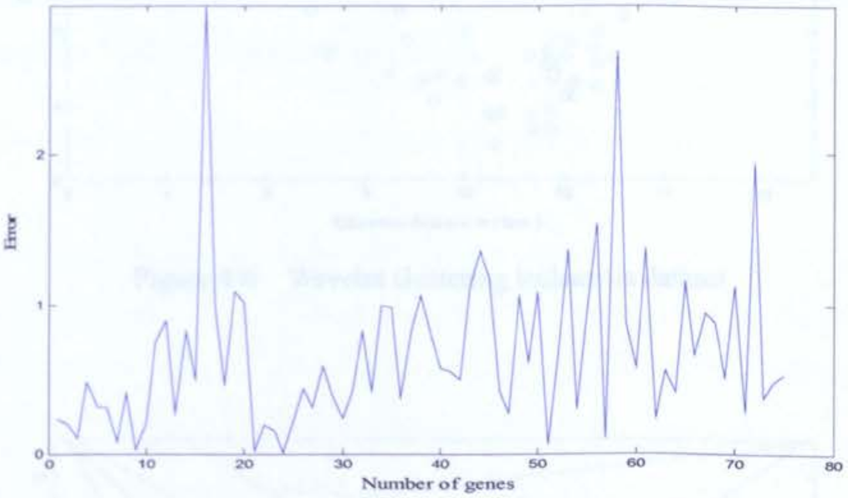


Figure 4.5 Wavelet analysis error signal concerning leukaemia dataset for selected *g=75*genes, level *Lv=2*

4.4.2- Quantisation evaluation criteria

The main influencing parameter effects on VQ analysis performance is based on the chosen value of the quantisation level which represents number of classes in cluster. In this work, the quantisation level set to two according to the microarray dataset specification.

Figure (4.6) shows the clustering result between two classes in Leukemia microarray. It shows there is an error in sample 35 where it should lie in class 1 segment; here we used *g=75* genes with DWD *Lv=2* which resulted in *MSE= 1.03*. Figure (4.7) shows the Voronoi regions which partition the entire space of clustering as represented in Figure (4.6)

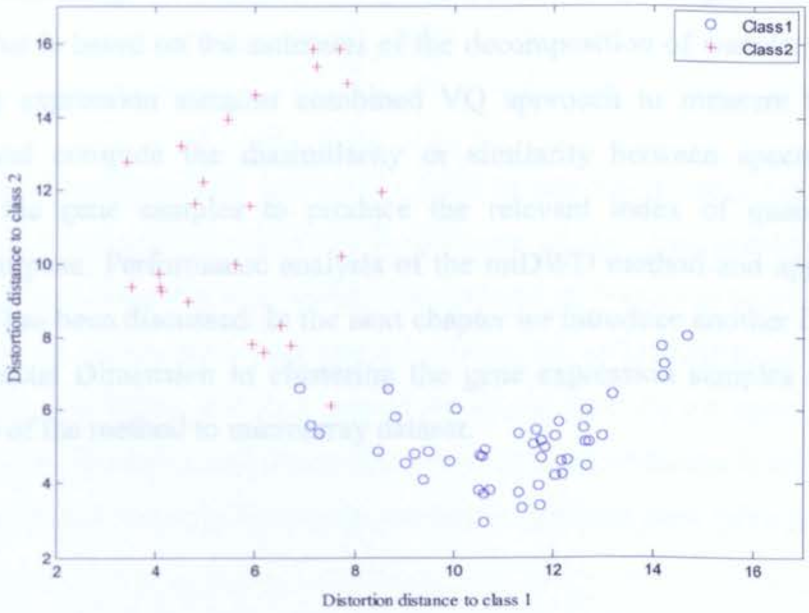


Figure 4.6 Wavelet clustering leukaemia dataset

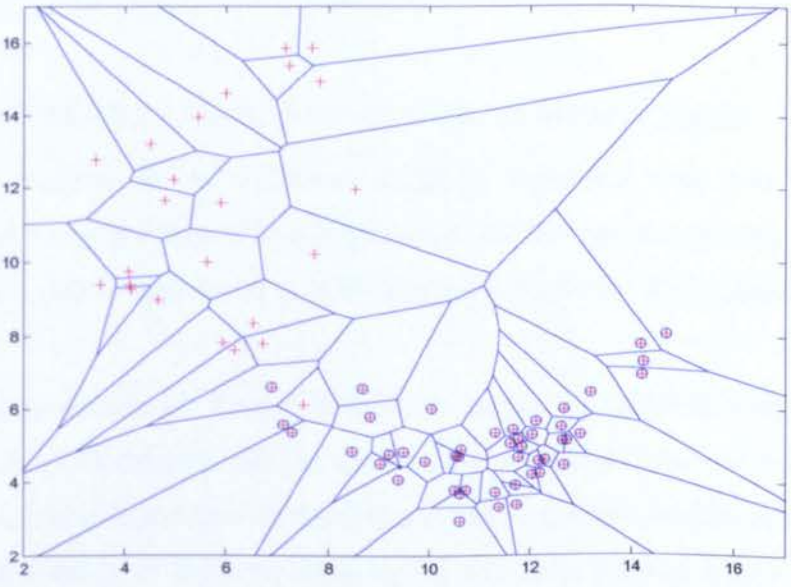


Figure 4.7 Voronoi Wavelet clustering for leukaemia dataset

4.5- Conclusion

In this chapter we have explained the DWD approach in data clustering. The chapter outlined the basic DWD characteristic of the microarray gene sample signals and illustrated different wavelet families. The wavelet technique was amalgamated with the VQ approach to provide the combined wavelet and VQ approach suitable for microarray clustering. We introduce the (miDWD) method designed for microarray clustering that is based on the estimates of the decomposition of wavelet coefficients to the gene expression samples combined VQ approach to measure the spectral distortion and compute the dissimilarity or similarity between spectral analysis vectors of the gene samples to produce the relevant index of quantisation for clustering purpose. Performance analysis of the miDWD method and application on test data set has been discussed. In the next chapter we introduce another DSP method which is Fractal Dimension to clustering the gene expression samples and discuss applications of the method to microarray dataset.

CHAPTER 5

Fractals for microarray clustering

Fractal Dimension (FD) are widely used to analyze a large variety of data patterns, most prominent amongst them is clustering of high dimensionality of a data space that embeds large different scales. The concept of fractals is mostly associated with geometrical objects satisfying two conditions, i.e. self-similarity and fractional dimensionality. Modelling and data mining by fractal analysis comprise methods to allocate a fractal dimension and fractal features to a signal or dataset in wide spectrum areas. Fractal based methods have been applied in different data mining approaches [92].

In this chapter we use fractal clustering in such a way that the gene expression sample points in the same cluster are more self- affine among themselves than to data points in other clusters. We study a sample of a specific microarray dataset and describe how this concept is used in microarray clustering.

5.1- Application of FD in gene expression sample signal

Fractal analysis is an effective scientific paradigm that has been used successfully in many areas including biomedical and biological sciences. It has been established as a useful method in quantifying the complexity of dynamical data and signals [93].

The determination of fractal dimension might be suitable method for the characterisation of microarray dataset analysis by a scaling exponent that measures the similarity of gene expression samples as a signal. It can be considered as a relative measure of the number of basic building blocks that form a genes sample pattern. In most situations, it is required to use a set with an invariant measure characterised by a whole spectrum of scaling exponents, instead of a single number of expression. Such a method is called multi-fractal. For instance, it can be used to measure signals in variation domain with different structural conditions dependent on the gradient occurrence in signal division. From this perspective, this occurrence is a gene expression sample feature that shifts the levels in a sample variation towards a composite of condition behaviour. Consequently, within FD the expression variations are linked with the changes in the expression sample signal, providing a computational means that tracks the existence of a fracture. However the FD method does not aim to identify the fractal characteristics of the expression signal vibrations but to identify the variations in their samples. In order to perform that, it evaluates the changes in the samples structural directly on the vibration signal by estimating its FD within a sample window across different gene profiles. As a result, the approach estimates the FD without requiring the reconstruction of the multidimensional phase space [94], resulting in a fast and efficient way of clustering the gene expression samples based on the depth of fracture in the sample signal. The work in this thesis is based on using the concepts of FD to clusters gene expression samples in such a way that the samples in the same cluster are more self-affine among themselves than to other clusters.

The fractal concepts of self-similarity and scaling invariance have been applied to many biological systems, from branching patterns of bronchial and circulatory vessels, to cardiac rhythms, to the geometry of shells and trees [95], and Local scaling and multifractal spectrum analyses of DNA sequences[96].

5.2- Fractal dimension analysis methods

The applications of FD in biomedical and signal processing include two types of approaches: (i) time domain where the original signal is considered as geometric and (ii) phase space domain which estimates the FD in state-space domain [97]. FD has many characteristics and different methods exist such as Hausdorff dimension, box dimension, information dimension and correlation dimension [94, 98]. These are summarized in Table (5-1). Clustering using FD is a type of grid-based clustering, where the data space is divided in cells by a grid. Some of the well known techniques that use grid-based clustering are STING [59], WaveCluster[60] and Hierarchical grid clustering[99].

Table 5-1 Fractal methods

Method	description	Expression
Hausdorff dimension	• Dataset is covered by cells s_i with variable diameter r_i, all $r_i < r$ • The collection of covering sets s_i with diameter less than or equal to r, which minimizes the sum • d-dimensional Hausdorff measure: • For every dataset $\Gamma^d{}_H$ is infinite if d is less than some critical value D_H, and 0 if d is greater than D_H • The critical value D_H is the *Hausdorff dimension* of the dataset	$\Gamma_{\mathrm{H}}^{\mathrm{d}}(\mathrm{r}) = \mathrm{inf}_{\mathrm{s_i}} \sum_{\mathrm{i}}(\mathrm{r_i})^{\mathrm{d}}$ $\Gamma_d^H = \lim_{n\to\infty} \Gamma_d^H(r)$
Box dimension	• Hausdorff dimension is not easy to calculate • Box-Counting D_B dimension is an upper bound of Hausdorff dimension, does not usually differ from it: • $v(r)$ – is the number of the boxes of size r needed to cover the dataset • Although Box-Counting dimension is easier to calculate than Hausdorff dimension, the algorithmic complexity grows exponentially with the set dimensionality => can be used only for low-dimensional datasets • Correlation dimension is computationally more feasible fractal dimension measure • Correlation dimension is an lower bound of the Box-Counting dimension	$D_B = \lim_{r\to 0} \frac{\log(v(r))}{\log(1/r)}$
Correlation dimension	• Let $x_1, x_2, x_3, \ldots, x_N$ be data points • Correlation integral can be defined as: $I(x)$ is indicator function: $I(x) = 1$, if x is true, $I(x) = 0$, otherwise. • Where $C(r)$ is number of points having smaller distance that a given distance r	$C_m(r)$ $= \lim_{N\to\infty} \frac{2}{N(N-1)} \sum_{i=1}^{N} \sum_{j=i+1}^{N} I(\vert x_j$ $- x_i\vert \le r)$ $D_r = \lim_{r\to 0} \frac{\log(C_m(r))}{\log(r)}$

5.2.1- Self-similar and Fractal developments

The idea of a fractal from a mathematical perspective is that, fractals are embodiments of iterations of nonlinear equations, commonly building a feedback loop. It is principally associated with geometrical objects satisfying two properties arranged as: self-similarity (means that an object is composed of small subunits on multiple levels that resemble the structure of the whole object) and fractional dimensionality (means that this condition requirement distinguishes fractals from Euclidean objects). Fractals are objects which have a structure of self-scaling: elements of the entire can be made to fit the whole by shifting and extending. FD is a measure of the self similar point set of the signals and has multiple definitions in general, their value is usually a non-integer, fractional number, and hence this dimension is referred to as fractal [100].

Considering morphological analysis, fractal features represent the morphology of the signals, which consists of a set of operators that transform signals according to specific characterizations. This morphological representation can be picked up and used by several applications based on fractal theory/morphological analysis [94].

Following this brief analysis, the application of morphological/fractal analysis is taken into consideration for the gene expression analysis in this work. Fractals are objects which possess a form of self-scaling: Parts of the whole can be made to fit the whole in some way or another by shifting and stretching. Fractals features represent the morphology of the signals in some way or the other. This utility of fractal/morphological analysis is a source of motivation to consider it as a useful tool for feature extraction since clustering of gene expression sample is all about extracting features and clustering the signals based on these features. Some of the features based on fractal analysis are described next:

1. **FRACTAL DIMENSIONS**. These represent a measure of the self similarity of the signals. A number of dimensions have been defined in this field. These include:

   Regularisation Dimension: This is derived as follows: initially computes smoother versions of the original signal, obtained through convolutions with a kernel. If the original signal is fractal, therefore its graph has infinite length, while all regularized versions have finite length. Moreover if the smoothing parameter tends to 0 then the smoothened version tends to the original signal, and its length will be likely to infinity[101].

Box Dimension: or the box-counting dimension which is the relation between the number and the size of the objects as follows:

$$N(s) \cong \left(\frac{1}{s^d}\right) \qquad when\, s \to 0 \qquad (5.1)$$

Where *N(s)* is the number of objects, *s* is the scale and *D* is the fractal dimension. This technique is used to estimate the scaling properties of a set by covering the set with boxes of size *s* and counting the number of boxes containing at least one pixel representing the object is given as [102]:

$$D = -\lim_{s \to 0} \frac{\log(N(s))}{\log(s)} \qquad (5.2)$$

The estimated value of *N(s)* is approximated by varying the origin of the grid until the smallest number is found. By means of Eq (5.2), the box-counting dimension *D* can be determined as the negative slope of *log N(s)* versus *log(s)*, measured over a range of box widths.

2. HOLDER FUNCTIONS/EXPONENTS. These are used to assess the continuity and differentiability of a function for measure the degree of regularity of the signals/functions [103]. Some examples include:

Point-wise Holder Functions: The point-wise Holder exponents, which characterizes the regularity of the measure/function under consideration.

The local Holder exponent: It characterizes the regularity of the function around any given point.

The long range dependence parameter: This one describes power law behaviour of the Fourier power spectrum near the zero frequencies.

3. ONE DIMENSIONAL MULTIFRACTAL SPECTRA. In this feature, the spectra provide information as to which singularities occur in a signal, and which are the dominant; a spectrum is a one dimensional curve, where the abscissa represents the Holder exponents present in the signal, and the ordinates are related to amount of points where will the encounter given singularity. There are two types of these spectra [103]:

The Legendre spectrum: This is based on the Legendre transform of a signal. It may be the Discrete Wavelet Transform based Legendre spectrum or a CWT based Legendre spectrum.

The large deviation spectrum: this spectrum yield statistical information related to the probability of finding a point with a given holder exponent in the signal. More precisely, it measures how this probability behaves with the change in resolution.

In general, the concept of a method for the estimation of the generalized fractal dimensions is by embedding the dataset in an n-dimensional grid which is seen as a partition step, and then computing the frequency with which data points fall into the *i-th* cell in the grid [92]. The length of the contour of a fractal signal in the plane is proportional to r^D, where r is the size of the grid used to measure the contour length and D is the fractal dimension. The sequence of test and steps is as follow:

i. Capture a subset of the signal and rescale it to the same size of the original, using the similar magnification feature for both its width and height.
ii. Compare the statistical properties of the rescaled signal with the original signal taking into consideration the magnification factors.

To formulate those into mathematical terms which enable calculation of self-similar process in data series time-dependent condition, the following equation must be satisfied:

$$y(t) \cong a^{\alpha} y(\tfrac{t}{a}) \qquad (5.3)$$

A self-similar process $y(t)$ with a parameter α has the identical probability distribution as a properly rescaled process [$a^{\alpha}\, y(t/a)$], i.e., a time data series which has been rescaled on the x-axis by a factor $a(t \rightarrow t/a)$ and on the y-axis by a factor of

$(a^{\alpha})(y \rightarrow a^{\alpha}\, y)$. The exponent ($\alpha$) is called the self-similarity parameter.

From Eq. (5.3) the self-similarity parameter α can be calculated by the following relation, where My (self similar pieces) and Mx (scaled down) are the appropriate magnification factors along the directions,

$$\textit{Fractal Dimension}\ (FD)\ \alpha = \frac{\ln M_y}{\ln M_x} \qquad (5.4)$$

Therefore,

$$\textit{Fractal Dimension}\ (FD)\ = \frac{\log\ (\textit{number of self similar pieces})}{\log\ (\textit{magnification factor})}$$

For the finite datasets, we assume it is statistically self-similar on a given range of scales (r_{min}, r_{max}) on which the self similarity assumption holds. To measure the FD, we use the slope of the correlation integral for a dataset which could be defined as

$C(r)$ = No. of pairs within distance r or less

Intuitively, the correlation FD indicates the growth rate of the number of pairs (No. of neighbours within a distance r) in the distance as follows:

No. of pairs $(\leq r)\ \alpha\ r^{D}$

Therefore, the FD could be used as a measure of the spread of the data and hence the intrinsic dimension of the dataset which is defined as the real number of dimensions in which the points can be embedded while keeping the distance among them. The embedded dimensionality of dataset might be representing as the number of attributes of the dataset that reflects its address space.

5.2.2- Self-similar Fractal to Integrated dataset series

The fluctuation involved in an integrated data series as a signal is a fundamental step in self-similarity fractal series analysis. Discrete data series are defined to be self-affine if their power-spectral density scales. Self-affine data series are where the power-spectral density scales as a power of the frequency. They appear in a wide variety of environments; examples in biomedical engineering include cardiac rhythms and gait dynamics [93]. Stochastic data series are characterized by a statistical distribution of values and by their persistence. Persistence is the degree to which values in a time series are internally correlated and can be classified in terms of range, short or long, and strength, weak or strong. Self-affine data series are scale invariant, thus always exhibit long-range persistence. Here, we quantify synthetic self-affine data series with varying degrees of long-range persistence strength and put them into the context of gene expression samples.

In the analyzing of data series, the calculation concerning Eq. (5.2) should always be performed using the following procedure:

1-the data series is divided into subsets of independent same size windows. In order to achieve more reliable estimation of the characteristic fluctuation at the window size, an average over all individual values of s obtained from these subsets should be processed.

2- Then repeat these calculations, not just for two window sizes, but for many different window sizes. The exponent α is estimated by fitting a line on the log-log plot of s versus n across the relevant range of scales.

5.2.3- Fractal of multi-dimensional data space

The dataset with multi-dimensional space characteristics is represented with columns as attributes (features) and rows as different data objects. Those datasets with numerical attributes are common in microarray datasets. It will be described as follow: the embedding dimension E of a microarray dataset is the dimension of its address space which represents the number of attributes of the dataset and the intrinsic dimension D is the dimension of the spatial object represented by the dataset, regardless of the space where it is embedded.

The Fractal datasets are characterized by their fractal dimensions. By embedding the dataset in an E-dimensional grid whose cells have sides of size r, the frequency of data points falling into the i-th cell can be calculated by:

$$FD = \frac{\log(\sum_i C_{r,i}^2)}{\log(r)} \qquad (5.5)$$

Where, r is the grid size, $C_{r,i}$ is the number of objects in the i-th cell under grid size r. Eq. (5.5) represent the correlation fractal dimension which measures the probability that two points chosen at random will be within a certain distance of each other. Changes in the correlation dimension mean changes in the distribution of points in the dataset. Here the use of correlation fractal dimension as the intrinsic dimension of a dataset is to identify the correlated attributes and discard those uncorrelated. The sum of occupancy can be defined as

$$S(r) = \sum_i C_{r,i}^2 \qquad (5.6)$$

The fractal dimension of the dataset is the derivative of *log(S(r))* with respect to the logarithm of the grid size. When assuming self-similar datasets, an expectation of the derivative results in a constant value. Thus, the correlation fractal dimension of a dataset can be obtained by plotting *S(r)* for different values of the grid size *r*, and calculating the slope of the resulting line.

5.2.4- Box-counting approach to Computing FD

This section presents the uses of box-counting algorithm to compute the fractal dimension of any given set of points in any E-dimensional space as shown in Figure(5.1).

```
Input: GEM dataset V (g rows, with n columns)
Output: Fractal Dimension FD
Begin
    i=0
    set r=grid-size
    $C_i$ = element part count in the i-th grid.
    $S(r) = \sum C_i^2$
    repeat r
end;
Plot the curve of log(r) and log(S(r))
Compute the slope of the curve which is equal to FD
```

Figure 5.1 Fractal dimension *FD* of a dataset *V* using *box-count approach*

A kind of multi-level grid structure to store the object count in different grids under different level (grid size) was used in the work [104]. The structure is easily built when considering each level has a size half of the size of the previous level, that is, the grid sizes are sequenced as (*r=1, 1/2, 1/4, 1/8, etc.*). Each level of the structure corresponds to a different size so the depth of the structure is equal to the number of points in the resulting plot. Since the structure is created in main memory, the depth of the structure is limited by the amount of main memory available.

5.3- miFD clustering analysis

Clustering microarray data in a *D*-dimensional space using fractal dimension method can be achieved using the box-counting and correlation fractal dimension algorithm [98]. The basic concept can be illustrated as a composition of multi resolution levels describing, for a given object, structures having a self-similarity on varying scales of magnification [101]. The method starts by partitioning the structure of the signal data space dimension into pieces of equal size in grid of the magnification factor size ε. Then, counting number of pieces that contain at least bit information of the original will occur. The process is repeated by iterative partitioning. The feature selection procedure based on correlation fractal dimension has been used to overcome the problems that are associated with higher dimensional datasets and are useful especially when the feature vector size is large[106].

If one defines $N(\varepsilon)$ as the number of self-similar cells occupied by points in the dataset, the plotting of $N(\varepsilon)$ versus the reciprocal size (ε) in a double-logarithmic diagram produces a graph called the box-counting plot. This plot yields a set of points on a line that exhibit a linear correlation. The slope of the best-fitting straight line to the plot represents the fractal box-counting dimension of the signal. Consequently FD can be calculated by taking the limit of the quotient of the log of the change in object size divided by the log of the change in the measurement scale, as the measurement scale approaches zero. The negative value of the slope of that plot is called Hausdorff fractal dimension as described in Eq. (5.7) for fractal dimension *D*:

$$D = -\,lim_{\varepsilon\to 0}\frac{log(N(\varepsilon)}{log(\varepsilon)} \qquad (5.7)$$

In practice, the spatial aggregation of the samples that produces a cluster, is specified by the correlation dimension D_r as defined in Eq.(5.8). Distance measures have been estimated based on changes in the correlation dimension D_r in the distribution of samples in the dataset. Let $C(r)$ be the correlation functions of pairs of data samples within distance *r*, then

$$D_r = lim_{r\to 0}\frac{log(C(r)}{log(r)} \qquad (5.8)$$

Therefore, the correlation dimension can be used to identify data clustering.

They are represented by a set of boxes that records the samples set. If *f* represents the number of clusters found in the initialization step, the data partition is $Cl=\{Cl_1,Cl_2,..,Cl_f\}$, where Cl_i is the composite of the set of boxes that represents cluster *i*. The method can compute the FD for each cluster, then detecting the minimal fractal to generate group of clusters.

5.4- Performance analysis of miFD

In order to perform the microarray FD analysis, some basic parameters must be carefully chosen since their variation causes varying performance. To obtain useful results with FD and to apply it successfully, it is necessary to understand the relationship and the effect of the changes in parameters on the clustering process. The main influencing parameter effects on FD analysis performance are based on the chosen level of partition and grid resolution which attempts to provide a reasonable model of the data.

Significantly, it is important to take into consideration how to make boxes that cover the whole signal without any dislocation. The occurrence of any miss boxing causes some occupying failures of the signal, therefore the calculated box-counting dimension affects the accuracy. Alternatively, since the number of boxes affects on the estimation of dimension, how the signal is boxed changes the value of the dimension. Consequently the efficiency of the box covering and the covering style can play a role in the value of miss-computation of the box-counting dimension and affect on clustering results.

The miFD method consists of the following steps:

1- From the microarray dataset, select the genes expression sample data and represent it as a signal.

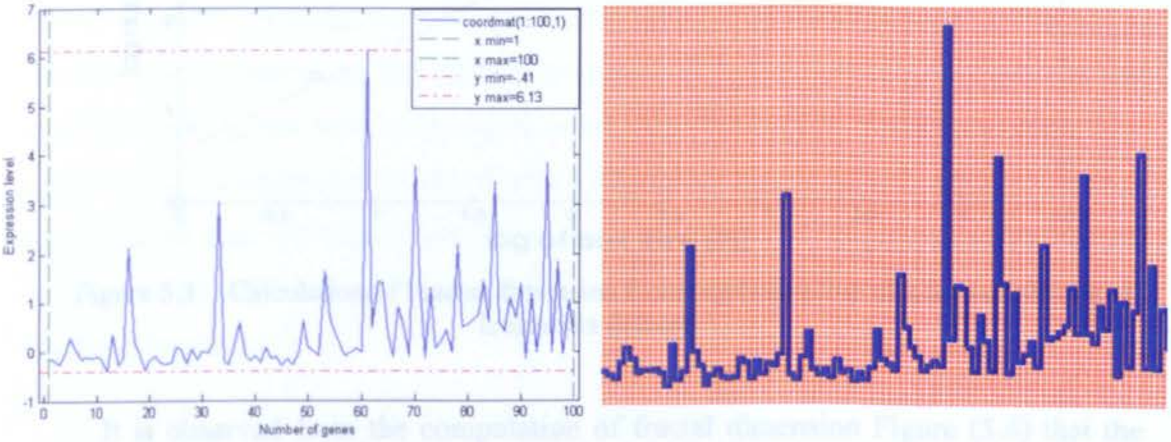
2- The variation in the signal is recorded according to a sample interval, concerning the starting, ending, minimum and maximum values.

3- Perform regularization of the sample signal into unit square.

4- Choose a sequence of τ_k's as a mesh grid, each τ_k unit scale of a grid as squares with length τ.

5- Count the number of grid squares $N\tau_k$, which intersects with signal. It is very significant to care about the covering of the whole signal without any dislocation, because the number of boxes affects the estimation of FD and, therefore, performance of the analysis.

6- Plot *log* $N\tau_k$ vs *–log* and find the slope of regression line. The slope is the fractal dimension of *F*.



a- Expression signal of Sample 1 b- After covering with boxes (with box size τ)

Figure 5.2 Evaluation of box counting fractal dimension for Leukaemia dataset

Figure (5.2a) shows the sample one in Leukaemia microarray dataset with *g=100genes* involved, while the next Figure (5.2b) illustrates the covering of sample signal with grid of boxes with size τ.

In order to make accurate measurement, different box sizes were used. Figure(5.3) shows the variation of the number of boxes according to changing box sizes, in addition the regression line that represents the linear relationship between log number of boxes value and the log of box size is plotted. Then we compute the regression equation: [*y=1.1x+.54*]. The slope of this regression line (i.e. 1.1) represents the box counting fractal dimension. The value of *R* represents the performance quality or regression.

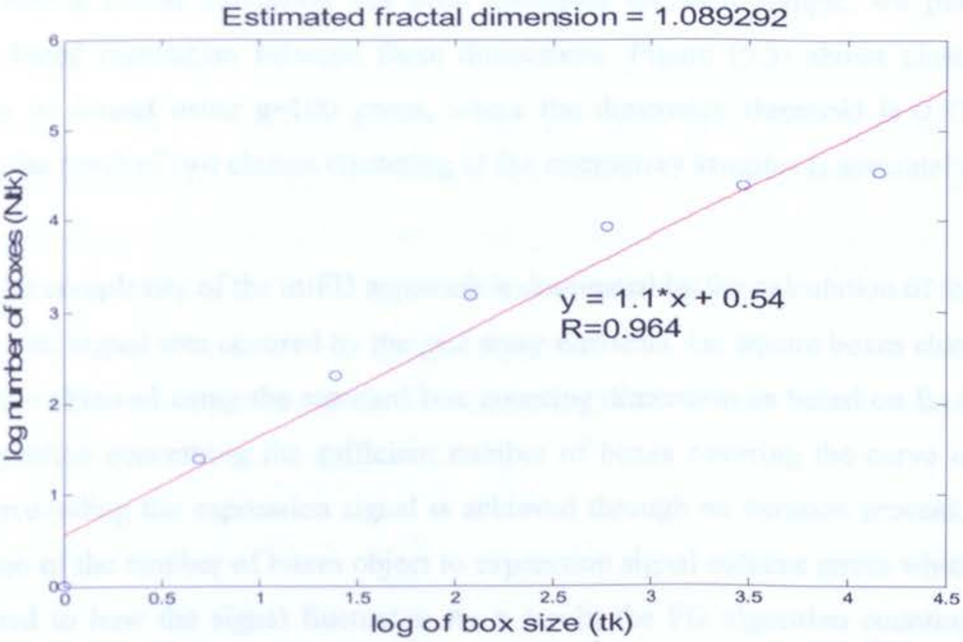


Figure 5.3 Calculation of Fractal dimension from regression line plot for sample one of leukaemia dataset

It is observed from the computation of fractal dimension Figure (5.4) that the number of boxes reaches a saturation value from where there is no further change even when the box size increases. Therefore, the best choice for the value of the fractal dimension corresponds to the last point before saturation.

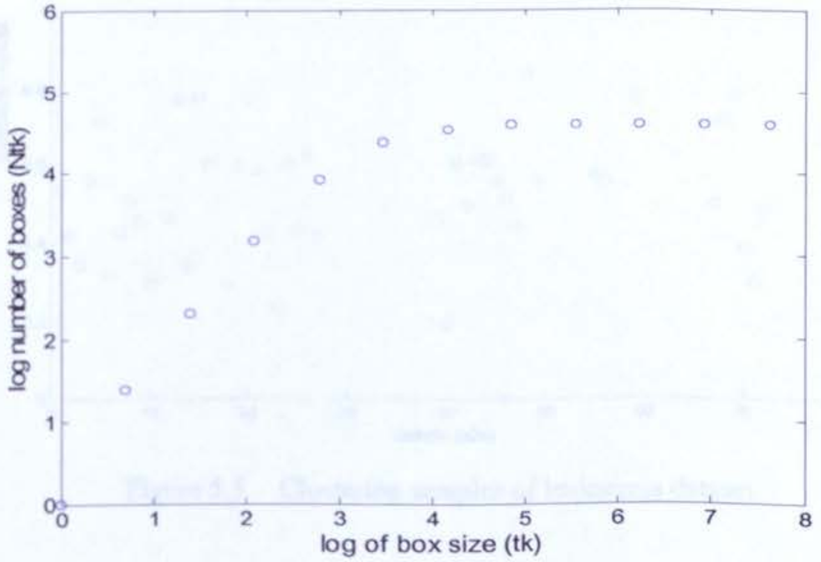


Figure 5.4 Plot of log of box size versus the log of the number of boxes for sample one of leukaemia dataset

Once a fractal dimension has been estimated for each sample, we perform cluster based correlation between these dimensions. Figure (5.5) shows clustering samples in dataset using g=100 genes, where the dimension threshold is 0.875. It reveals the result of two classes clustering of the microarray samples is accurately.

The complexity of the miFD approach is dominated by the calculation of the FD for the total signal area covered by the grid array elements, i.e. square boxes elements which are obtained using the standard box counting dimension as based on Eq.(5.7). An inspection concerning the sufficient number of boxes covering the curve in the area surrounding the expression signal is achieved through an iteration process. The reduction of the number of boxes object to expression signal induces errors which are correlated to how the signal fluctuates. As a result, the FD algorithm counting the number of boxes required to cover the curve relatively with box sizes and then establish log-log plot.

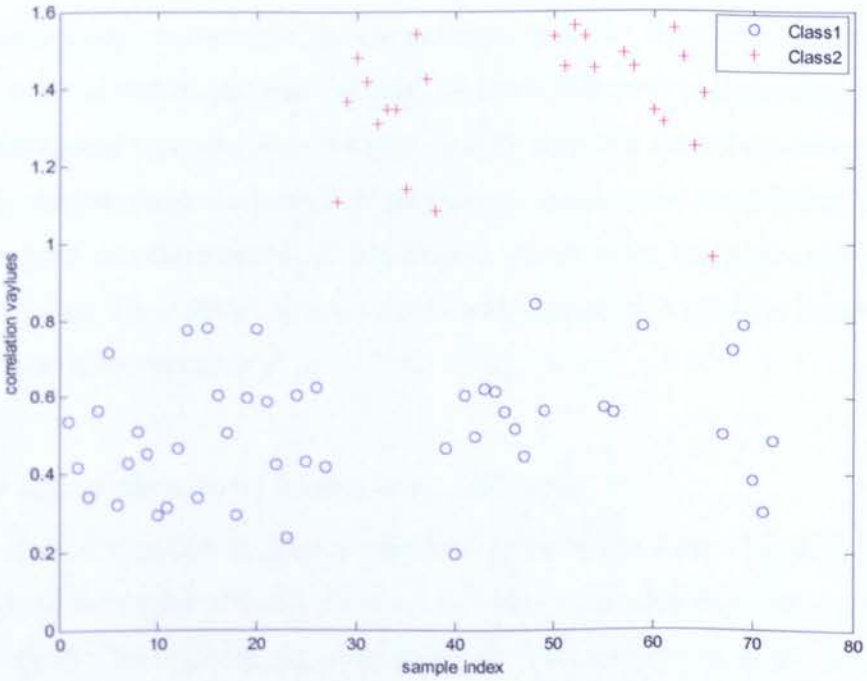


Figure 5.5 Clustering samples of leukaemia dataset

5.5- Conclusion

In this chapter we have explained the FD approach in data clustering. The chapter outlined the basic FD characteristic of the microarray gene sample signals and illustrated different fractal methods. We introduced the miFD method to estimate the fractal dimension of the gene expression samples and then the correlation between these dimensions is applied to produce the relevant index of clustering. Performance analysis of the miFD method and application on test data set has been discussed. In the next chapter we apply the three DSP methods to different microarray datasets in order to validate the GSP clustering abilities.

CHAPTER 6

Performance analysis of GSP methods

In the previous chapters we have introduced three GSP methods for microarray data clustering. In this chapter, we first provide an overview of the characteristics of the microarray gene expression test datasets that we used in this work. We then apply the GSP methods to these microarray datasets to validate their clustering abilities. Finally, we will provide a comparative performance study between the proposed GSP methods.

6.1- Microarray test data types

In this section we describe the characteristics of the microarray datasets that we use in this work as summarised in Table (6-1). Each dataset is composed of two subsets namely training and test sets. However, since GSP methods do not depend on any form of training, we combine both sets to produce a unique test sets. These dataset are selected as they are considered as benchmark datasets for relevant microarray data clustering studies. The following is a brief description of each of these datasets and the clustering tasks they require.

1-Acute leukaemia dataset (Golub et al., 1999) [8]

This dataset contains measurements corresponding to Acute Myeloid Leukaemia (AML) and Acute Lymphoblastic Leukaemia (ALL) samples from bone marrow and peripheral blood. The training set consists of 38 bone marrow samples obtained from adult acute leukaemia patients. 11 suffer from AML and 27 from ALL. The test set consists of 34 patients, 14 suffer from AML and 20 from ALL. Therefore, the total number of samples is 72 and the number of gene expression levels in the microarray is 7129. The goal is to classify 47 patients as being ALL and 25 as AML.

2- Colon cancer dataset (Alon et al., 1999) [106]

This dataset contains measurements corresponding to Colon Adenocarcinoma and normal colon tissues which were collected from patients. The training set consists of 40 colon tissues, 14 are normal and 26 are tumour samples. The test set consists of 22 tissues, 8 are normal and 14 are tumour samples. Therefore, the total number of samples is 62 and the number of gene expression levels is 2000. The goal here is to classify 40 tissues as being cancerous and 22 as normal.

3- Hepatocellular carcinoma dataset (Iizuka et al., 2003) [107]

This dataset contains measurements corresponding to Hepatocellular carcinoma tissues. The training set consists of 33 Hepatocellular carcinoma tissues, 12 suffer from early intrahepatic recurrence and 21 do not. The test set consists of 27 Hepatocellular carcinoma tissues, 8 suffer from early intrahepatic recurrence and 19 do not. Therefore the total number of samples is 60 and the number of gene expression levels is 7129.

The goal is to classify 20 tissues as being suffer from early intrahepatic recurrence and 40 do not.

4- Prostate cancer dataset (Singh et al., 2002).[108]

This dataset contains measurements derived from patients with prostate tumours and non-tumour prostate samples. The training set consists of 102 prostate tissues, 50 are normal and 52 are tumour samples. The test set consists of 34 tissues, 9 are normal and 25 are tumour samples. Therefore the total number of samples is 136 and the number of gene expression levels is 12600. The goal is to classify 77 tissues as being tumour and 59 as normal.

5- High-grade glioma dataset (Nutt et al., 2003) [109]

This dataset contains measurements corresponding to High-grade glioma derived from different group of patients. The training set consists of 21 gliomas with classic histology, 14 are glioblastomas and 7 are anaplastic oligodendrogliomas. The test set consists of 29 gliomas with non-classic histology, 14 are glioblastomas and 15 are anaplastic oligodendrogliomas. Therefore, the total number of samples is 50 and the number of gene expression levels is 12625.The goal is to classify samples as 28 glioblastomas and 22 as anaplastic oligodendrogliomas.

Table 6-1 Summary of the tested microarray datasets

Study	Type of disease	No. of genes	Training set			Test set			Total no. of samples	Goal
			Total	Class1	Class2	Tota l	Class1	Class2		
Golub, 1999[8]	Leukaemia	7129	38	11 AML	27 ALL	34	14 AML	20 ALL	72	47 ALL 25 AML
Alone, 1999[106]	Colon	2000	40	14 normal	26 tumour	22	8 normal	14 tumour	62	40 tumour 22 normal
Iizuka, 2003[107]	Hepato-cellular	7129	33	12 sick	21 healthy	27	8 sick	19 healthy	60	20 sick 40 healthy
Singh, 2002[108]	Prostate	12600	102	52 tumour	50 normal	34	25 tumour	9 normal	136	77 tumour 59 normal
Nutt, 2003[109]	Gliomas	12625	21	14 glio	7 oligo	29	14 glio	15 oligo	50	28 glio 22 oligo

A comparison performance has been made between well known state-of-art clustering approaches as summarised in Table (6-2). It shows the description of their methods with minimum number of genes used to satisfy the clustering. These methods are based on a variety of techniques dealing with dimensionality reduction and production of distance and similarity measures. They include statistical, deterministic, probabilistic and computational methods.

Since most classification and clustering methods require a predefined gene sample similarity or distance metric, their performance rely on how well that metric reflects the real relationship among samples. These metrics, which are data-dependent, include Euclidean distance, Manhattan distance, and Pearson-correlation. However, in practice, it is desirable to estimate the metric adaptively based on input data that depends on the local features of the gene sample data.

Table 6-2 Summary of microarray clustering studies

Study	Dataset	Techniques	Generation procedure	No. of genes
Golub, 1999[8]	Leukaemia	T-test	T-statistics for gene selection Weighting voting for classification	50
Alone, 1999[107]	Colon	T-way	Correlation for gene selection Deterministic annealing algorithm for clustering	500
Iizuka, 2003[107]	Hepatocllular	FLC, SVM	Classification using either Fisher Linear Classifier or Support Vector Machine	12 50
Singh, 2002 [108]	Prostate	kNN	K-Nearest Neighbour clustering	5
Nutt, 2003 [109]	Gliomas	kNN	K-Nearest Neighbour clustering	19
Tibshirani, 2002[110]	Leukaemia	PAM	Class prediction using Prediction Analysis of Microarrays - statistical technique using nearest shrunken centroid	21
Mukkamala, 2005 [111]	Leukaemia, Prostate and Colon	MARS, LGP, CART,RF	Classification using either Linear Genetic Programs or Multivariate Regression Splines or Classification & Regression Tress or random forest	6 27 53
Nguyen, 2002[112]	Leukaemia, and Colon	PLSLD	Dimension reduction using Partial Least Square Classification using Logistic Discrimination and quadratic discriminant analysis	25
Liu, 2005[113]	Leukaemia, and Colon	KPCA	Dimension reduction using Kernel Principal Component Analysis Classification with logistic regression (discrimination).	150
Jong, 2003[114]	Leukaemia, and Colon	FJC	Preprocessing using support vector classifiers Clustering using Find and Join Clusters method.	50 & 200
Chanda, 2006[115]	Leukaemia, and Colon	Two-way	Preprocessing using entropy and correlation measure Clustering based on fuzzy C-means	287 & 294
Furey, 2000[116]	Leukaemia, and Colon	SVM	Classification using Support Vector Machine	2,5 & 10
Ding, 2004[117]	Leukaemia, and Colon	MRMR	Minimum redundancy - maximum relevance (MRMR) feature selection	60
Huerta, 2006[118]	Leukaemia, and Colon	GA/SVM	Preprocessing using Genetic Algorithm Classification using Support Vector Machine	25 &10
Huang, 2006[119]	Leukaemia, Colon, Glioma and Hepatocellular	P-ICR	Regularizing gene expression data using Independent Component analysis Classification using Penalized discriminant method	20

6.2- Preprocessing

Preprocessing is the initial step of preparing microarray datasets for their analysis. Its main concerns is gene selection as described earlier in chapter 3, section 3.4. A microarray dataset consists of a set of experiments, which generate gene expression profiles measured under several conditions or from different patients. Usually, it contains a considerable number of genes which are irrelevant to the clinical process. Therefore, it is required to pre-process microarray data prior to its analysis. The expression values of gene profiles often demonstrate little variation over the different experiments and show seemingly random and meaningless profiles. Another problem with microarray datasets is that they regularly contain highly unreliable expression profiles with a considerable number of missing values that affect accuracy. If such datasets were passed to the clustering algorithms, the quality of the clustering results could significantly degrade. A solution that has been proposed is to combine normalisation with feature selection, i.e. at least a fraction of the undesired genes are removed from the data because they do not satisfy one or possibly more criteria such a threshold of one standard deviation of the expression values in a profile.

The normalisation step produces a collection of expression profiles which have an average of zero and a standard deviation of one. The results are often represented by Boxplot that allow visualization of the normalisation performance, as shown in Figure(6.1) with samples (1-30) of the Leukaemia dataset. Boxplot is useful for revealing the centre, the spread, the distribution of the data and the presence of outliers. They consist of a rectangular box and whisker plot for each sample of the microarray[3]. The box has lines at the lower quartile, median, and upper quartile values. The whiskers are lines extending from each end of the box to show the extent of the rest of the data. Therefore, the figure shows the corresponding statistical distribution of gene expression variance level after normalisation. While most gene expression values are inside the box, values outside the box may reveal potential gene outliers. For example, the median for sample 6 is (0.05) while it is (0.146) for sample 9. This sample has more potential outliers as expressed by the variance.

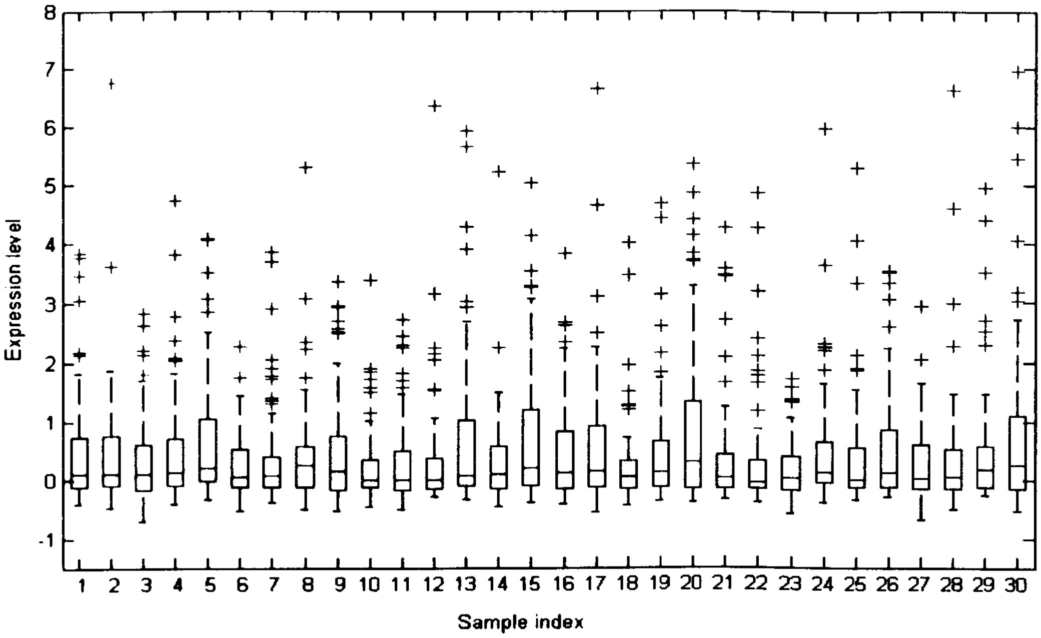
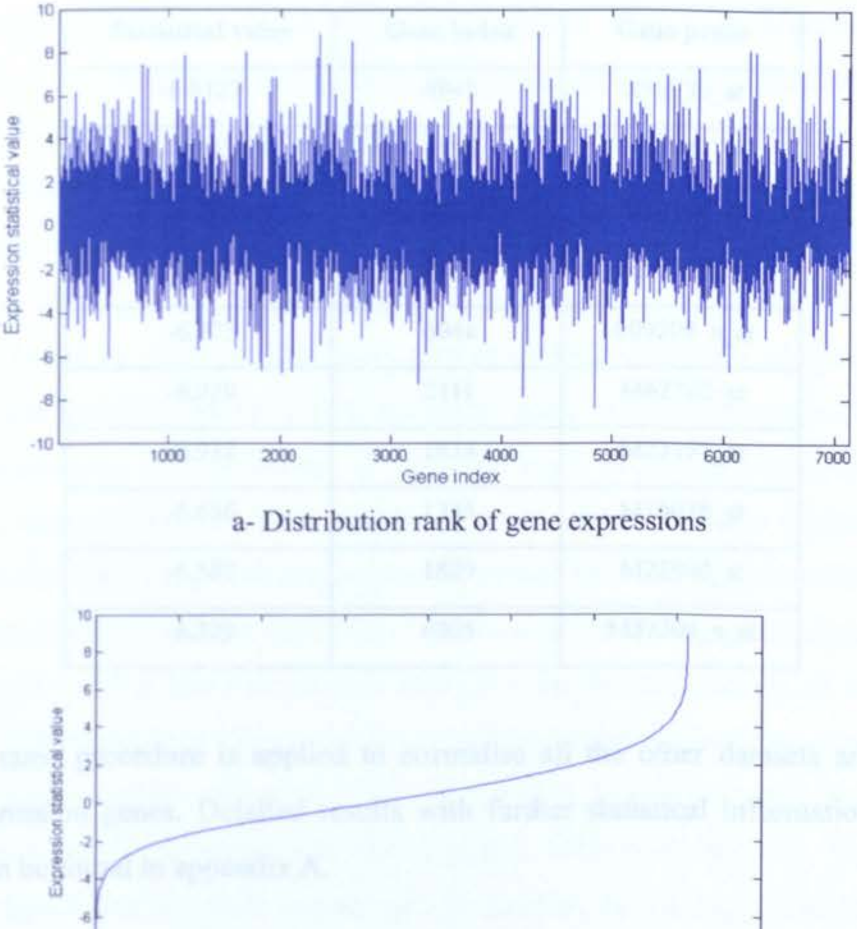


Figure 6.1 Boxplot analysis for the Leukaemia dataset.

After the statistical normalisation process, gene selection is based on gene expression ranking in the microarray dataset. Figure (6.2a) shows a sample of the distribution rank score relative to the differential gene expression activity, under expressed genes have negative values, whereas over expressed ones have positive values. In Figure (6.2b), genes are ranked to find the top genes located in the under and over expression regions. The histogram in Figure (6.2c) shows the normal distribution of the statistical expression values plotted against the number of genes. It shows a normal distribution, which is the type of distribution generally found in microarray data analysis.

The top 10 genes, according to their statistical score, are listed in Table (6-3). In this analysis, only genes which show distinct expression values between samples are truly relevant for sample classification. This is a way to reduce the dimensionality of microarray.



a- Distribution rank of gene expressions

b- Order sort of gene expressions rank

c- Relative distribution rank of gene expressions

Figure 6.2 Distribution of expressed genes in Leukaemia dataset

Table 6.3 Highly expression genes in Leukaemia dataset

Statistical value	Gene index	Gene probe
-8.8127	4847	X95735_at
-7.843	4196	X17042_at
-7.319	3252	U46499_at
-7.263	2020	M55150_at
-6.975	6041	L09209_s_at
-6.939	2111	M62762_at
-6.932	1834	M23197_at
-6.656	1745	M16038_at
-6.589	1829	M22960_at
-6.329	6005	M32304_s_at

The same procedure is applied to normalise all the other datasets and find the highly expression genes. Detailed results with further statistical information on other datasets can be found in appendix A.

6.3- Application of miLPC clustering

This section presents the results obtained from applying the LPC approach, explained in chapter 3, on the selected datasets. The application of the method is performed with multiple runs to achieve the analysis using different LPC orders with different numbers of genes to detect the best MSE value according to the estimated gene expression coefficients model. Then vector quantisation is applied to cluster the predictive coefficients. Moreover an exploration of different combination of parameters that affect clustering performance is presented.

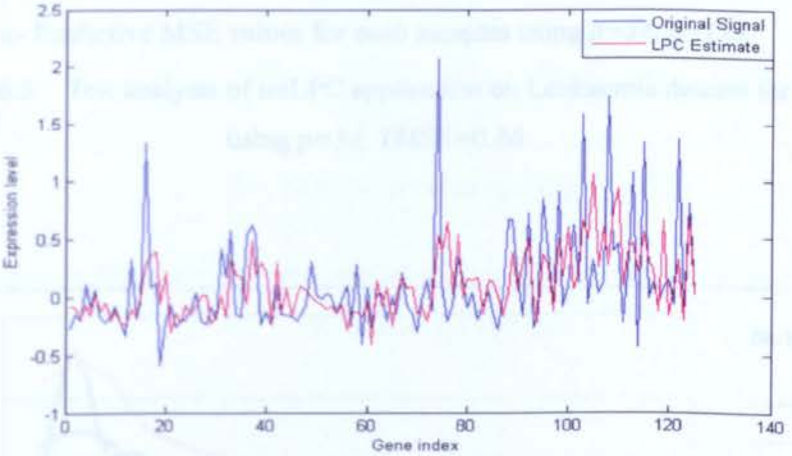
6.3.1- miLPC on Leukaemia dataset

The pre processing step selects the genes that will be processed using the miLPC method. Then the LPC algorithm estimates the best prediction coefficients over these genes. In order to test the estimated coefficients of the gene signal, a reconstruction process is performed to re-establish the gene signal and then compare it with the original signal through calculation of the MSE value. The minimum value detects the best prediction coefficients for a given gene signal. Figure (6.3a) illustrates the performance of the algorithm in estimating the reconstruction gene signal for sample 23, involved *g=125* genes with LPC order *p=34*. The estimated MSE for this reconstruction as shown in Figure (6.3b) is (0.13) which represent the minimum MSE value relative to other samples in the microarray, whereas the total MSE for all samples is *TMSE=0.86* as shown in Figure (6.3c). Figure (6.4) shows a spectrum of MSE for predictive coefficients based on multiple runs according to different numbers of involved genes with respect to multiple values of LPC order. By passing the estimated coefficients to vector quantisation, the clustering is accomplished to group those coefficients. Figure(6.4a) demonstrates the MSE to different LPC order for specific number of genes including the average (*Avg*) that represented by dotted line. Note that the MSE value is influenced by the LPC order, it appears to be inversely related to the model order due to the prediction order *p* has the greatest influence on the complexity of coefficients calculation that comes from increasing the linear combination of previous samples taken from combinations of the enhanced digital filter stages. Considering this higher model order appears to be advantageous, because the difference between the original expression signal and the final results will be smaller. Accordingly there is possible range of LPC order values could result given accurate clustering, the figure shows that a minimum MSE identified as (*Th =0.907*) can separate the classes with accurate cluster which is below the average line. On the other hand Figure (6.4b) demonstrates the MSE to different number of genes for specific number of LPC order including the average (*Avg*) that represented by dotted line. Note that the MSE value also influenced by the specific number of genes involved. However in many cases a considerable range of genes are differentially expressed, this sense is due to a relatively small variation in expression have considerable noise that affected on the coefficients value.

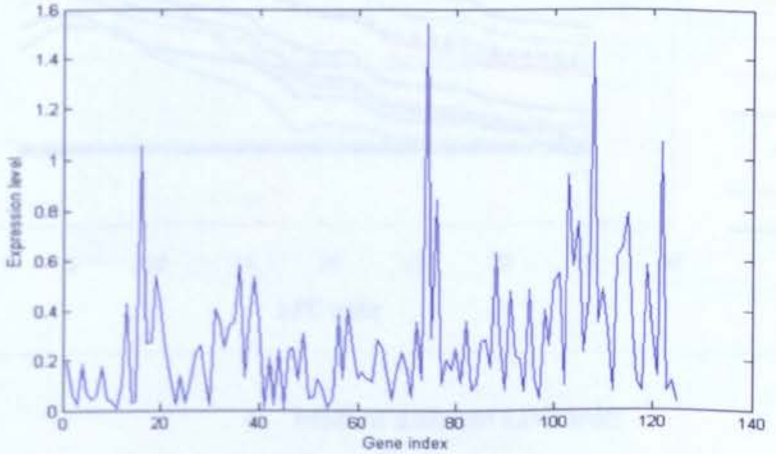
Figure (6.5) gives the graphical representation of clustering error of different value of gene number with respect to different LPC orders. It shows the minimum number of samples that are not clustered accurately, while obtained accurate clustering when involved range of genes *g={75-125}* processing with LPC order *p={34, 35}*. Figure (6.6) shows voronoi diagram to the result of two classes clustering of the microarray samples.

On the subject of the computational aspects of the miLPC approach, the calculation of the LPC spectral distortion measure mostly engages in the computations of the LPC covariance algorithm enhanced with line spectral frequency. Therefore, truncation errors and the associated filter behaviour might have considerable impact on accuracy.

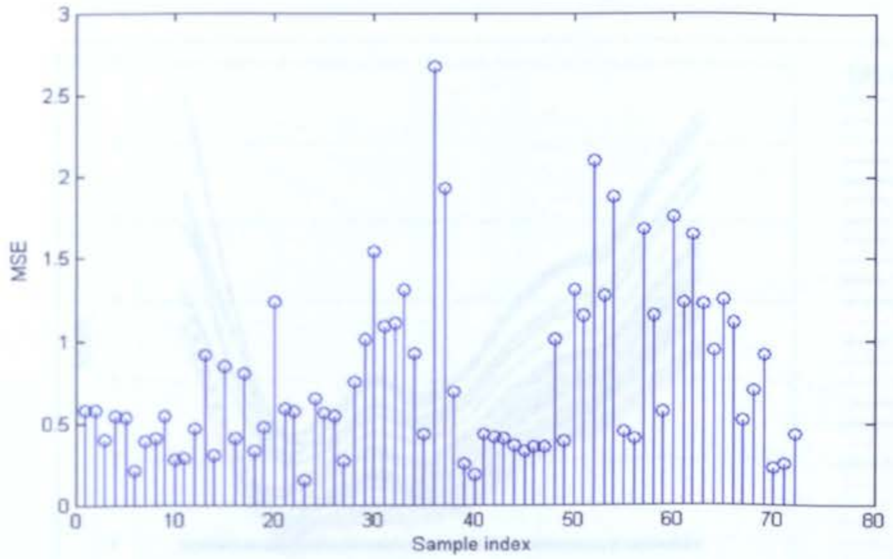
The experimental results and comparisons of model prediction have shown the performance of the miLPC approach. The model is mathematically reasonable concerning the flow of calculation in computing the framework of LPC embedded with VQ. This feature is very efficient for real-time implementation and comparison of gene sample signals.



a- Comparative expression profile between original and estimated sample signal for *g=125* of *sample 23* using *p=34*

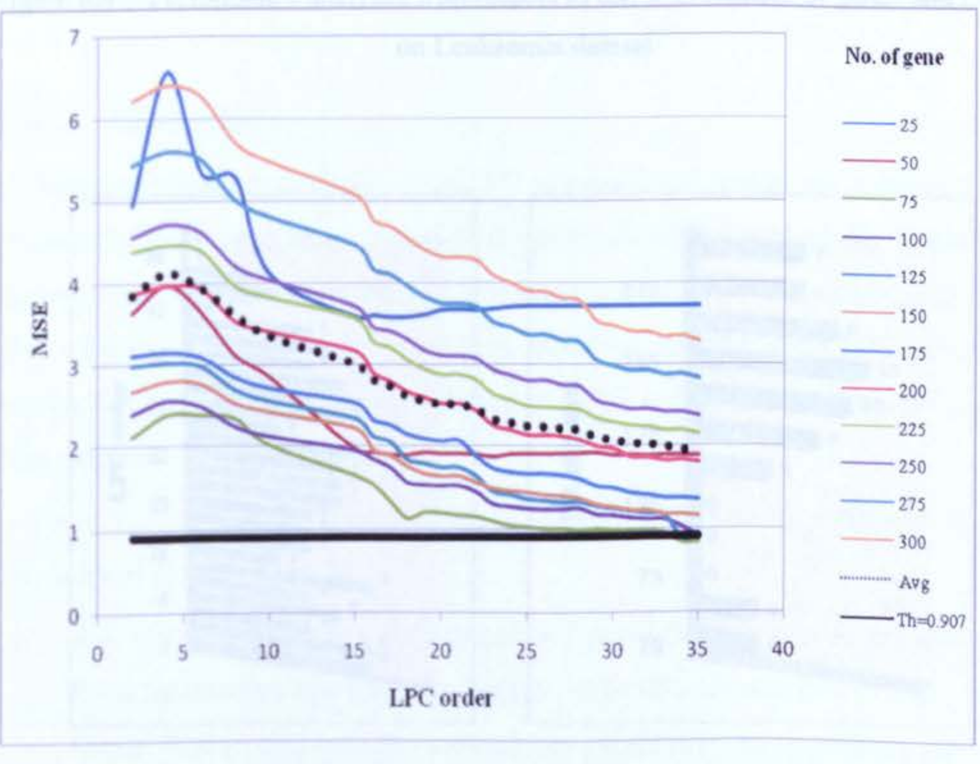


b- Error signal between original and estimated expression signal, *MSE=0.13*

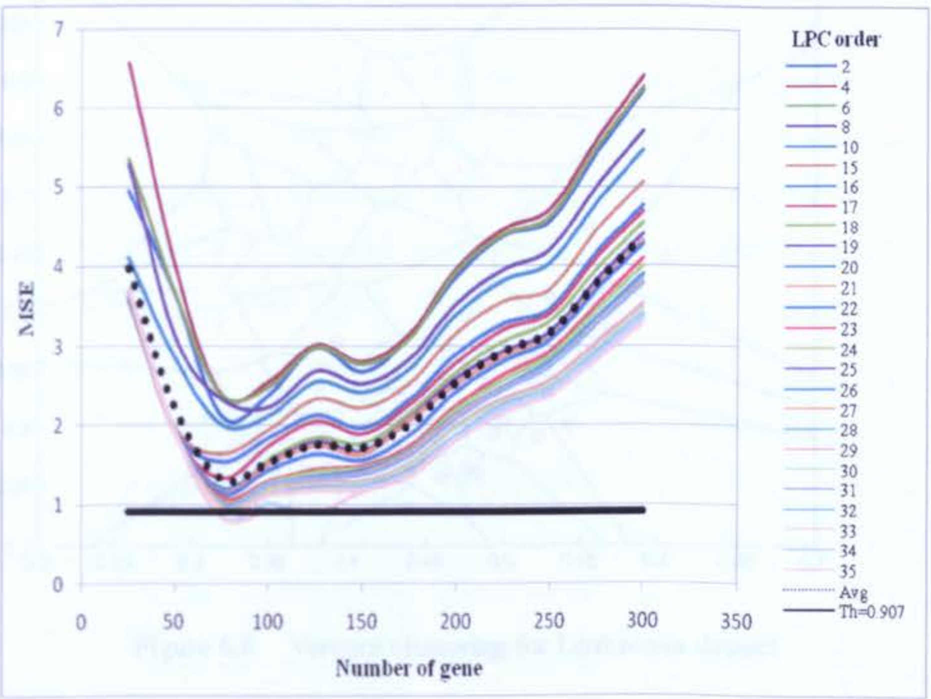


c- Predictive MSE values for each samples using *p=34, g=125*

Figure 6.3 Test analysis of miLPC application on Leukaemia dataset for *g=125* using *p=34, TMSE=0.86*



a- MSE of different LPC order



b- MSE of different number of gene

Figure 6.4 Performance analyses with respect to different number of genes and LPC orders on Leukaemia dataset

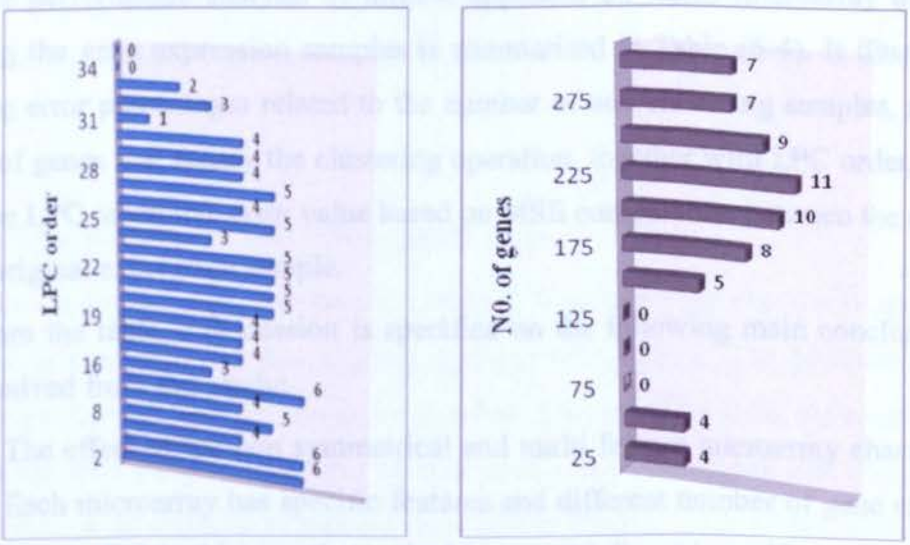


Figure 6.5 Clustering errors with respect to different number of genes and LPC orders on Leukaemia dataset

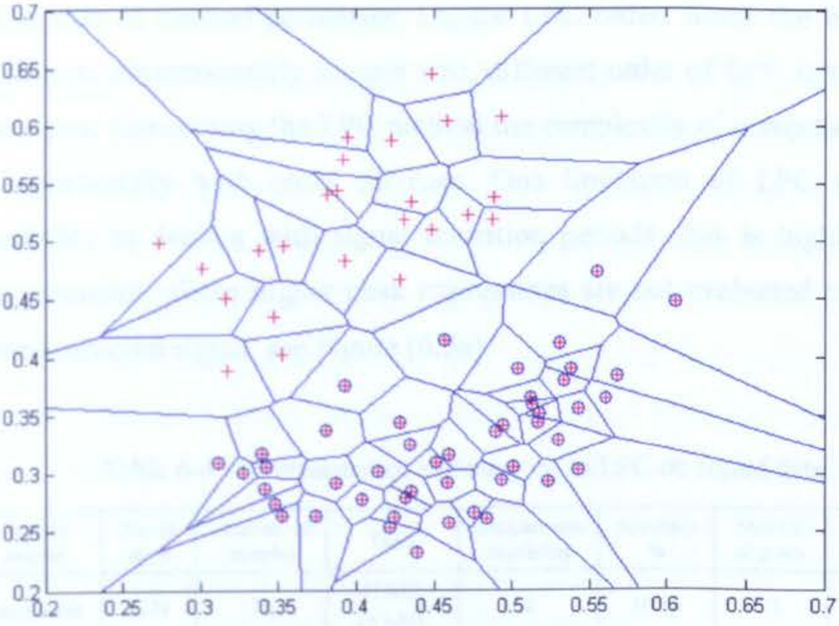


Figure 6.6 Voronoi clustering for Leukaemia dataset

The same procedure is applied to the other datasets to achieve their clustering. Detailed results with further information on other datasets can be found in appendix.

6.3.2- Discussion

The performance analysis of miLPC approach on tested microarray datasets to clustering the gene expression samples is summarised in Table (6-4). It illustrates the clustering error percentages related to the number of non clustering samples, minimum number of genes that satisfy the clustering operation, together with LPC order, and also shows the LPC predictive error value based on MSE computation between the estimated and the original expression sample.

From the table, a discussion is specified on the following main conclusions that can be derived from this study:

i- The effect of the non symmetrical and multi feature microarray characteristic: Each microarray has specific features and different number of gene expression levels. Within this case the method has sensitivity and specificity for individual clustering analysis of samples. In this case, the use of normalization and dimensionality reduction allow obtaining a common scale for analysis.

ii- The role of method parameter, i.e, the LPC order: Since the microarray has different dimensionality sample size, different order of LPC is required in the analysis. Concerning the LPC method the complexity of computation increases proportionally with order increase. One limitation of LPC method is its inability in dealing with signal transition periods: this is highlighted in our experiments where higher peak expressions are not evaluated properly in the reconstructed signal, see Figure (6.3a).

Table 6-4 Performance summary of miLPC on tested data.

Study	Type of disease	No. of genes	Total no. of samples	Goal	Samples non clustering	Accuracy %	Min no. of genes	LPC order	Predictive error
Golub	Leukaemia	7129	72	47 ALL 25 AML	0	100%	75	34	0.838
Alone	Colon	2000	62	40 tumour 22 normal	3	95%	100	32	2.97
Iizuka	Hepato-cellular	7129	60	20 sick 40 healthy	14	76%	125	29	0.528
Singh	Prostate	12600	136	77 tumour 59 normal	14	90%	125	28	0.92
Nutt	Gliomas	12625	50	28 glio 22 oligo	5	90%	75	26	1.47

A comparative performance of the miLPC approach with other well known state-of-art clustering approaches as summarised in Table (6-11) shows, that despite its limitations, the miLPC approach outperformed all the other methods in all the tested datasets.

6.4- Application of miDWD clustering

This section presents the results obtained from applying the DWD approach as explained in chapter 4 on the selected microarrays datasets. The application of the method is performed with multiple runs to achieve the analysis using different values of DWD levels with different numbers of gene involved to detect the best MSE value in the estimated gene expression wavelet coefficients model. Then vector quantisation is applied to achieve the clustering of the predictive wavelet coefficients. Moreover an exploration of different combination of parameters that affect on clustering performance for each dataset will be introduced.

6.4.1- miDWD on Leukaemia dataset

Following the pre processing step for selecting number of gene from Leukaemia dataset, the miDWD algorithm estimates the best prediction coefficients over these genes. In order to test the estimated coefficients of the gene signal, a reconstruction process is performed to re-establish the gene signal and then compare it with the original signal through calculation of the MSE value. The miDWD method is started from scale-oriented decomposition, and then to analyse the obtained signals on frequency subbands to estimate the coefficients model. Using these decomposition coefficients, microarray data clustering can be achieved by measuring similarities between coefficients model using the vector quantisation method. Figure (6.7) illustrates the performance of the miDWD algorithm in estimating the reconstruction gene signal using two levels of Daubechies Wavelet D2 and involved *g=100* genes.

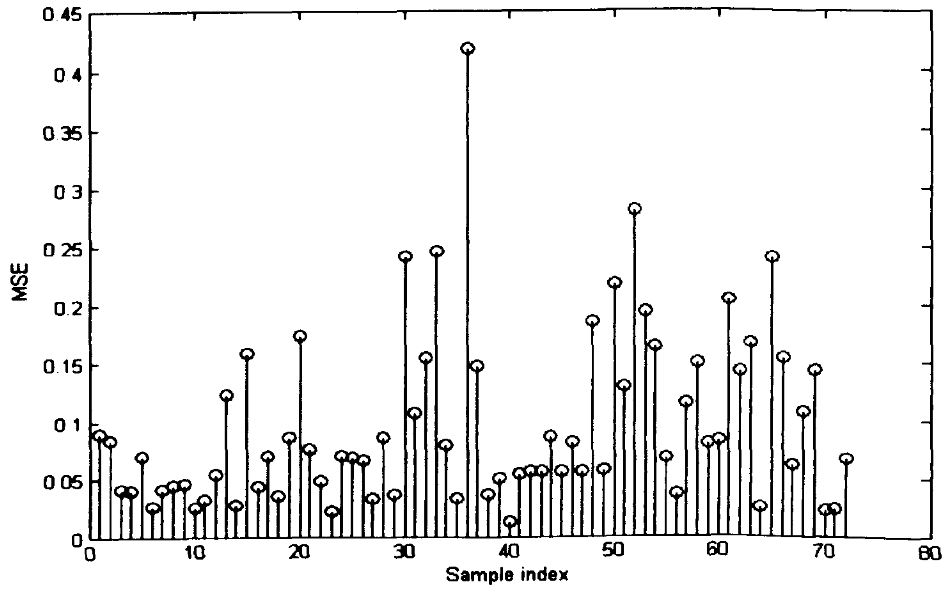
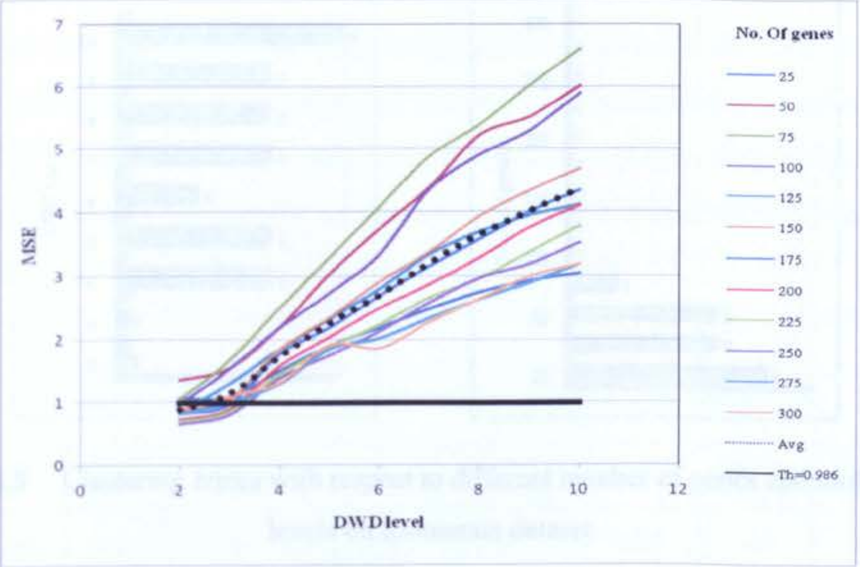


Figure 6.7 MSE estimation for each sample using *lv=2* and *g=100* of miDWD application on leukaemia dataset

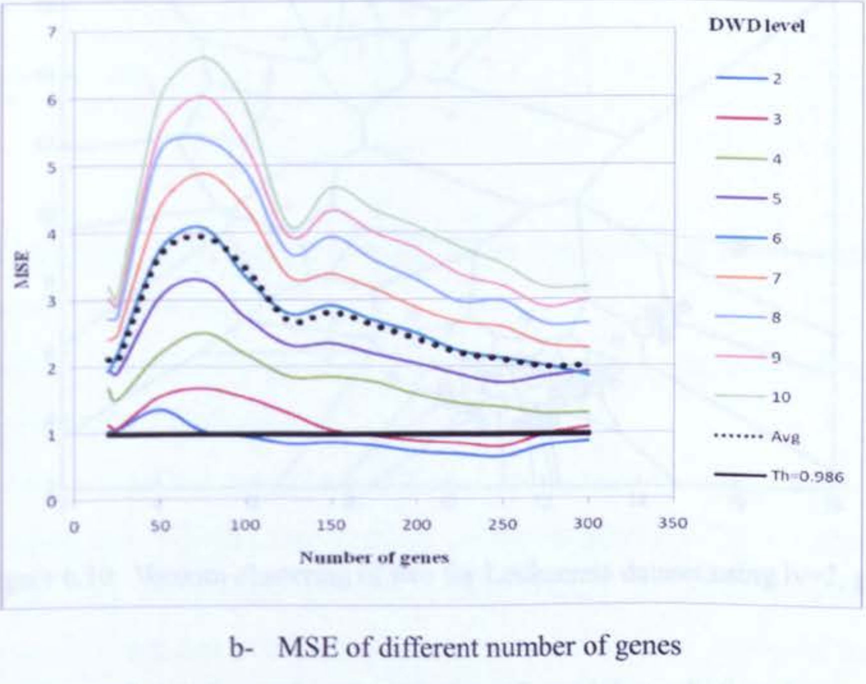
The total estimated MSE for this reconstruction to all samples is (0.986), which will cause to separate the classes with accurate cluster. Figure (6.8) shows a spectrum of MSE for predictive coefficients based on multiple run according to different number of involved genes with respect to multiple values of DWD levels. By passing the estimated coefficients to vector quantisation, the clustering is accomplished to group those coefficients. Figure (6.8a) demonstrates the MSE to different DWD levels for specific number of genes including the average (*Avg*) that represented by dotted line. Note that the MSE value influenced by the DWD levels as increasing the level of wavelet causes to proportionally increasing MSE value due to the weakness happened in the partitioning of expression signals and by the way increasing the noise. Considering this lower level appears to be advantageous, because the difference between the original expression signal and the final results will be smaller. Accordingly there is possible range of DWD level values could result given accurate clustering, the figure shows that a minimum MSE identified as (*Th =0.986*) can separate the classes with accurate cluster which is below the average line. On the other hand Figure (6.8b) demonstrates the MSE to different number of genes for specific number of levels including the average (*Avg*) that represented by dotted line. Note that the MSE value also influenced by the specific number of involved genes. However increasing the dimensionality of data represented by increasing number of genes, it has improved the effectiveness of clustering.

Figure (6.9) represents the graphical representation of clustering error with respect to different number of genes and multiple DWD levels. It shows the number of samples that are not clustered accurately, while obtained accurate clustering when involved range of genes *g={100-300}* processing with DWD level *lv={2, 3}*. Figure (6.10) shows Voronoi diagram to the result of two classes clustering accurately of the microarray samples.

On the subject of the computational aspects of the miDWD approach, the calculation of the DWD spectral distortion measure mostly engages in the computations of the DWD algorithm in partitioning the expression signal as data space into different frequency sub-bands. This partitioning reduces the number of data objects in expression signal while inducing small errors due to fluctuation of expression. The high frequency regions of the signal related to the regions of the expression signal behaviour where there is a quick change in the expression distribution. The low frequency regions of the signal related to the part of the features where the expression content is concentrated. As a result, simultaneous information on both the frequencies partition and the spatial distribution of these frequencies transformed as wavelet coefficients.



a- MSE of different DWD level



b- MSE of different number of genes

Figure 6.8 Performance analyses with respect to different number of genes and multiple DWD levels on leukaemia dataset.

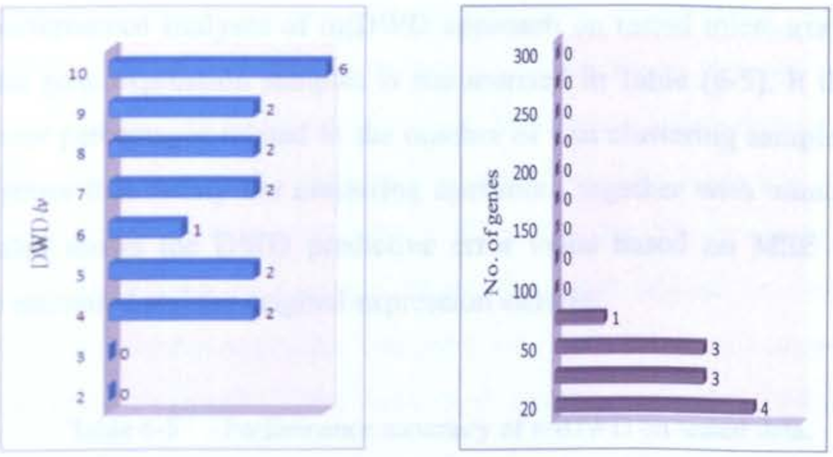


Figure 6.9 Clustering errors with respect to different number of genes and multiple DWD levels on leukaemia dataset

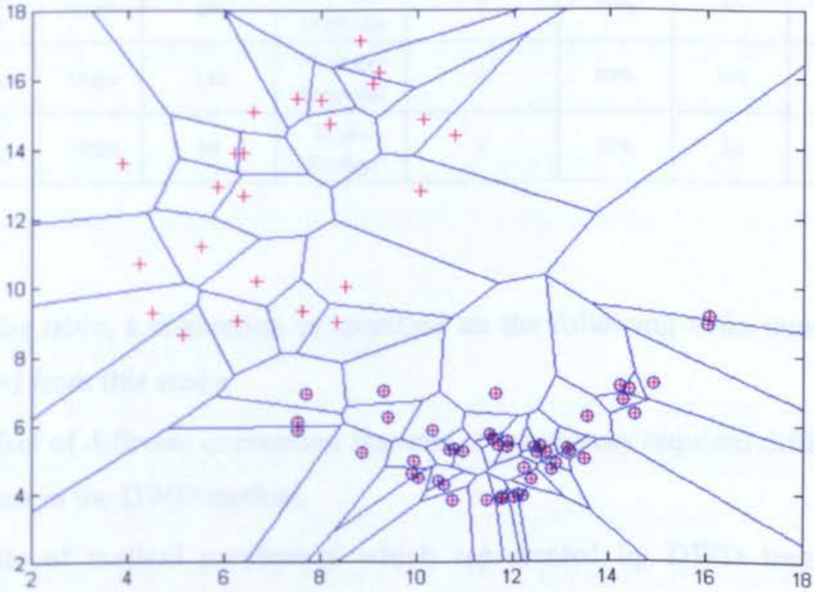


Figure 6.10 Voronoi clustering of two for Leukaemia dataset using lv=2, g=100

The experimental results and comparisons of model prediction have shown the performance of the GSP methods. The miDWD method can be implemented easily and especially concerning the flow of calculations in computing the DWD framework embedded with the VQ method. This feature is very efficient for real-time implementation compared to LPC method. This method is a powerful tool in the data clustering since it outperforms any unsupervised method.

6.4.2- Discussion

The performance analyses of miDWD approach on tested microarray datasets to clustering the gene expression samples is summarised in Table (6-5). It illustrates the clustering error percentages related to the number of non clustering samples, minimum number of genes that satisfy the clustering operation, together with number of DWD level, and also shows the DWD predictive error value based on MSE computation between the estimated and the original expression sample.

Table 6-5 Performance summary of miDWD on tested data.

Study	Type of disease	No. of genes	Total no. of samples	Goal	Samples non clustering	Accuracy %	Min no. of genes	DWD level	Predictive error
Golub	Leukaemia	7129	72	47 ALL 25 AML	0	100%	100	2	0.956
Alone	Colon	2000	62	40 tumour 22 normal	2	97%	25	3	3.7
Iizuka	Hepato-cellular	7129	60	20 sick 40 healthy	7	90%	50	8	4.2
Singh	Prostate	12600	136	77 tumour 59 normal	8	94%	175	6	3.42
Nutt	Gliomas	12625	50	28 glio 22 oligo	4	92%	50	9	1.3

From the table, a discussion is specified on the following main conclusions that can be derived from this study:

i- The effect of different expression features of microarray required different level of partitions in the DWD method.

ii- The role of method parameters which represented by DWD level. Since the microarray has different size, therefore different level of DWD performed in the analysis. The complexity of partitioning signals in DWD method affect on computation, and that will increase proportionally with the raising of levels.

iii-Concluding that wavelet based on filters could be useful for reconstructing signal without loss the original spatial features, due to small value of predictive error.

iv- The wavelets provide a proper estimation of coefficients model for the analyses of a variety of expression levels, hence to obtain best quantisation.

In most cases, the miDWD methods provided the finest ratios of clustering against other traditional clustering process. Generally, most classification and clustering methods have required a predefined gene sample similarity or distance metric, while their achievement performance rely on how well that metric reflect the real relationship among samples. In addition, the traditional methods required extensive preprocessing and denoising as the microarray data are sensitive to noise. The powerful of miDWD has its potential to analyse the genomic dataset in global fashion hence the effect of local noises can be slightly negligible comparatively and therefore it causes suitability for analysing genomic signals in the area of dimension reduction and classification problems.

6.5- Application of miFD clustering

This section presents the results obtained from applying the FD approach explained earlier in chapter 5 on the selected microarrays. The application is performed with multiple runs to achieve the analysis using different values of FD level with different number of gene involved to detect the optimum value with less fractal in estimated gene expression signal. Moreover an exploration of different combination of parameters that affect on clustering performance for each dataset will be introduced.

6.5.1- miFD on Leukaemia dataset

Following to the pre processing step for selecting number of gene from Leukaemia dataset, the miFD algorithm will begins. Fractals dimensions considered as an indicator for an infinite set of points in the microarray to test the distribution of expression samples data without require an assumption of an average density. It computed from the expressions vector of microarray sample and the scale beyond which the fractal dimension is close to the physical dimension of the sample. It identified the scale of several degrees of expression fluctuation complexity and then finds the cluster of distribution sample points based on computed scales.

The results for fractals dimension and the performance of these based on multiple run concerning different number of involved genes is shown in Figure (6.11). Iteration is used to calculate the box counting dimension data, box sizes and box numbers in order to find the best regression equation satisfied by the model. The figure illustrates that at *g=100* genes obtain minimum value of average FD equal to *FD=0.192* which provide accurate clustering. It presents an observation that the lowest average fractal dimension provides the lower clustering error with respect to number of genes involved. Figure (6.12) shows the Logarithmic plots to the local representation of the sample 9 between box number and box sizes were used to compute the regression line equation which is [y=0.0132x+5.7], therefore the slope of this regression line provides the box counting dimension. It may perhaps identify as coefficients and explore from the formula which is equal to (0.0132). It is more likely to establish significant fractal correlation dimension to the coefficients, based on the possibility that correlation dimension has its own range in the boundaries of fractal dimension. Therefore it derived from the correlation integral which is a cumulative correlation function that measures the fraction of points in the two dimensional space.

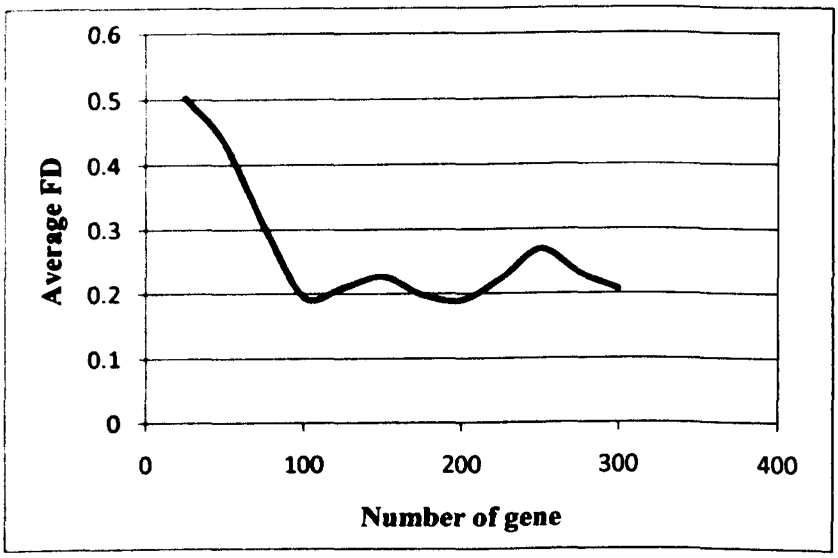


Figure 6.11 Performance analyses of miFD method of Leukaemia dataset

In order to find the cluster, the process to compute the nearest distance based on the average between the boundary points is considered as a barrier level which is equal to *β=0.8755*. Figure (6.13) shows clustering samples in dataset using *g=100* genes, Note that the perfect value of barrier level is due to the fact that the accurate calculation depends on the variation in the complexity comes from large number of iterations that are required until convergence. It shows the result of two classes clustering of the microarray samples accurately.

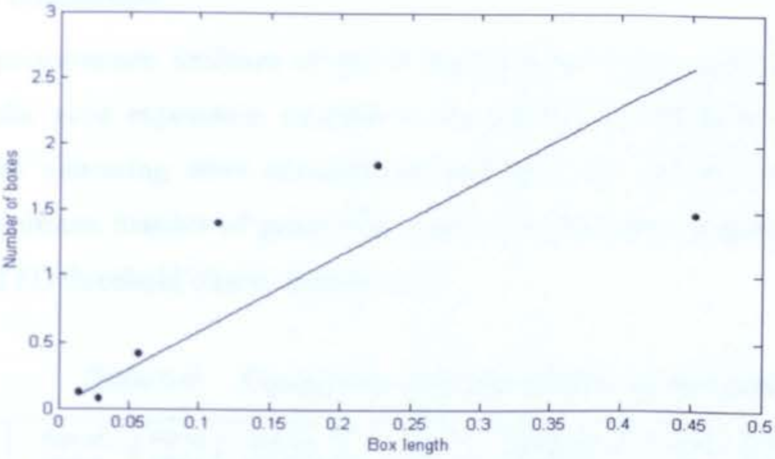


Figure 6.12 Local representation of sample 9 of Leukaemia dataset

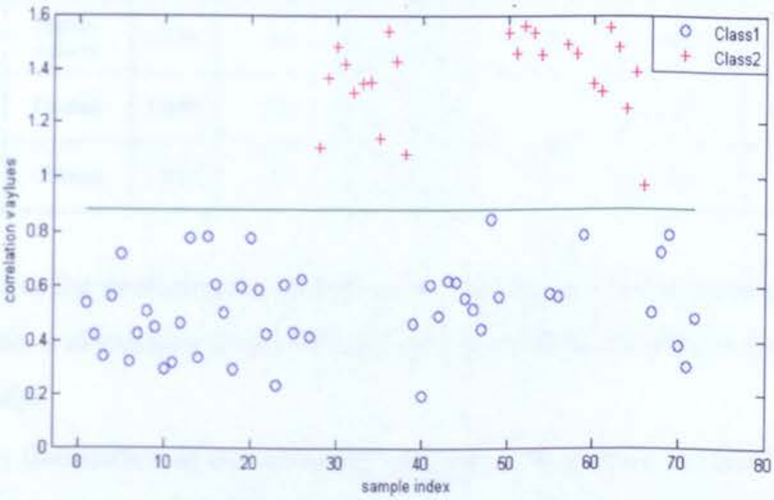


Figure 6.13 miFD clustering of Leukaemia dataset using $g=100$, where $\beta=0.8755$

The complication of the miFD approach is dominated by the computing aspects of the calculation of the FD for the total area covered by the grid array elements that will converge to the measure of the curve, i.e. for square boxes elements is to obtain the standard box counting dimension as based on Eq. 5.7. An inspection concerning the sufficient number of boxes covering the curve in the area surrounding the expression signal is achieved through the iteration process. In the course of reduces the number of boxes object to expression signal, it inducing small errors due to fluctuation of expression. As a result, the FD algorithm counting the number of boxes required to cover the curve relatively with box sizes and then establish log-log plot. The slope of a linear fit to the plotted curve approximates to the fractal dimension.

6.5.2- Discussion

The performance analyses of miFD approach on tested microarray datasets for clustering the gene expression samples is summarised in Table (6-6). These results illustrate the clustering error percentages related to the number of non clustering samples, minimum number of genes that satisfy the clustering operation, together with the value of FD threshold cluster barrier level.

Table 6-6 Comparative summary of miFD on tested data.

Study	Type of disease	No. of genes	Total no. of samples	Goal	Samples non clustering	Accuracy %	Min no. of genes	β
Golub	Leukaemia	7129	72	47 ALL 25 AML	0	100%	100	0.87
Alone	Colon	2000	62	40 tumour 22 normal	1	98%	75	0.55
lizuka	Hepato-cellular	7129	60	20 sick 40 healthy	5	92%	100	0.20
Singh	Prostate	12600	136	77 tumour 59 normal	9	93%	100	0.92
Nutt	Gliomas	12625	50	28 glio 22 oligo	3	94%	100	0.56

To assess the performance pertaining to fractal dimension estimation as described in Table (6-6), a discussion in the following is specifying the effects that can be derived from the study:

i- The fluctuation of the different expression amplitude required different values concerning number of boxes and its resolution size to estimate the fractals.

ii- It is noted that due to large data points, the fractal dimension based on box counting acts effectively. Since it require large number of data points to estimate the fractal values.

iii- The fractal dimension is evaluated by itself through the correlation dimension approach by best fitting the log-log plot curve. The accuracy of the determination is affected by the finite size of the dataset. The linearity of the curve reveals the self similarity of the expression at successive scales.

iv- As a results presented so far, it is likely that the FD approach achieves efficient performance regarding its ability to provide a proper estimation of fractals model for the analyses of a variety of expression levels.

Accordingly, the miFD method in most cases, produce efficient ratios of clustering, i.e. corresponding to 92% as a minimum clustering error. Generally, FD assessment is a best indicator of the spread of the data that can be used as indicator for the quantity of hidden information in the dataset. Noticing that uses the FD to measure of the physical complexity of different expression levels with the same calculation amount. During the calculation and over the linear range of log-log plot, the fractal dimension can describe the roughness of the expression level.

The power of miFD is its potential to analyse the genomic expression data in large-scale mode hence the effect of local noises can be slightly insignificant comparatively due to the linearity of the log and therefore it causes suitability for analysing genomic signals in the area of dimension reduction and classification problems. The miFD approach in most cases provided better clustering against other traditional clustering process which have require a predefined gene sample similarity. While miFD approach rely on generating a grid that dividend gene signal into smaller parts that will reflect the real relationship among samples parts. The strength of miFD is its potential in analysing the genomic dataset in comprehensive mode because the formula of their correlation dimensions are concerning to the smaller parts of gene signal behaviour that causes any minor change in the signal, varies the complete fractal dimension value. Hence this approach is suitable for analysing genomic signals in the areas where there are classification problems. It is recognized that the box-counting dimension calculation seems to give the best result because it covers the whole gene signal in dataset. Using boxes of the small size causes that the coverage is precisely occupied more parts without losses information there.

6.6- Cluster validation methods

Cluster validation process aims at evaluation of the approaches by satisfying the clustering target of genomic data. There are two validation methods of the clustering results: internally, by evaluating the quality of a clustering result based on statistical properties that can also be used for selecting the best clustering result when comparing different clustering methods, and externally, by comparing the level of agreement of a clustering result with an external partition.

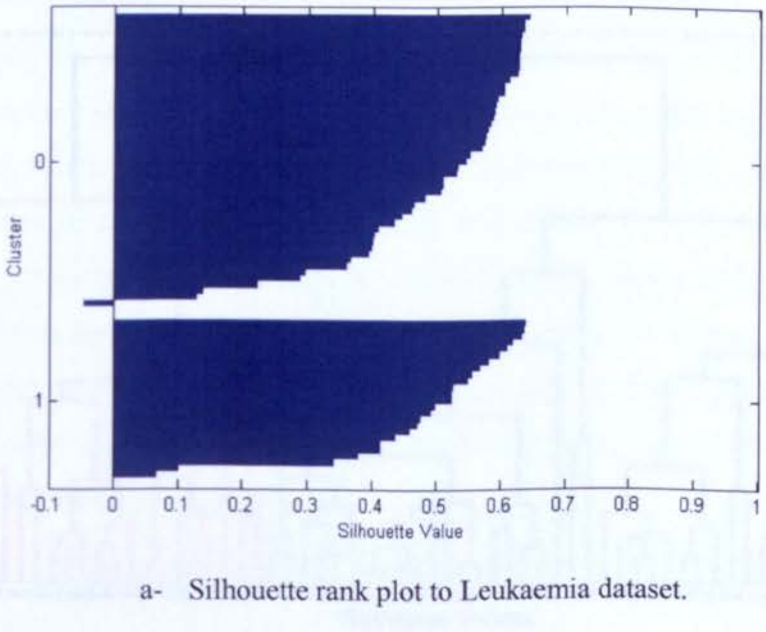
In this section, well-known internal and external validation methods, respectively Silhouette index and Rand index are used to evaluate the genomic signal processing clustering approaches.

6.6.1- Silhouette index

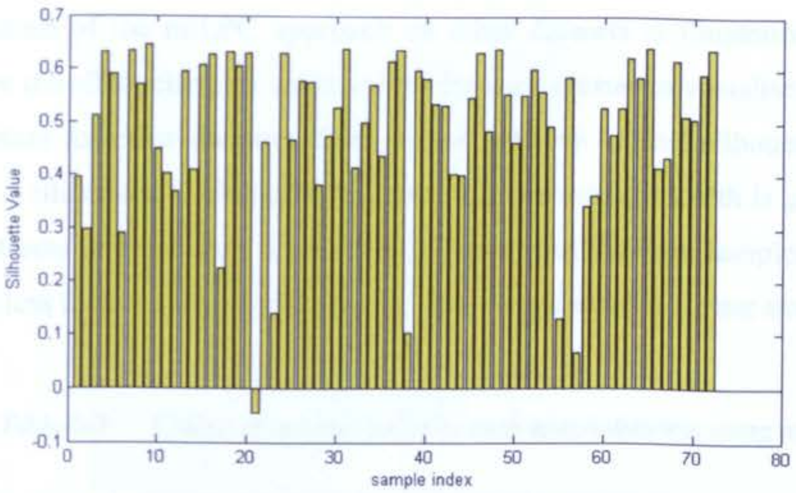
This method is applied to test the performance of GSP clustering on the selected datasets. To measure the global goodness of clustering using the Silhouette index, two parameters are required to be calculated. They are the Silhouette Width range, which is between 1 and -1, and the Average Silhouette Width (*ASW*). If the value of the Average Silhouette Width is greater than 0.5 it indicates that clusters achieved a reasonable partition of the data. However, if its value is lower than 0.2, it expresses that the data do not exhibit cluster structure.

6.6.1.1- miLPC approach

Evaluating the miLPC approach for use in clustering genomic data samples is carried out on the selected microarray datasets. Concerning the Leukaemia dataset, Figure (6.14a) shows the silhouette values for each cluster group, ranking them in decreasing order to allow rapid visualization and assessment of cluster structures. Figure (6.14b) presents the value silhouette index for each sample.



a- Silhouette rank plot to Leukaemia dataset.



b- Silhouette value to each sample in Leukaemia dataset

Figure 6.14 Silhouette plot to Leukaemia dataset

From the observation of Figure (6.14a) we see silhouette width values are generally positive. There is only one exception (sample 21) which has a small negative value, although it is in the range of silhouette width. Since the global silhouette index that represented the average value of silhouette width value is equal to (*ASW=0.49*), this indicates that the formed clusters are suitably to recover all the samples in the dataset. Figure (6.15) shows the dendrogram of the distribution of cluster hierarchical procedure in a tree diagram. It contains structural of organisational information associated to the samples similarity.

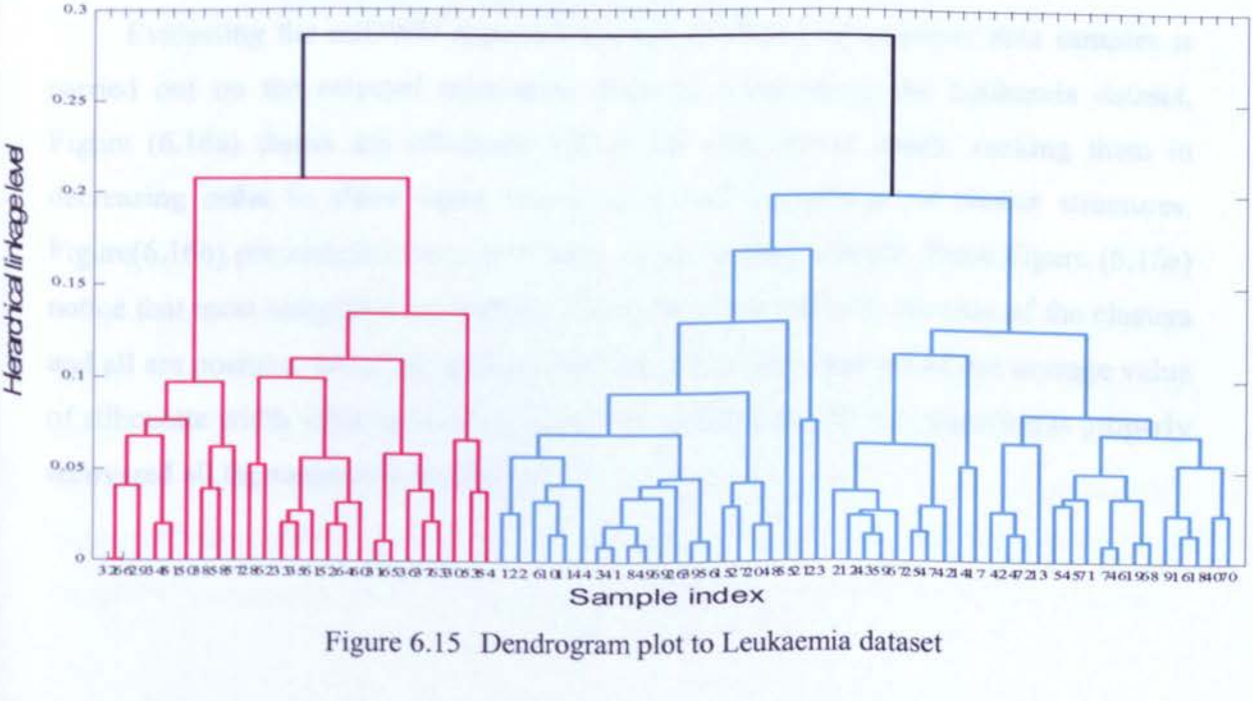


Figure 6.15 Dendrogram plot to Leukaemia dataset

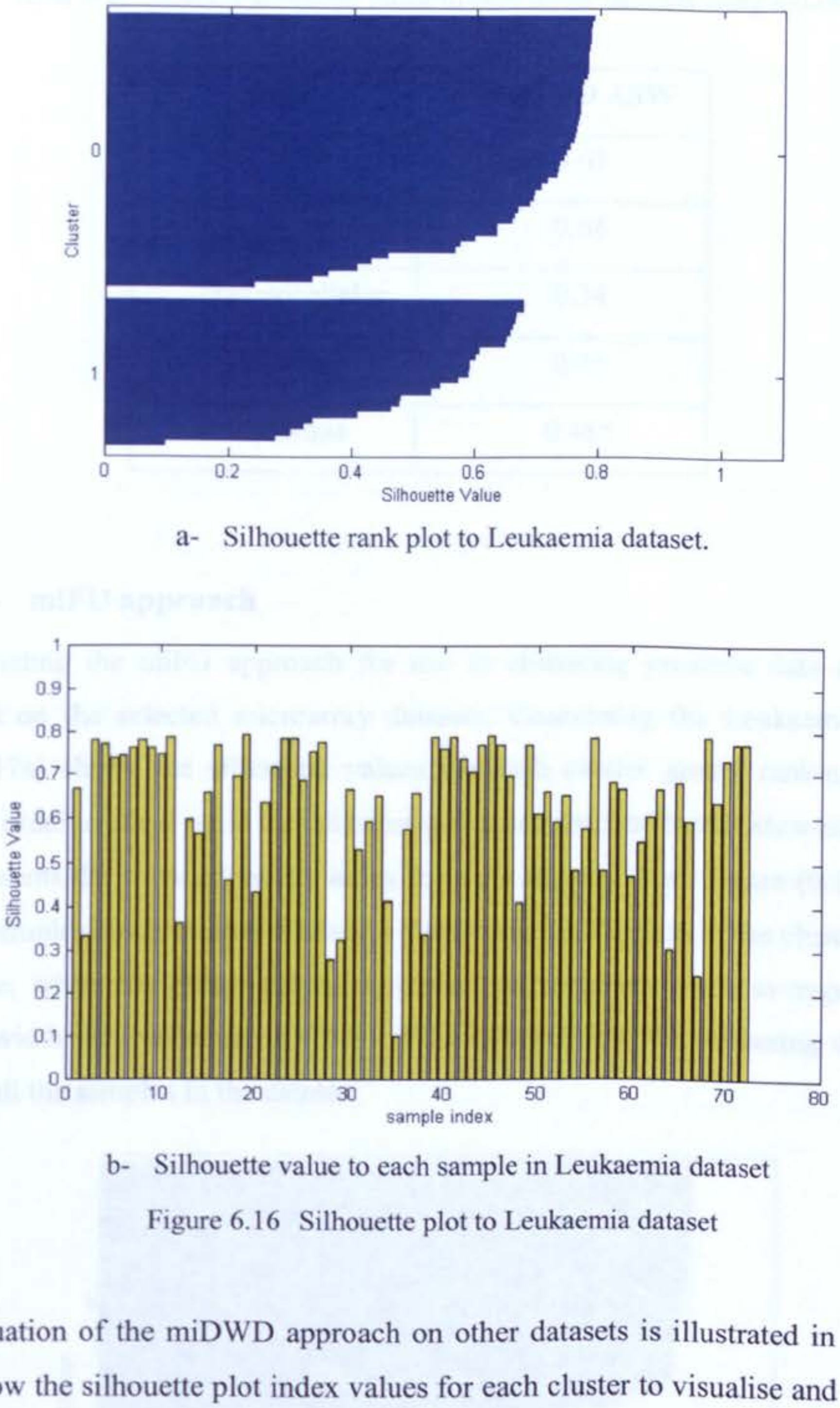
Evaluation of the miLPC approach on other datasets is illustrated in appendix. Figures show the silhouette plot index values for each cluster to visualise and assess the cluster structure for other datasets. Comparison between Global silhouettes indexes to the datasets is illustrated Table (6-7). It shows that the average width is greater than 0.5 for some datasets that indicates a reasonable partition of the data samples, while others has value of less than 0.2 would indicate the data do not exhibit cluster structure.

Table 6-7 Global silhouette index to each tested datasets using miLPC

Dataset	miLPC ASW
Leukaemia	0.49
Colon	0.58
Hepatocellular	0.145
Prostate	0.35
Gliomas	0.27

6.6.1.2- miDWD approach

Evaluating the miDWD approach for use in clustering genomic data samples is carried out on the selected microarray datasets. Concerning the Leukemia dataset, Figure (6.16a) shows the silhouette values for each cluster group, ranking them in decreasing order to allow rapid visualization and assessment of cluster structures. Figure(6.16b) presents the value silhouette index for each sample. From Figure (6.16a) notice that most samples have suitable silhouette width value in the case of the clusters and all are positive, while the global silhouette index that represented the average value of silhouette width value is equal to (0.63). This indicates that the clustering is properly recovered all the samples in the dataset.



a- Silhouette rank plot to Leukaemia dataset.

b- Silhouette value to each sample in Leukaemia dataset

Figure 6.16 Silhouette plot to Leukaemia dataset

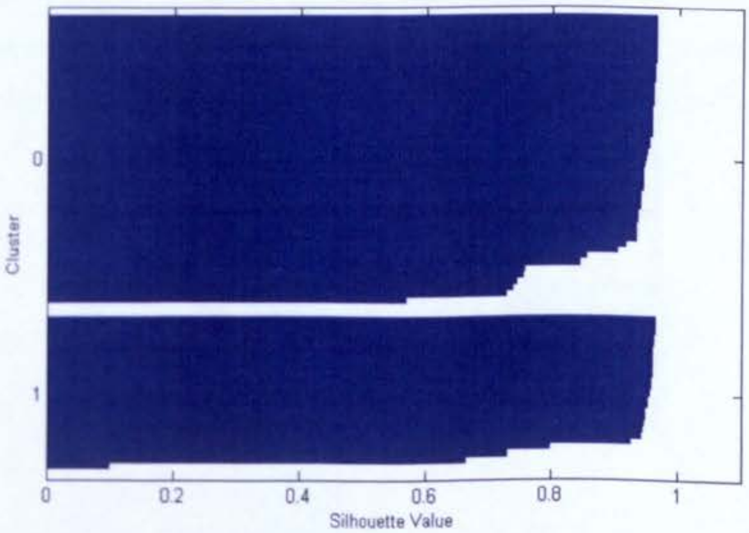
Evaluation of the miDWD approach on other datasets is illustrated in appendix. Figures show the silhouette plot index values for each cluster to visualise and assess the cluster structure for other datasets. Comparison between Global silhouettes indexes to the datasets is illustrated Table (6-8). It shows that the average width is greater than 0.5 for some datasets that indicates a reasonable partition of the data samples, while Hepatocellular dataset has value of 0.34 would indicate that the clustering process based on miDWD approach is almost has proper structure than the previous miLPC approach.

Table 6-8 Global silhouette index to each tested datasets using miDWD

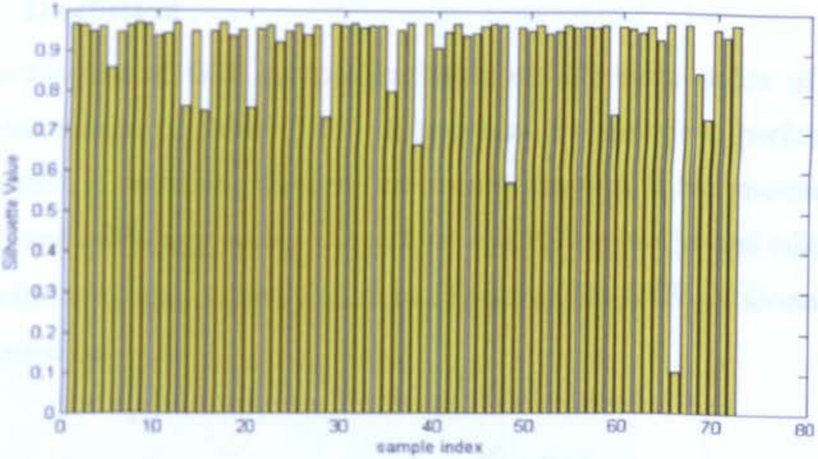
Dataset	miDWD ASW
Leukaemia	0.63
Colon	0.64
Hepatocellular	0.34
Prostate	0.46
Gliomas	0.485

6.6.1.3- miFD approach

Evaluating the miFD approach for use in clustering genomic data samples is carried out on the selected microarray datasets. Concerning the Leukaemia dataset, Figure (6.17a) shows the silhouette values for each cluster group, ranking them in decreasing order to allow rapid visualization and assessment of cluster structures. Figure (6.17b) presents the value silhouette index for each sample. From Figure (6.16a) notice that most samples have suitable silhouette width value in the case of the clusters and all are positive, while the global silhouette index that represented the average value of silhouette width value is equal to (0.91). This indicates that the clustering is properly recovered all the samples in the dataset.



a- Silhouette rank plot to Leukaemia dataset.



b- Silhouette value to each sample in Leukaemia dataset

Figure 6.17 Silhouette plot to Leukaemia dataset

Evaluation of the miFD approach on other datasets is illustrated in appendix. Figures show the silhouette plot index values for each cluster to visualise and assess the cluster structure for other datasets. Comparison between Global silhouettes indexes to the datasets is illustrated Table (6-9). It shows that the average width is greater than 0.5 for all datasets that indicates a reasonable partition of the data samples. It would indicate that the clustering process based on miFD approach is almost has proper structure than others approaches.

Table 6-9 Global silhouette index to each tested datasets using miFD

Dataset	miFD ASW
Leukaemia	0.91
Colon	0.87
Hepatocellular	0.74
Prostate	0.69
Gliomas	0.78

6.6.1.4- Discussion

The evaluation of GSP approaches based on silhouette index of experimental results are summarise in Table (6-10). It provides a comparison performance of the proposed method in clustering the gene expression samples in the microarray. It shows that miDWD and miFD approaches outperform miLPC approache and other well known state-of-art clustering approaches. Figure (6.18) shows the ASW performance summary of GSP on tested data.

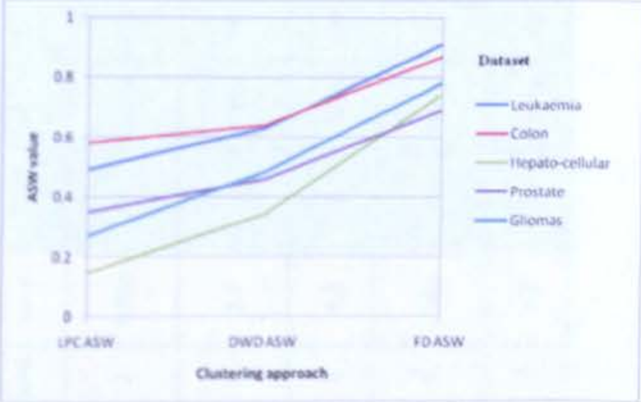


Figure 6.18 ASW analysis to GSP for tested microarray dataset

6.6.2- Davies-Bouldin (DB) index

This method is based on the maximization of the distances between clusters while minimizing the distances within a cluster itself. A DB-index is determined as a function of the ratio of the sum of the distances within a cluster to the distance between clusters: the smaller the DB- index, the greater the quality of the achieved clustering. Figure (6.19) shows the DB-indices for all tested datasets. It shows that miFD consistently achieves significantly better results than the other approaches.

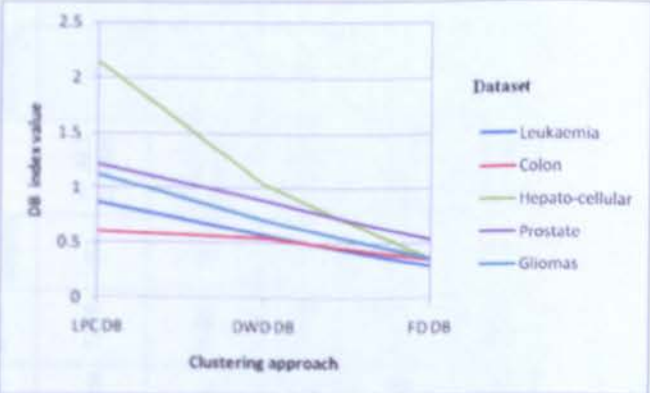


Figure 6.19 DB analyses of GSP techniques for all tested microarray datasets

Table (6-10) ASW comparison performance summary of the GSP approaches

Datasets					miLPC approach					miDWD approach					miFD approach				Validation					
Study	Type of disease	No. of genes	Total No. of samples	Goal	Samples non clustering	Accuracy %	Min No. of genes	LPC order	Predictive error	Samples non clustering	Accuracy %	Min No. of genes	DWD level	Predictive error	Samples non clustering	Accuracy %	Min No. of genes	FD	LPC ASW	DWD ASW	FD ASW	LPC DB	DWD DB	FD DB
Golub	Leukemia	7129	72	47 ALL 25 AML	0	100%	75	34	0.838	0	100%	100	2	0.956	0	100%	100	0.87	0.49	0.63	0.91	0.865	0.574	0.292
Alone	Colon	2000	62	40 tumour 22 normal	3	95%	100	32	2.97	2	97%	25	3	3.7	1	98%	75	0.55	0.58	0.64	0.87	0.605	0.553	0.359
Iizuka	Hepato-cellular	7129	60	20 sick 40 healthy	14	76%	125	29	0.528	7	90%	50	8	4.2	5	92%	100	0.2	0.145	0.34	0.74	2.14	1.03	0.366
Singh	Prostate	12600	136	77 tumour 59 normal	14	90%	125	28	0.92	8	94%	175	6	3.42	9	93%	100	0.92	0.35	0.46	0.69	1.214	0.885	0.538
Nutt	Gliomas	12625	50	28 glio 22 oligo	5	90%	75	26	1.47	4	92%	50	9	1.3	3	94%	100	0.56	0.27	0.485	0.78	1.119	0.706	0.36

6.7- Conclusion

In this section, a complete comparison performance is presented between the experimental results obtained from analysing the selected datasets using the proposed GSP approaches. These are also compared with other well known state-of-art clustering approaches and summarised in Table (6-11). These results show the performance of the proposed methods in clustering the microarray gene expression samples. It can be seen, first, that the three techniques outperform the existing state of the art methods. Second, the miDWD and miFD methods outperform the miLPC method. Finally, cluster evaluation suggests that miFD is the best method for microarray data clustering.

Table 6-11 Comparative summary of GSP methods on tested data.

Method	Author	Leukaemia	Colon	Hepatocellular	Prostate	Gliomas
T-test	Golub,1999[8]	85%				
T-test	Alone,1999[106]		87%			
FLC	Iizuka, 2003[107]			93%		
kNN	Singh,2002 [108]				90%	
kNN	Nutt,2003 [109]					86%
PAM	Tibhirani,2002[110]	95%	83%	59%		67%
MARS	Mukkamala,2005 [111]	85%	80%		92%	
CART	Mukkamala,2005 [111]	92%	95%		96%	
LGP	Mukkamala,2005 [111]	95%	85%		96%	
RF	Mukkamala,2005 [111]	100%	90%		88%	
PLSLD	Nguyen,2002[112]	97%	92%			
KPCA	Liu,2005[113]	97%	100%			
FJC	Jong,2003[114]	91%	54%			
Two-way	Chanda,2006[115]	96%	88%			
SVM	Furey,2000[116]	94%	90%			
MRMR	Ding,2004[117]	100%	94%			
GA/SVM	Huerta,2006[118]	100%	99%			
P-ICR	Huang, 2006[119]	95%	86%	62%		74%
miLPC		**100%**	**95%**	**76%**	**90%**	**90%**
miDWD		**100%**	**97%**	**90%**	**94%**	**92%**
miFD		**100%**	**98%**	**92%**	**93%**	**94%**

CHAPTER 7

Conclusion and Future work

In this chapter we present the conclusion of this work and outline some of the limitations of this study and suggestions for future work in this evolving area.

7.1- Conclusions

In this thesis we have presented three GSP approaches for enhanced microarray data clustering. These are linear predictive coding, wavelet and Fractal methods. The first three chapters presented the details of these methods with last chapter describing the performance analysis of these methods. The aim of the work was also to provide a comprehensive description for the most common traditional clustering approaches compared with GSP approaches for clustering gene expression samples.

In summary, the contribution of the thesis are summarised as following:

1. Comprehensive and Review of microarray data clustering

A comprehensive study was carried out to understand the traditional and current clustering procedures by searching and studying extensively. It starts from statistical view framework, estimation and prediction algorithms in biological orientation and advances in digital communication and signal processing approaches that finds how can it adapted in genomic region and other sources cited in the thesis.

A comprehensive review on state of art of genomic clustering approaches was performed. This review involved studying past and existing research methods, and examined their advantages and shortcomings. From these, we selected the three GSP methods for the study

2. Design of a GSP clustering and toolbox

Three clustering approaches based on advanced digital signal processing methods (LPC, DWD and FD) combined with vector quantisation algorithm is presented. A GSP based toolbox for application in microarray data samples using these methods was designed and implemented (miLPC, miDWD and miFD). These can read the data from any microarray gene expression samples and produce a predictive coefficients array relative to the microarray data that can be quantised in discrete levels, and consequently represents the clustering output. The other operations such as pre-processing the microarray data and normalization are also embedded in this toolbox. The design of the toolbox was based on MATLAB™ that can provide easy usage and procedures to modify.

3. Comparative performance analysis

The thesis also presents a comparative performance analysis of the three GSP methods of clustering microarray on different microarray datasets from the literature. Two well known validation methods (Silhouette and Davies- Bouldin index) have been used to evaluate the GSP clustering results. Internally, to evaluate the quality of a clustering result, and externally, by imitating the level of agreement of a clustering result with an external partition. In conclusion, the miDWD and miFD outperformed all the test datasets with more clustering accuracy compared to other methods. However, the local features of the gene expression signals were better clustered using the miDWD method compared to the miFD.

7.2- Limitation and future works

The area of GSP is considered as an emerging field of modern genomic analysis. In this section we outline some of the limitations of each GSP method presented and propose some future research directions and work in this evolving area.

7.2.1- Limitation of the study

1. Statistical methods

The traditional statistical methods had the capability of detecting the variation in a variety of datasets under a variety of conditions. The amount of variation in gene expression samples in the dataset, as well as the period sometimes cannot affect the detection, when they are within reasonable limits of expression. Very low or very high expressions are difficult to detect, especially in cases where the sampling is inadequate. Within those limits these methods also has high sensitivity and specificity for individual clustering of correlated samples. This results causes to the disadvantage of aggregation of the dominant frequencies in the dataset and the approximation in predictions. However, the GSP methods presented are able to detect the variation in the datasets. This can happen when the frequencies of expression samples are distinct enough and the datasets have a high signal to noise ratio.

2. Microarray LPC method

The LPC analysis methods were introduced and explained in chapter3. Further investigation of vector quantisation to LPC coefficients strength and accuracy for microarray clustering were performed as well. LPC coefficients are used to provide a representation of the spectral gene sample signals. Thus efficient representation of LPC parameters is main issue in the coding analysis. The most often used is line spectral frequencies which have two characteristics that make the coefficients willing to efficient quantisation: their ordering which relates to the stability condition and their localization. The key point of VQ modelling is to derive a codebook which is commonly achieved by using a clustering technique. From VQ notion, the quantiser has an equivalent reference codebook which minimizes the overall distortion, it is determined by first encoding the source vector into their corresponding partition regions and then taking the centroid of all the source vectors assigned to each particular region to achieved the clustering process.

Concerning to the tested datasets, the miLPC approach is able to predict results that are comparable with the originally identified genes sample. Thin method is affected by the signal to noise ratio in the analysis of gene samples. The amount of noise in the microarray data affects both the ability of the analysis to calculate the predictive coefficient and lowers the sensitivity of the methods. However, one disadvantage of linear prediction coding is that it requires a large amount of computations for analyzing the data in relative with increasing the order of coding. While the second disadvantage of the LPC spectral in the analysis (gene sample signal with a prevalent partial structure) is that it will tend to cover the spectrum of gene signal as tightly as possible, and will under certain conditions descend down to the level of residual noise in the gap between two partials. This will happen whenever the space between partials is large, as in highly gene expression, and when the order is high.

3. Microarray DWD method

The main idea behind the application of DWD approaches was to represent gene profile into a set of orthonormal wavelet basis functions in a time-frequency domain to extract the spectral features of microarray data in order to enhance clustering. This method has the ability to remove the random data as a noise from the microarray data itself. This happen because the gene expression profile produced by different technologies were contaminated with errors. Furthermore, wavelet allows the decomposition of input gene sample signal at different scales and levels of detail, which might be achieved improvement in the quality through the applications. DWD is also good localization both in time and spatial frequency, since wavelet analysis combines the concept of scale into the wavelet equations, therefore it is appropriate to resolve the sample transient of gene expression data. Then choosing the number of scales is important issues as the computation of the mother wavelet will start from high frequencies and proceed towards low frequencies. While increasing the value of scale, the wavelet will dilate. From the observation, lower scales (high frequencies) have better scale resolution which corresponds to low frequency resolution, and hence small-scale wavelet coefficients are fundamental to encode that information. From the results in this thesis, it is noticeable that wavelets are better suited to the analysis of different gene expression signals in small basis functions or wavelet filters while use of large basis produced distortion error. We also need to emphasize that DWD is not translation invariant. Therefore, the proper performance is the ability or need to select wavelet basis functions for particular applications. Hence further work can be done in these areas.

4. Microarray FD method

Clustering based on the usage of the fractal dimension was presented. The algorithm address the problem of discovering clusters of points according to the effect they have on the FD of the clusters. Each vector of points in the dataset can be mapped to a local representation consisting of a density coefficient and a dimensionality coefficient. Mathematically, fractal dimension is used to give a dimension of the statistical measure of the geometry of a cloud of points and can be assigned to any arbitrary dataset

A limitation of this partitioning scheme, however, is the use of range size. Many regions of a given expression signals are too complex to be partitioned into boxes square, because the variations in signal may not possess corresponding size domain box blocks that closely match the regions. However, further limitation come from the difficulty that there is multi different combination of expression level which results in the same displacement and solved by applying two-dimensional vectors of analysis, which is probably required further study in the future

7.2.2- Future work

There are several issues that could be studied further in this area and to evaluate further the potential use of the methods presented:

1. A need for an optimal criterion while searching for the optimal-evaluation-environment in all the proposed methods. These include for example a suitable learning approach to optimise the choice of the quantisation level in VQ selection procedures.
2. Prediction are limited to a fixed dimension (the fractal dimension calculated using GP algorithm), this need further investigation and future work.
3. Symmetrical environments while searching for an optimal evaluation environment - poor results near sharp areas of the system's behaviour. These issues are especially important in the miDWD and miFD methods. In the miFD method, the requirement of optimal grid resolution is important in clustering analysis.
4. Further testing on larger microarray datasets and disease types.
5. Selection of other advanced GSP and digital communications methods currently applied in other domains. For example, adding adaptation mechanisms to the presented methods.

References

[1] Istepanian R., (2003), "Microarray image processing: current status and future directions", IEEE transactions on NanoBioscience, Vol. 2, No. 4, pp. 173-175.

[2] Hand D. J. and Heard N. A., (2005), "Finding groups in gene expression data", J of Biomedicine and Biotechnology, No. 2, pp. 215-225.

[3] Draghici S., (2003), "Data analysis tools for DNA microarrays", Chapman & Hall/CRC.

[4] Pham T. D. *et al.*, (2006), "Analysis of microarray gene expression data", Current Bioinformatics, Vol. 1, No.1, pp. 37-53.

[5] Jiang D. *et al.*, (2004), "Cluster Analysis for Gene Expression Data: A Survey", IEEE Transactions on Knowledge and Data Engineering (TKDE), Vol. 16, No. 11, pp. 1370-86.

[6] King H.C. and Sinha A. A., (2001), "gene expression profile analysis by DNA microarreys", Journal of the American Medical Association, Vol. 286, No. 18, pp. 2280-8.

[7] Lockhart D. J. and Winzele E. A., (2000), "genomics, gene expression and DNA arrays", Nature,.Vol. 405, 15June., pp. 827-36.

[8] Golub T. *et al.*, (1999), "Molecular classification of cancer: class discovery and class prediction by gene expression monitoring", Science, Vol. 286, No. 15, pp. 531-537.

[9] Nadler S. T. *et al.*, (2000), "The expression of adipogenic genes is decreased in obesity and diabetes mellitus". Proc. of the National Academy of Science USA, Vol. 97,No. 21, pp. 11371-76.

[10] Elbein S.C., (2002), "Perspective: the search for genes for type 2 diabetes in the post-genome era", The endocrine Society, Vol.143, No. 6, pp. 2012-18.

[11] Murphy D., (2002), "Gene expression studies using microarrays: principles, problems, and prospects", Advances in physiology education, Vol. 26, No. 4. ,pp. 256-270.

[12] Ye S. and Day I., (2003), "Microarrays and Microplates: Applications in Biomedical Sciences", BIOS Scientific publishers.

[13] Nguyen D. *et al.*, (2002), "DNA microarray experiments: biological and technological aspect", Biometrics, Vol. 58, No. 4, pp. 701-717.

[14] Wilson A. *et al.*, (2002), "The microarray: potential applications for ophthalmic research", Molecular vision, Vol. 8, 17Jul, pp. 259-270.

[15] http://atlasgeneticsoncology.org/Deep/ComparCancerCytogID20011.html, (Accessed 20May2009).

[16] Weiner M. P. and Hudson T. J., (2002), "Introduction to SNPs: Discovery of markers for disease", BioTechniques, Vol. 32, June, pp. 4-13.

[17] Chen J. *et al.*, (2003), "How will Bioinformatics impact signal Processing research," IEEE Signal Processing Magazine, Vol.20, No. 6, pp16-26.

[18] Dougherty E. R. *et al.*, (2005), "Research Issues in Genomic Signal Processing," IEEE Signal Processing Magazine, Vol. 22, No. 6, pp. 46-68

[19] Antoniol G. *et al.* (2005), "Linear predictive Coding and Cepstrum coefficients for mining time variant information from software repositories", Int. workshop on mining software repositories, 17 May 2005, Missouri, USA, pp. 1-5.

[20] Lio P., (2003), "Wavelets in bioinformatics and computational biology: state of art and perspectives", Bioinformatics, Vol. 19, pp. 2–9.

[21] Kumaraswamy K. and Megalooikonomou V., (2004), "Fractal dimension and Vector quantization", Information processing letters, Vol. 91, pp. 107-113.

[22] Young R. A., (2000), "Biomedical discovery with DNA arrays", Cell, Vol.102, July 7,2000,pp. 9-15.

[23] Fujita A. *et al.*, (2006),"Evaluating different methods of microarray data normalization", BMC Bioinformatics, Vol. 7, pp. 469-480.

[24] Thygesen H. and Zwinderman A., (2004), "Comparing transformation methods for DNA microarray data", BMC Bioinformatics, Vol. 5, pp. 77-89.

[25] Tan P. *et al.*, (2006), "Introduction to data mining", Addison Wiley NY.

[26] Guyon I. and Elissee A., (2003), "An Introduction to Variable and Feature Selection", Journal of Machine Learning Research, Vol. 3, pp. 1157-82.

[27] Reimers M., (2005), "Statistical analysis of microarray data", Addiction biology; Vol.10, No. 1, pp. 23-35.

[28] Troyanskaya O. *et al.*, (2002), "Nonparametric methods for identifying differentially expressed genes in microarray data", Bioinformatics, Vol.18,No.11, pp. 1454-61.

[29] Levner I, (2005), "Feature selection and nearest centroid classification for protein mass spectrometry", BMC Bioinformatics, Vol.6, pp. 68.

[30] Jaeger J. *et al.*, (2003), "Improved Gene Selection for Classification of Microarrays", Pacific Symposium on Biocomputing 8, pp. 53-64.

[31] Alter O. *et al.*, (2000), "Singular value decomposition for genome-wide expression data processing and modeling", Proceedings of the National Academy of Science PNAS-USA, Vol. 97, No.18, pp. 10101-06.

[32] Berrar D. P. *et al.*, (2003), "A Practical approach to microarray data analysis", KAP press.

[33] Valafar F., (2002), "Pattern recognition techniques in microarray data analysis: a survey", Special issue of Annals of New York Academy of Sciences, Techniques in Bioinformatics and Medical Informatics, Vol.980, pp. 41-64.

[34] Nguyen D.V. and Rocke D. M., (2002), "Tumor classification by partial least squares using microarray gene expression data", Bioinformatics, Vol 18, pp. 39-50.

[35] Statnikov A. *et al.*, (2005), "A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis". Bioinformatics; Vol. 21, pp. 631-643.

[36] Zhu J. and Hastie T., (2005), "classification of gene microarray by penalized logistic regression", Biostatistics, Vol. 5, No.3, pp 427-443.

[37] Huang D. and Zheng C., (2006), "Independent component analysis based penalized discriminant method for tumor classification using gene expression data", Bioinformatics, Vol. 22, NO. 16, pp.1855-1862.

[38] Yeung k. *et al.*, (2004), "Dominant spectral component analysis for transcriptional regulations using microarray time-series data", Bioinformatics, Vol. 20, pp. 742-749.

[39] Daniel K., (2002), "Information Visualization and Visual data mining", IEEE Tran. On Visualization and computer graphics, Vol.8, No. 1, pp. 1-8.

[40] Zhang L. *et al.*, (2006), "3D Visualisation of Gene Clusters", Computational image and Vision, Vol. 32, pp. 349-354.

[41] Chen1 X. *et al.*, (2006), "An effective structure learning method for constructing gene networks", Bioinformatics, Vol. 22, No. 11, pp. 1367–1374.

[42] Boris K. et al., (2002), "Untangling the wires: A strategy to trace functional interactions in signalling and gene networks", Cell Biology, Vol. 99, No. 20, 1Oct. pp. 12841–46.

[43] Tejaswi G. *et al.*, (2007), "Threshold logic gene regulatory networks", 5th IEEE International Workshop on Genomic Signal Processing and Statistics, Gensips07,10-12 June, Finland.

[44] Yeung Y. and Ruzzo L., (2000), "An empirical study on principal component analysis for clustering gene expression data", Technical. Report, UW-CSE-2000-11-03.

[45] Kotlyar M. *et al.*, (2002), "Spearman correlation identifies statistically significant gene expression clusters in spinal cord development and injury", Neurochemical Research, Vol.27, No.10, pp. 1133–40.

[46] Yeung K. Y. *et al.*., (2001), "Model based clustering and data transformations for gene expression data", Bioinformatics, Vol.17, No.10, pp. 977–987.

[47] Ramoni M. F. *et al.*, (2002), "Cluster analysis of gene expression dynamics", Proc. Natl. Acad. Sci., Vol. 99, No.14, pp. 9121–9126.

[48] Bar-Joseph Z. *et al.*, (2002), "A new approach to analyzing gene expression time series data", Proc. of the Sixth Annual International Conference on Computational Biology', Washington DC, USA, pp. 39–48.

[49] Daub Co. *et al.*. (2004),"Estimating mutual information using B-spline functions- an improved similarity measure for analysing gene expression data", BMC Bioinformatics, Vol.5, pp. 118.

[50] Priness I. *et al.*, (2007), "Evaluation of gene expression clustering via mutual information distance measure", BMC Bioinformatics, Vol. 8, pp. 111.

[51] Li H. *et al.*, (2004), "Minimum entropy clustering and applications to gene expression analysis", Proc. Of 3rd IEEE Computational systems, Bioinformatics conf., Stanford, CA, pp. 142-151.

[52] Segal N. H. *et al.*, (2003), "Classification and subtype prediction of adult soft tissue sarcoma by functional genomics", Amer. Jour. of Pathology, Vol. 163, No.2, pp. 691-700.

[53] Shyamsundar R. *et al.*, (2005), "A DNA microarray survey of gene expression in normal human tissues", Genome Biology, Vol. 6, No.3, pp. 404.

[54] Luo F. *et al.*, (2004), "A dynamically growing self organizing tree (DGSOT) for hierarchical clustering gene expression profiles", Bioinformatics, Vol. 20, No.16, pp. 2605–17.

[55] Zhong S. and Ghosh J. (2003), "A unified framework for model-based clustering", Journal of Machine Learning Research, Vol.4, pp. 1001–37.

[56] Alexandridis R. *et al.*, (2004), "Class discovery and classification of tumor samples using mixture modelling of gene expression data- a unified approach", Bioinformatics, Vol. 20, No. 16, pp 2545-52.

[57] Schleip A. *et al.*, (2003), "Using hidden Markov models to analyze gene expression time course data", Bioinformatics, Vol. 19, No. 1, pp. 255–263.

[58] Luan Y. and Li H., (2003), "Clustering of time-course gene expression data using a mixed-effects model with B-splines", Bioinformatics, Vol. 19, No. 4, pp. 474–482.

[59] Wang W. *et al.*, (1997), "STING: A statistical information grid approach to spatial data mining", Proc. of the 23rd Int. Conf. on Very Large Data Bases, Morgan Kaufmann, Athens, Greece, pp. 186-195.

[60] Sheikholeslami G. *et al.*, (1998), "WaveCluster: Amultiresolution clustering approach for very large spatial dataset", Proc. of the 24th Int. Conf. on Very Large Data Bases, Morgan Kaufmann, New York, USA, pp. 428–439.

[61] Ester M. *et al.*, (1996), "A density-based algorithm for discovering clusters in large spatial databases with noise", Proc. of the 2nd Int. Conf. on Knowledge Discovery and Data Mining, Portland, OR, USA, pp. 226–231.

[62] Ankerst M. *et al.*, (1999), "OPTICS: Ordering points to identify the clustering structure", Proc. of the ACM SIGMOD Int. Conf. on Management of Data, Philadelphia, PA, USA, pp. 150 151.

[63] Chung S. *et al.*, (2004), "Mining gene expression datasets using density-based clustering", Proc. of the Thirteenth ACM Conference on Information and Knowledge Management, Washington, DC, USA, pp. 150–151.

[64] Jiang D. *et al.*, (2003), "DHC: A density-based hierarchical clustering method for time series gene expression data", Proc. of the IEEE Computer Society Bioinformatics Conference, Stanford, CA, USA, pp. 137–147.

[65] Sharan R. *et al.*, (2003), "CLICK and EXPANDER: A system for clustering and visualizing gene expression data", Bioinformatics, VOL. 19, No. 14, pp.1787-99.

[66] Bellaachia A. *et al.*, (2002), "E-CAST: A Data Mining Algorithm For Gene Expression Data", Workshop on Data Mining in Bioinformatics (BIOKDD02), 23July, Canada, pp. 49-54.

[67] Lesot M. J. *et al.*, (2003), "Evaluation of topographic clustering and its kernelization", Proc. of the European Conference on Machine Learning, pp.265–276.

[68] Nikkilä J., *et al.*, (2002), "Analysis and Visualization of Gene Expression Data using Self-Organizing Maps", Neural Networks Special Issue on New Developments on Self-Organizing Maps, Vol. 15, No. 8, pp. 953-966.

[69] Halkidi M. *et al.*, (2001), "On clustering validation techniques", Journal of Intelligent Information Systems, Vol. 17, No.2, pp107–145.

[70] Bolshakova N. and Azuaje F., (2003), "Cluster validation techniques for genome expression data", Signal Processing, Vol. 83, No.4, pp 825–833.

[71] Datta S. and Datta D., (2003), "Comparisons and validation of statistical clustering techniques for microarray gene expression data", Bioinformatics,Vol.19,No.4, pp.459–466.

[72] Quatieri T., (2002), "Discrete time speech signal processing: principles and practice", Prentice Hall.

[73] Dhillon I. *et al.*, (2004), "Kernel K-mean, Spectral Clustering and Normalized cuts", Proc. Of KDD'04, 22-25 Aug, Washington, USA.

[74] Kannan R. *et al.*, (2004), "On Clusterings: Good, Bad and Spectral", Journal of the ACM, Vol. 51, No. 3, May, pp. 497–515.

[75] Higham D. *et al.*, (2007), "spectral clustering and its use in Bioinformatics", J. of computational and applied mathematics, Vol. 204, pp. 25-37

[76] Rabiner L. and Schafer R., (2007), "Introduction to Digital Speech Processing", Foundations and Trends® in Signal Processing, Vol. 1, No. 1–2, pp. 1-194.

[77] Nocerino N. *et al.*, (1985), "Comparative study of several distortion measures for speech recognition". IEEE Proc Acoustics, Speech and Signal Processing; Vol. 10, pp. 25-28.

[78] Gersho A. and Gray R., (1992), "Vector Quantization and Signal Compression", Kluwer Academic Press.

[79] Goldbeter A., (2007), "Biological rhythms as temporal dissipative structures", Advances in Chemical Physics, Vol. 135, pp. 253-295.

[80] Hayes M., (1996), "Statistical digital signal processing and modelling", John Wiley.&Sons

[81] Gene H. *et al.*, (1989), "Matrix computations", The johns Hopkins university press.

[82] Paliwal K. And Kleijn B., (1995), "Quantization of LPC parameters", Speech Coding and Synthesis, Elsevier Science B.V., Amsterdam, Nov. 1995, pp. 433-466

[83] So S. and Paliwal K. K., (2005), "A comparison of LSF and ISP representations for wideband LPC parameter coding using the switched split vector quantiser", Proc. Int. Symp. on Signal Processing and Its Applications (ISSPA-2005), Sydney, Australia

[84] Itakura F., (1975), "Line spectrum representation of linear predictive coefficients of speech signals", J. Acoust. Soc. Am., Vol.57, No.535, pp. 535.

[85] Li T. *et al.*, (2002), "A Survey on Wavelet Applications in Data Mining", SIGKDD Explorations, Vol. 4, No. 2, p.p. 49-68.

[86] Otazu X. and Pujol O., (2006), "Wavelet based approach to cluster analysis. Application on low dimensional datasets", Pattern Recognition Letters, Vol.27, pp. 1590–1605

[87] Stark H. G.,(2005),"Wavelet and signal processing", Springer

[88] Prabakaran S. *et al.*, (2006), " Characterization of microarray data using wavelet power spectrum", Int. J. of Knowledge-based and Intelligent Engineering system, Vol. 10, pp. 493-501

[89] Liu Y., (2009), "Wavelet feature extraction for high-dimensional microarray data", Neurocomputing, Vol.72, No. 4, pp. 985– 990.

[90] Liu Y., (2008), "Detect Key Gene Information in Classification of Microarray Data", EURASIP Journal on Advances in Signal Processing, Vol. 2008.

[91] Moesa H. A. *et al.*, (2005), "Efficient Determination of Cluster Boundaries for Analysis of Gene Expression Profile Data Using Hierarchical Clustering and Wavelet Transform", Genome Informatics, Vol. 16, No.1, pp. 132-141.

[92] Barbará D. and Chen P., (2000), "Using the fractal dimension to cluster datasets", Proc. Int'l Conf. on Knowledge Discovery and Data Mining, Boston, USA, pp. 260-264.

[93] Jelinek F. *et al.*, (1998), "Is there meaning in fractal analyses", Complex systems, Conference 98 (UNSW Sydney, 1998), pp.144-149.

[94] Hu X. *et al.*, (2005), "Classification of surface emg signal with fractal dimension," Journal of Zhejiang University Science, Vol. 6, No.8, pp. 844–848.

[95] Seymour G., (2004)," Fractal properties of the human genome", Journal of Theoretical Biology, Vol. 230, pp. 251–260.

[96] Zhi-Yuan S. *et al.*, (2007),"Local scaling and multifractal spectrum analyses of DNA sequences – GenBank data analysis", Chaos, Solitons and Fractals, Vol.40, No. 4, pp. 1750-65.

[97] Carlin M., (2000), "Measuring the complexity of non-fractal shapes by a fractal method", Pattern Recognition Letters, Vol. 21, No. 11, pp. 1013–1017.

[98] Camastra F., (2003), "Data dimensionality estimation methods: a survey", Pattern Recognition, Vol. 36, pp. 2945-54.

[99] Peter G. and Borisov A., (2002), "Using Grid-Clustering Methods in Data Classification", Int Conf. on Parallel Computing in Electrical Engineering (PARELEC'02), pp. 425.

[100]Katz M. J., (1988), "Fractals and the Analysis of Waveforms", Comput. Biol. Med., Vol. 18, No. 3, pp. 145-156.

[101]Mandelbrot B. and Novak M., (2004),"Thinking in Patterns: Fractals and Related Phenomena in Nature", World Scientific Publishing Co.

[102]Hadjileontiadis L. J. and Rekanos I. T., "Detection of Explosive Lung and Bowel Sounds by Means of Fractal Dimension", IEEE Signal Processing Letters,Vol.10,No.10, pp.311.

[103]Peitgen H. O. *et al.*, (1992), "Chaos and Fractals: New Frontiers of Science", New York, NY: Springer-Verlag.

[104]Traina C. J. *et al.*, (2000), "Fast feature selection using the fractal dimension". In XV Brazilian Symposium on Databases (SBBD).

[105]Bhavani S. *et al.*, (2008), "Feature selection using correlation fractal dimension: Issues and applications in binary classification problems", Applied soft computing, Vol. 8, No.1, pp 555-563.

[106]Alon U. *et al.*, (1999), "Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays", In Proc. Natnl. Acad. Sci., Vol. 96, pp. 6745-50.

[107]Iizuka N. *et al.*, (2003), "Oligonucleotide microarray for prediction of early intrahepatic recurrence of hepatocellular carcinoma after curative resection", The Lancet, Vol. 361, No. 9361, pp. 923-929.

[108]Singh D. *et al.*, (2002), "Gene expression correlates of clinical prostate cancer behavior". Cancer Cell, Vol.1, No.2, pp. 203-209.

[109]Nutt C. *et al.*, (2003), "Gene expression-based classification of malignant gliomas correlates better with survival than histological classification", Cancer Research, Vol. 63, April, pp. 1602-1607.

[110]Tibshirani R. *et al.*, (2002), "Diagnosis of multiple cancer types by shrunken centroids of gene expression", PNAS, Vol. 99, No.10, pp. 6567–6572.

[111]Mukkamala S. *et al.*, (2005), "Computational Intelligent Techniques for tumor classification using microarray gene expression data", Int. Jour. of Lateral Computing, Vol. 2, No. 1, pp. 38-45.

[112] Nguyen D. and Rocke D., (2002), "Tumor classification by partial least squares using microarray gene expression data", Bioinformatics, Vol. 18, No. 1, pp. 39-50.

[113]Liu Z. *et al.*, (2005), "Gene Expression Data Classification with Kernel Principal Component Analysis", J. Biomed. Biotechnol., Vol. 2, pp. 155–159.

[114]Jong K. *et al.*, (2003), "Finding clusters using support vector classifiers", ESANN-1th European Symposium on Artificial Neural Networks Bruges, Belgium, April 23-25,.

[115]Chandra B. *et al.*, (2006), "A new approach: Interrelated two way clustering of gene expression data", Jour of statistical methodology, Vol. 3, pp. 93-102.

[116]Furey T. *et al.*, (2000), "Support vector machine classification and validation of cancer tissue samples using microarray expression data". Bioinformatics, Vol. 16, No. 10, pp. 906–914.

[117]Ding C. and Peng H., (2003), "Minimum redundancy feature selection from microarray gene expression data", IEEE Computer Society Bioinformatics conf.- CSB, Aug 2003, Stanford, CA, USA., pp. 523-529

[118]Huerta E. *et al.*, (2006), "A Hybrid GA/SVM Approach for Gene Selection and Classification of Microarray Data", Evo Workshops, LNCS, Vol. 3907,.pp. 34-44

[119]Huang D. and Zheng C., (2006), "Independent component analysis-based penalized discriminant method from tumor classification using gene expression data", Bioinformatics, Vol.22, No.15, pp. 1855-1862.

List of publications

A. Sungoor, R. S. H. Istepanian, J.-C. Nebel, (2005), *"Channel Coding Theory for Microarray Data Analysis"*, Young Bioinformaticians Forum 2005, October, London,

R. S. H. Istepanian, A. Sungoor, J.-C. Nebel, (2007), *"Linear predictive coding for enhanced microarray data clustering"*, GENSIPS'07, Tuusula, Finland, June, (*2007*)

R. S. H. Istepanian, A. Sungoor, J.-C. Nebel, (2007), *"Linear Predictive Coding and Wavelet Decomposition for Robust Microarray Data Clustering"*, IEEE-EMBC07, August, Lyon, France.

R. S. H. Istepanian, A. Sungoor, J.-C. Nebel, (2008), *"Fractal Dimension and Wavelet Decomposition for Robust Microarray Data Clustering"*, IEEE-EMBC08, August, Vancouver, British Columbia, Canada.

R. S. H. Istepanian, Karima Zitouni, Diane Harry, A. Sungoor, Bee Tang and Kenneth A Earle, (2009), "*Evaluation of a mobile phone telemonitoring system for glycaemic control in patients with diabetes*", J Telemed Telecare, Vol. 15, pp. 125-128

R. S. H. Istepanian, A. Sungoor, (2009), "*Technical and Compliance Issues of Mobile Diabetes Self- monitoring using Glucose and Blood Pressure Measurements*", 31st IEEE EMBS Conference, Sept. 2-6, MN, USA

R. S. H. Istepanian, Joseph Cafazzo, Emily Seto, Alexander Logan and A. Sungoor, (2009), "*UK and Canadian Perspectives of the Effectiveness of Mobile Diabetes Management Systems*", 31st IEEE EMBS Conference, Sept. 2-6, MN, USA

Appendix A

Application of GSP methods for clustering microarray

In this appendix we present and discuss the application of GSP methods for clustering further microarray datasets.

A.1- Preprocessing

The same procedure for normalisation described in section 6.2 is applied to normalise the other datasets and find the top genes expression. Figure (a.1) shows the result when applied to Colon dataset, while Table (a-1) list the top 10th highly expression genes.

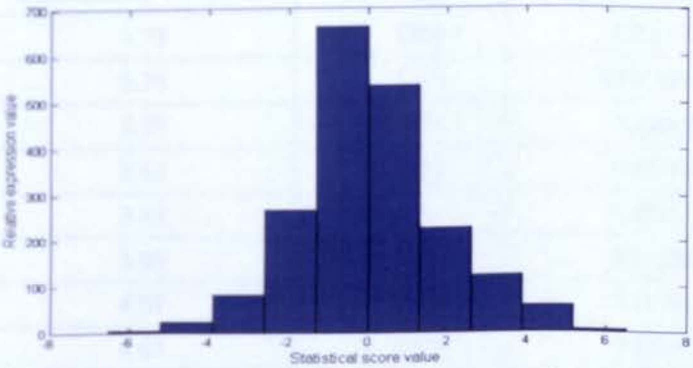


Figure a.1 Distribution of gene expression for Colon dataset

Table a-1 Highly expression genes in Colon dataset

Statistical value	Gene index	Gene Probe
-6.51	493	R87126
-5.64	1423	J02854
-5.63	249	M63391
-5.32	377	Z50753
-5.00	49	T61661
-4.97	66	T71025
-4.95	245	M76378
-4.90	267	M76378
-4.82	14	H20709
-4.72	765	M76378

Consequently, normalisation process of Hepatocellular dataset is shown in Figure(a.2) and the results concerning list of the top 10th highly expression genes is shown in Table (a-2).

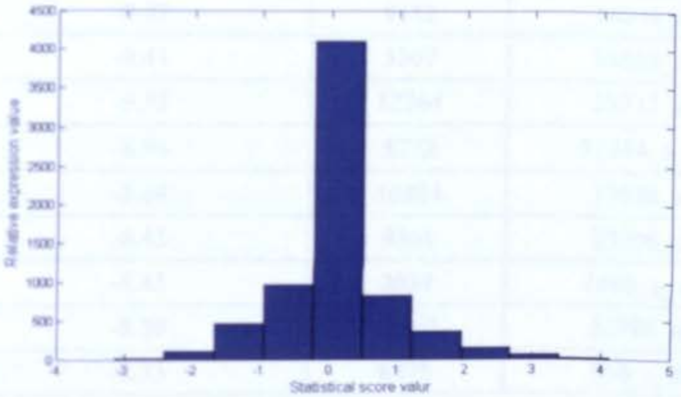


Figure a.2 Distribution of gene expression for Hepatocellular dataset

Table a-2 Highly expression genes in Hepatocellular dataset

Statistical value	Gene index	Gene Probe
3.73	1806	U22431
3.75	5280	AF008445
3.77	2686	X16663
3.81	5956	U19495
3.83	3213	D28915
3.95	5429	X03100
4.01	1831	X51345
4.05	5718	Y10032
4.11	6150	AB000409
4.26	336	U20734
4.87	6585	L36033

Normalisation process of Prostate dataset is shown in Figure (a.3) and the results concerning list of the top 10th highly expression genes is shown in Table (a-3)

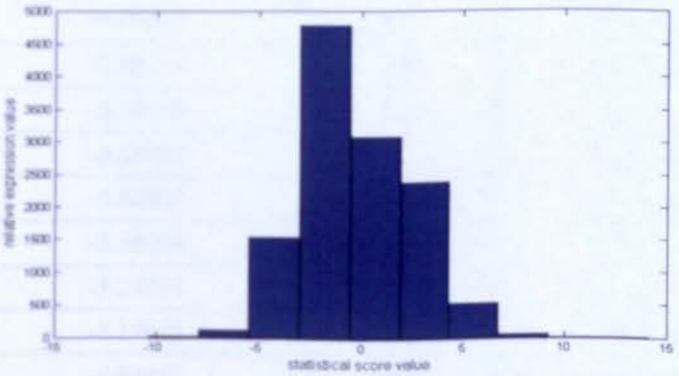


Figure a.3 Distribution of gene expression for Prostate dataset

Table a-3 Highly expression genes in Prostate dataset

Statistical value	Gene index	Gene Probe
-10.37	3017	39939_at
-9.47	9112	38044_at
-9.41	5307	38028_at
-9.35	12264	39315_at
-8.96	5272	31444_s_at
-8.64	10204	32076_at
-8.45	9361	32206_at
-8.43	2034	1598_g_at
-8.39	11263	32780_at
-8.33	6475	556_s_at

Normalisation process of Gliomas dataset is shown in Figure (a.4) and the results concerning list of the top 10th highly expression genes is shown in Table (a-4)

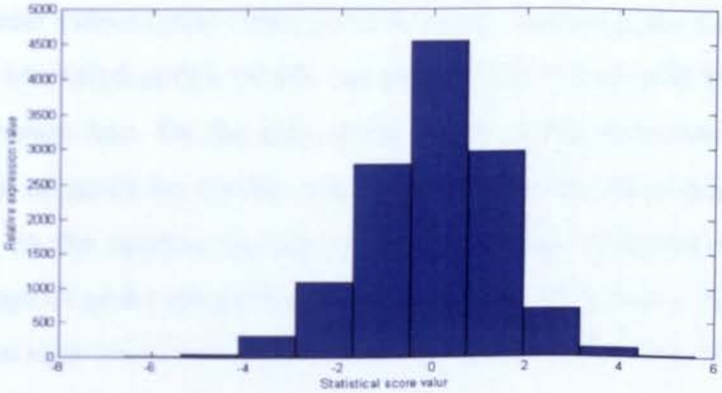


Figure a.4 Distribution of gene expression for Gliomas dataset

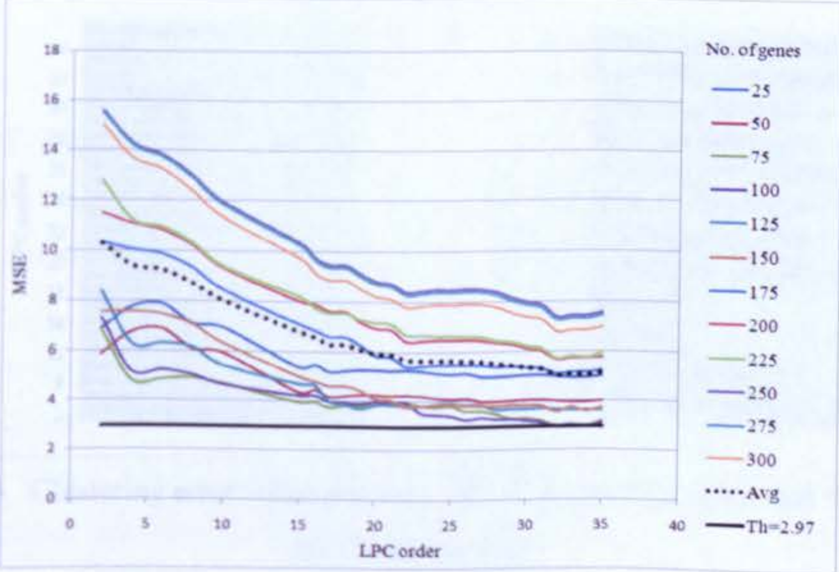
Table a-4 Highly expression genes in Gliomas dataset

Statistical value	Gene index	Gene Probe
-6.60786	10605	630_at
-6.05365	6697	31600_s_at
-5.93784	4920	38421_at
-5.74179	1841	267_at
-5.66961	1876	40581_at
-5.66905	8740	35163_at
-5.38224	12471	446_at
-5.27635	2604	631_g_at
-5.17945	5811	32183_at
-4.93865	3344	39691_at

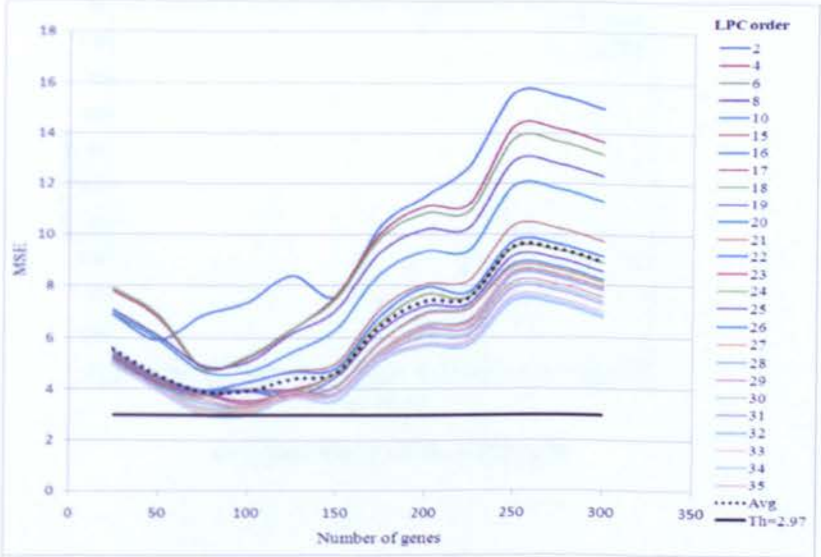
A.2- Application of miLPC

1- Colon dataset

Figure (a.5) shows a spectrum of MSE for predictive coefficients based on multiple runs according to different numbers of involved genes with respect to multiple values of LPC order. By passing the estimated coefficients to vector quantisation, the clustering is accomplished to group those coefficients. Figure (a.5a) demonstrates the MSE to different LPC order for specific number of genes. Note that the MSE value is influenced by the LPC order, it appears to be inversely related to the model order due to the prediction order *p* has the greatest influence on the complexity of coefficients calculation that comes from increasing the linear combination of previous samples taken from combinations of the enhanced digital filter stages. Considering this higher model order appears to be advantageous, because the difference between the original expression signal and the final results will be smaller. Accordingly there is possible range of LPC order values could result given accurate clustering, the figure shows that a minimum MSE identified as (*Th =2.97*) can separate the classes with best cluster which is below the average line. On the other hand Figure (a.5b) demonstrates the MSE to different number of genes for specific number of LPC order. Note that the MSE value also influenced by the specific number of genes involve. However in many cases a considerable range of genes are differentially expressed, this sense is due to a relatively small variation in expression have considerable noise that affected on the coefficients.



a- MSE of different LPC order



b- MSE of different number of gene

Figure a.5 miLPC analyses with respect to different number of genes and LPC orders for Colon dataset

Figure (a.6) represents the graphical representation of clustering error of different value of gene number with respect to different LPC orders. It shows the minimum number of samples that are not clustered accurately, when $g=100$ genes processing with LPC order $p=\{32, 34\}$. Figure (a.7) shows Voronoi diagram to the result of two classes clustering of the microarray samples.

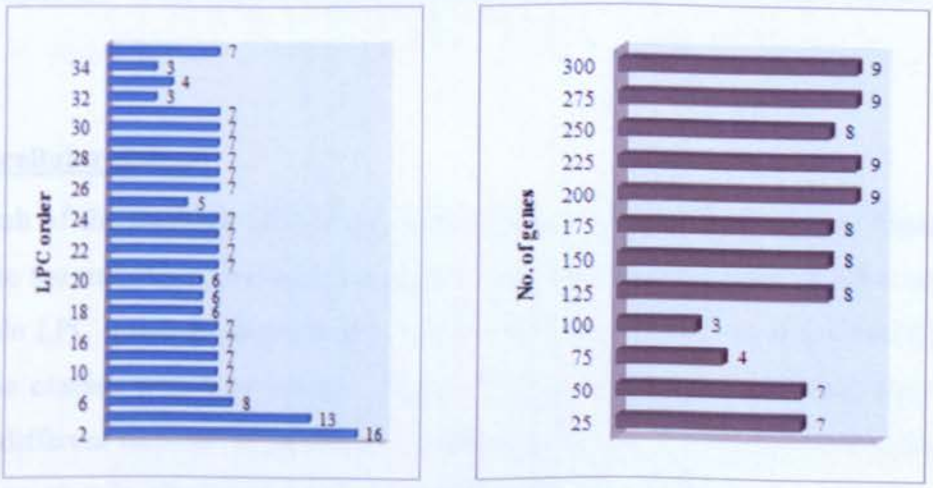
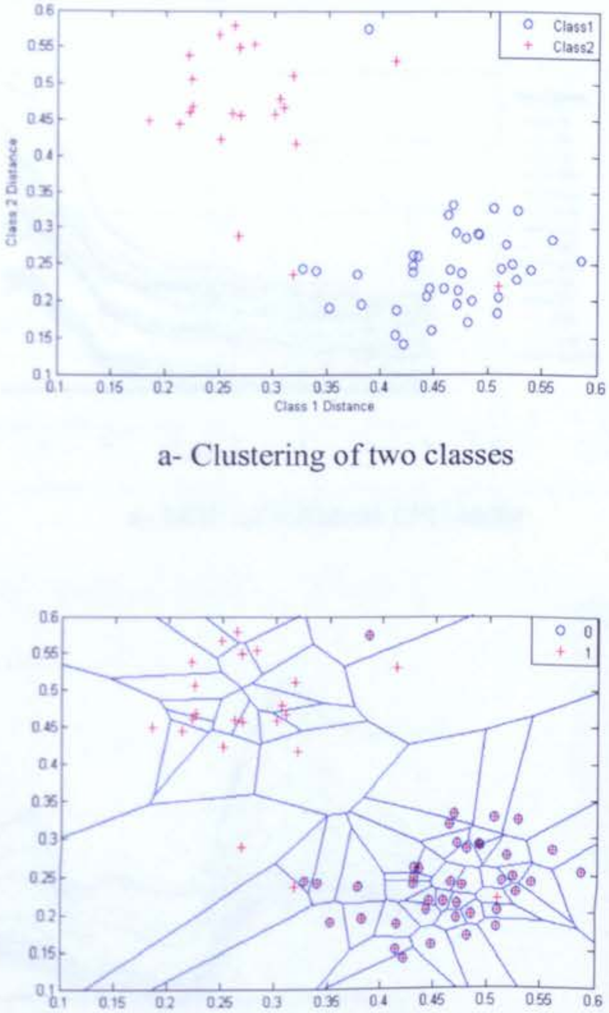


Figure a.6 Clustering error with respect to different number of gene and LPC orders for Colon dataset



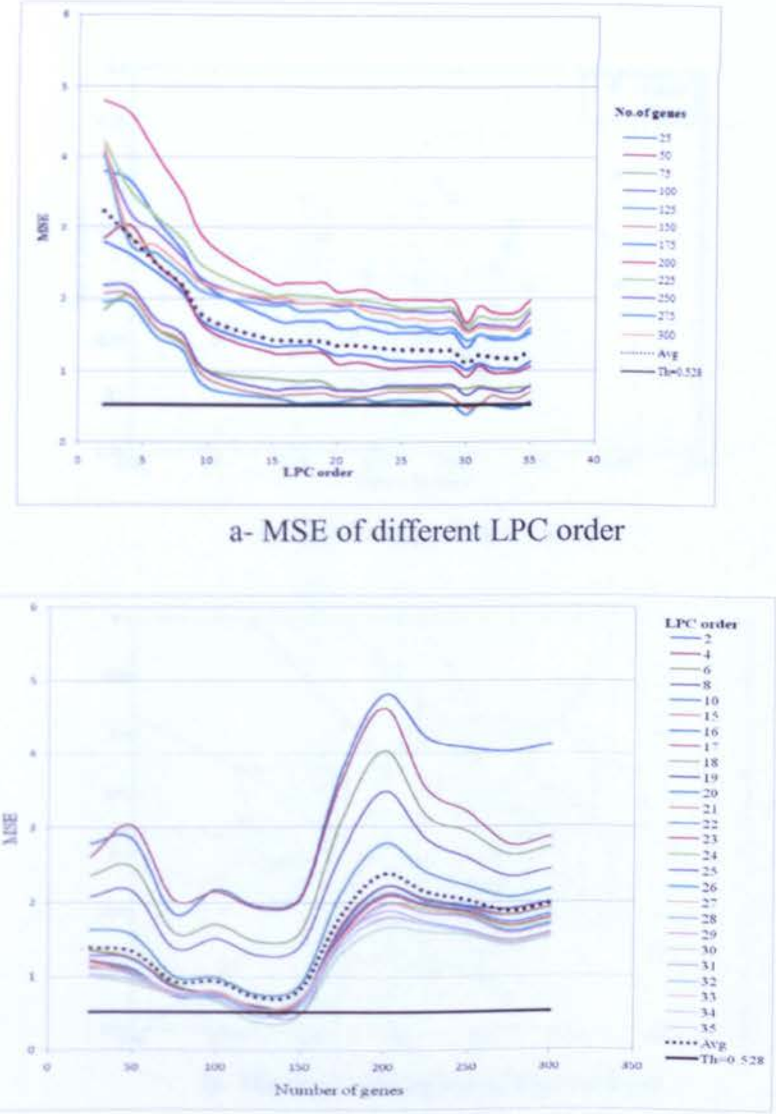
a- Clustering of two classes

b- Voronoi clustering of two classes

Figure a.7 Two class clustering of microarray samples for Colon dataset

2- Hepatocellular dataset

The result of the analysis concerning Hepatocellular dataset is shown in Figure(a.8) demonstrate the miLPC performance analyses with respect to different number of gene and multiple LPC order. It shows that the minimum MSE identified as (*Th=0.528*) can separate the classes with best cluster. Figure (a.9) demonstrates clustering error with respect to different number of genes and multiple LPC order. Figure (a.10) shows the result of two classes clustering of the microarray samples with accuracy 76%. It is clear to highlight that there are 14 samples unclassified out of 60 samples in the microarray known as {6,18,19,22,23,28,30,35,36,38,43,52,54,60}.



a- MSE of different LPC order

b- MSE of different number of gene

Figure a.8 miLPC analyses with respect to different number of gene and LPC orders for Hepatocellular dataset

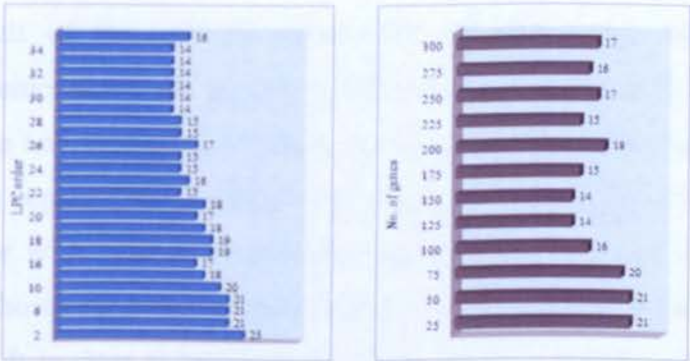
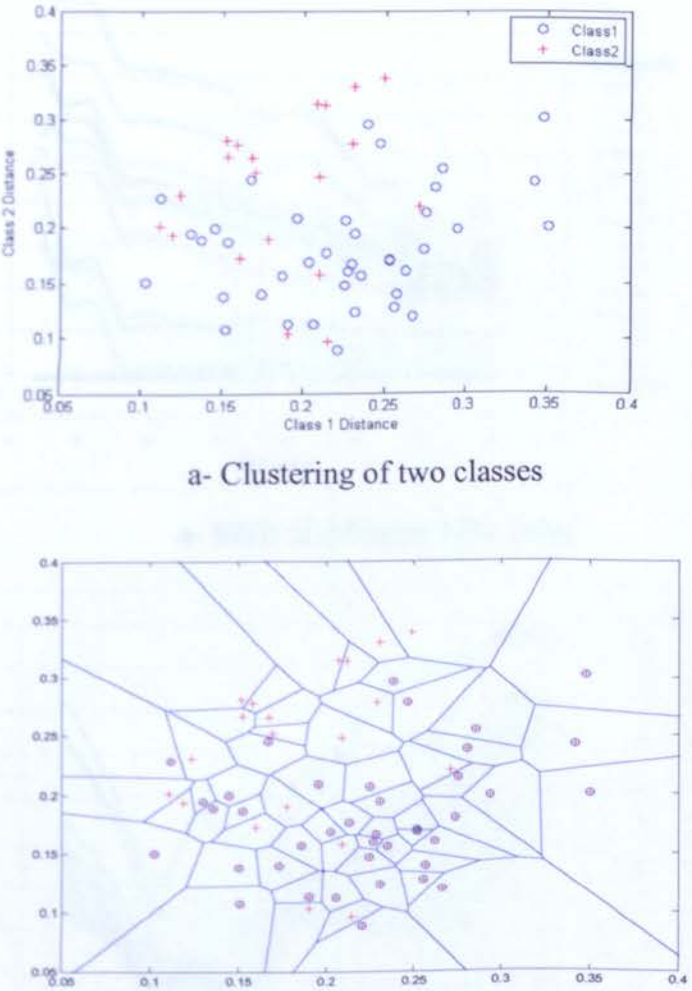


Figure a.9 Clustering error with respect to different number of gene and LPC orders for Hepatocellular dataset



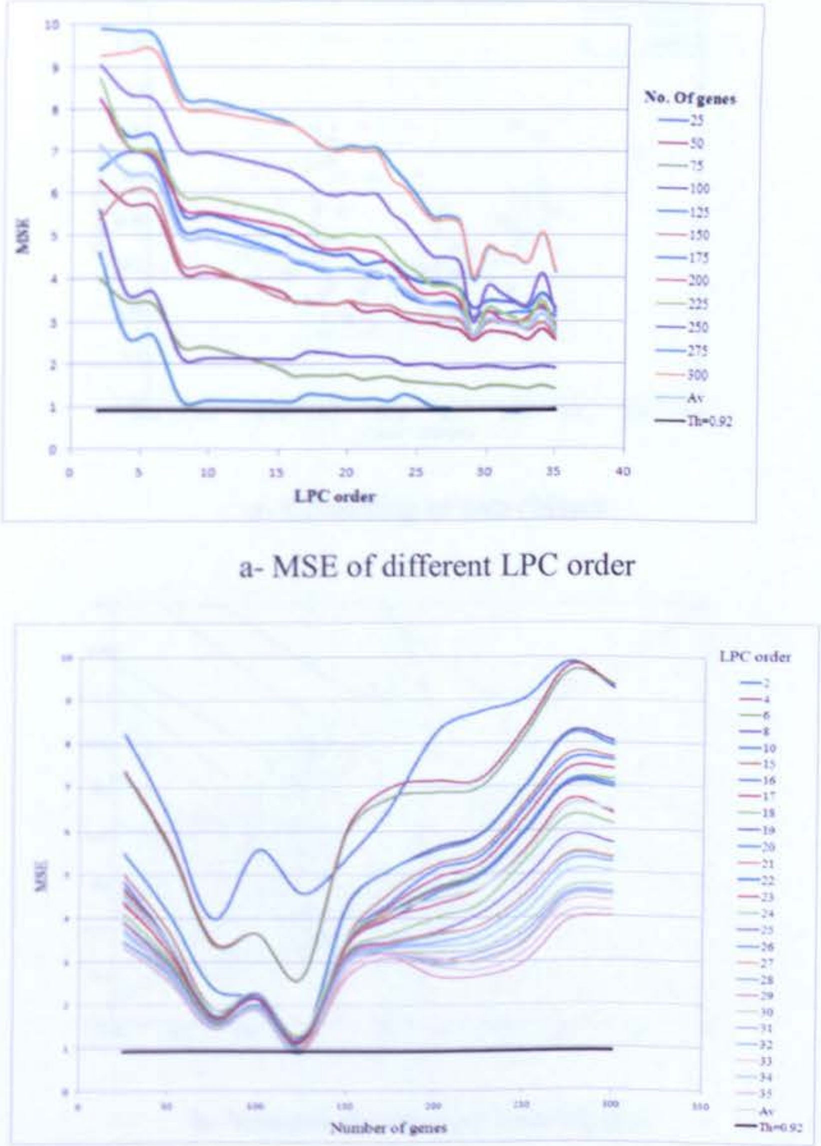
a- Clustering of two classes

b- Voronoi clustering of two classes

Figure a.10 Two class clustering of microarray samples for Hepatocellular dataset

3- **Prostate dataset**

The result of the analysis concerning Prostate cancer dataset is shown in Figure(a.11) demonstrate the miLPC performance analyses with respect to different number of gene and multiple LPC order. It shows that the minimum MSE identified as (*Th=0.92*) can separate the classes with accurate cluster. Figure (a.12) demonstrates clustering error with respect to different number of genes and multiple LPC order. Figure (a.13) shows the result of two classes clustering of the microarray samples with accuracy 90%. It is clear to highlight that there are 14 samples unclassified out of 136 samples in the microarray known as {32,47,53,54,59,61,62,64,68,80,81,84,92,95}.



a- MSE of different LPC order

b- MSE of different number of gene

Figure a.11 miLPC analyses with respect to different number of gene and LPC orders for Prostate dataset

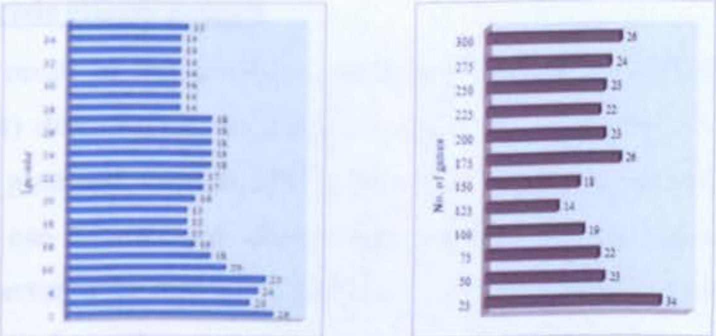
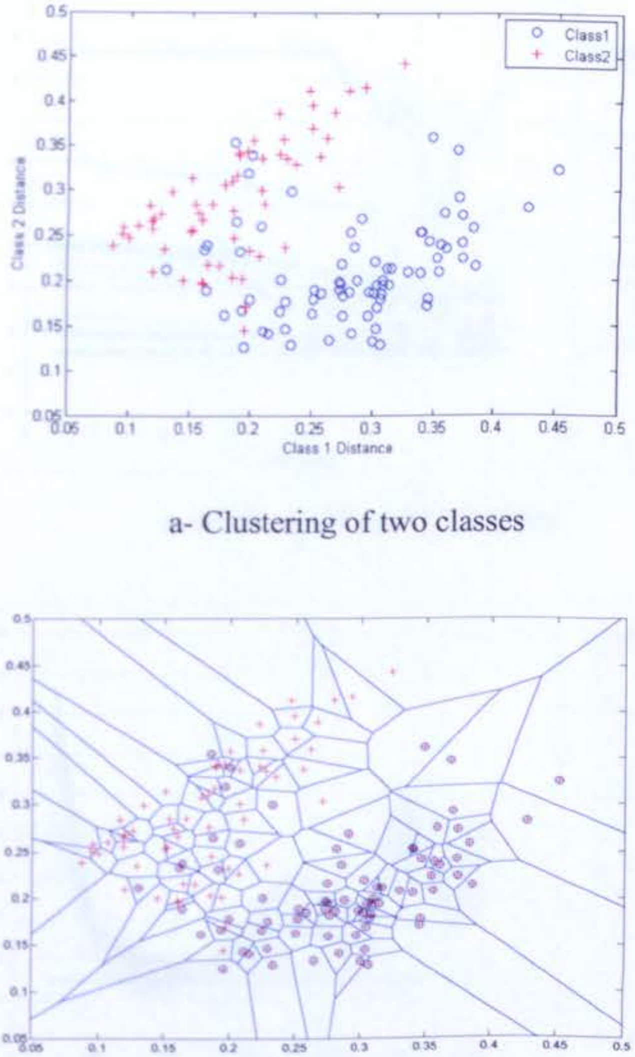


Figure a.12 Clustering error with respect to different number of gene and LPC orders for Prostate dataset



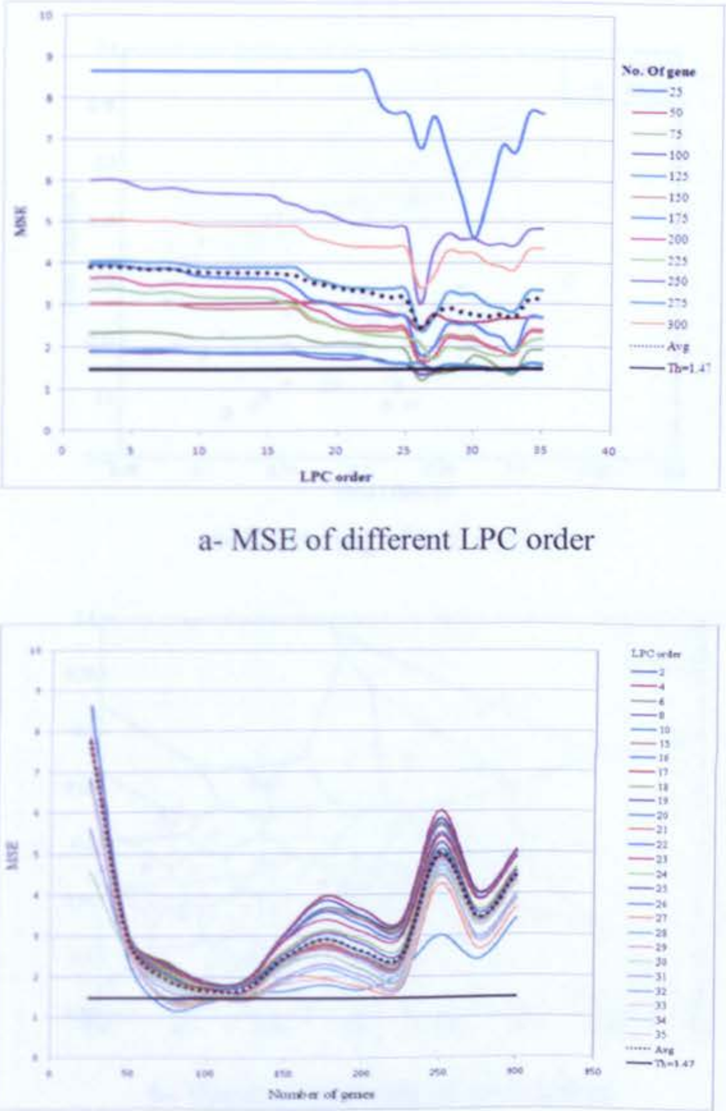
a- Clustering of two classes

b- Voronoi clustering of two classes

Figure a.13 Two class clustering of microarray samples for Prostate dataset

4- High-grade glioma dataset

The result of the analysis concerning high-grade glioma dataset is shown in Figure(a.14) demonstrate the miLPC performance analyses with respect to different number of gene and multiple LPC order. It shows that the minimum MSE identified as (*Th=1.47*) can separate the classes with accurate cluster. Figure (a.15) demonstrates clustering error with respect to different number of genes and multiple LPC order. Figure (a.16) shows the result of two classes clustering of the microarray samples with accuracy 90%. It is clear to highlight that there are 5 samples unclassified out of 50 samples in the microarray known as {12,18,22,28,35}.



a- MSE of different LPC order

b- MSE of different number of gene

Figure a.14 miLPC analyses with respect to different number of gene and LPC orders for high-grade glioma dataset

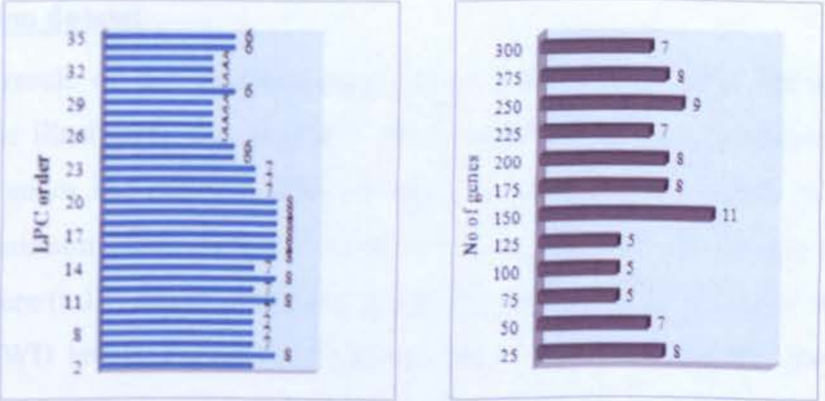
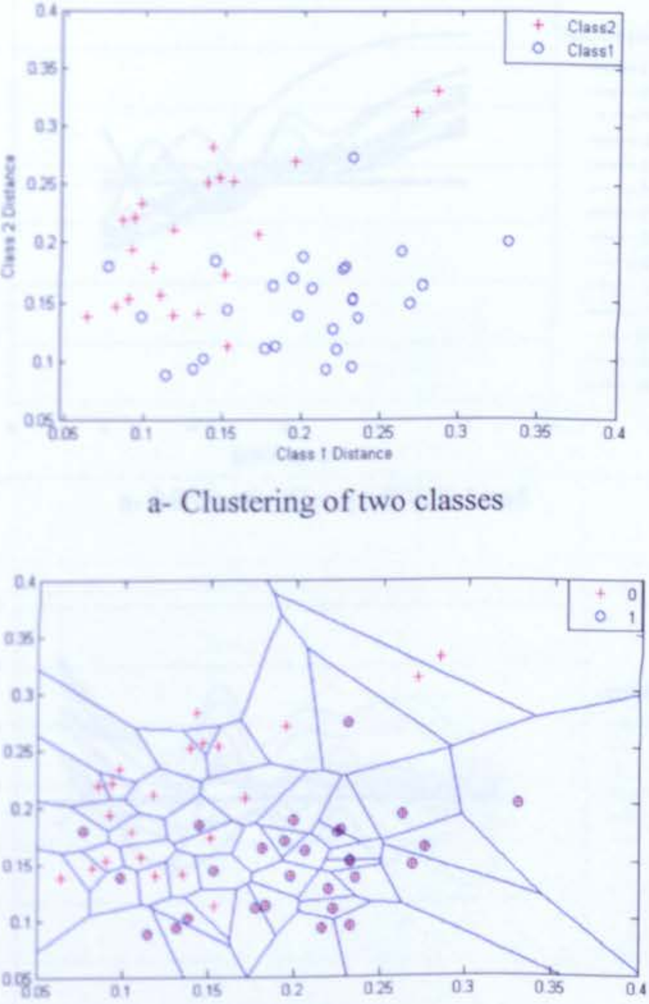


Figure a.15 Clustering error with respect to different number of gene and LPC orders for high-grade glioma dataset



a- Clustering of two classes

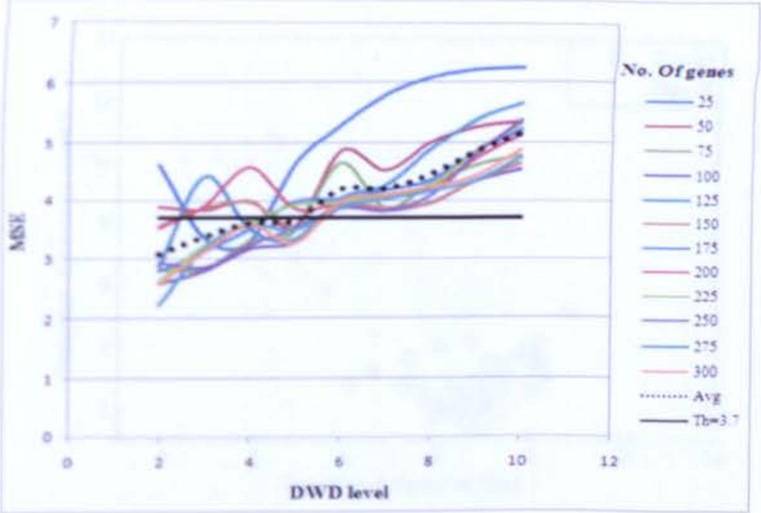
b- Voronoi clustering of two classes

Figure a.16 Two class clustering of microarray samples for high-grade glioma dataset

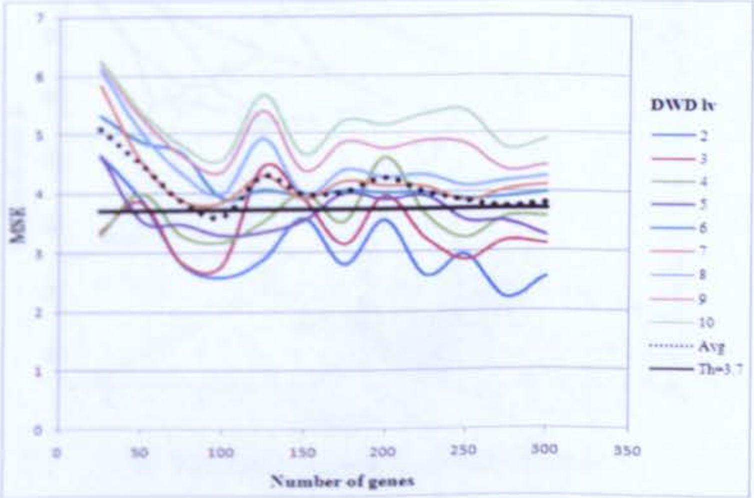
A.3- Application of miDWD

1- Colon dataset

The result of the analysis concerning Colon dataset after applying miDWD algorithm is illustrated. The miDWD performance analyses with respect to different number of genes and multiple DWD levels is demonstrated in Figure (a.17). It shows that the minimum MSE identified as (*Th=3.7*) can separate the classes with accurate cluster. Figure (a.18) shows clustering error of different number of genes with respect to different DWD levels. Figure (a.19) shows the result of two classes clustering of the microarray samples accurately with accuracy 97%. It is clear to highlight that there are two samples unclassified out of 62 samples in the microarray known as {16, 51} which are particularly difficult to classify since they have weak expression levels.



a- MSE of different DWD level



b- MSE of different number of genes

Figure a.17 miDWD analyses with respect to different number of genes and DWD levels for Colon dataset

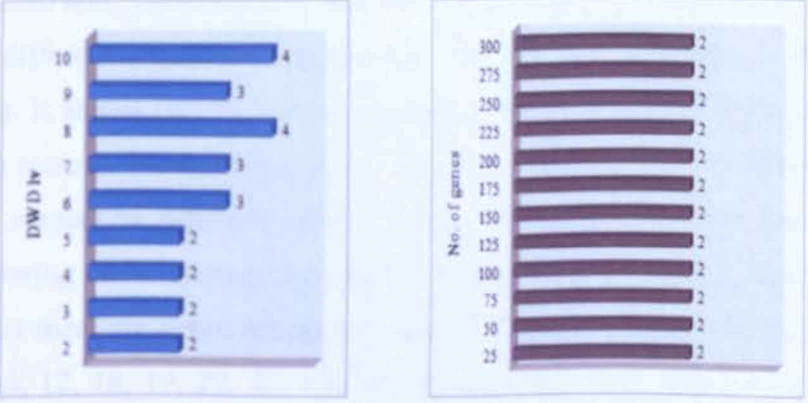
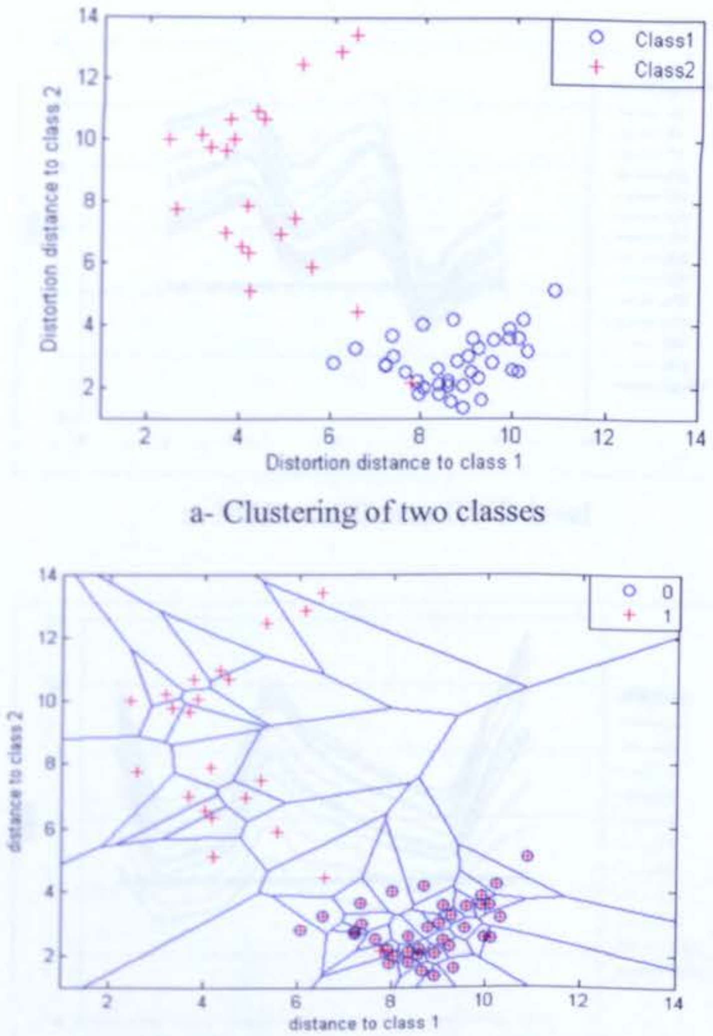


Figure a.18 Clustering error with respect to different number of gene and DWD levels for Colon dataset



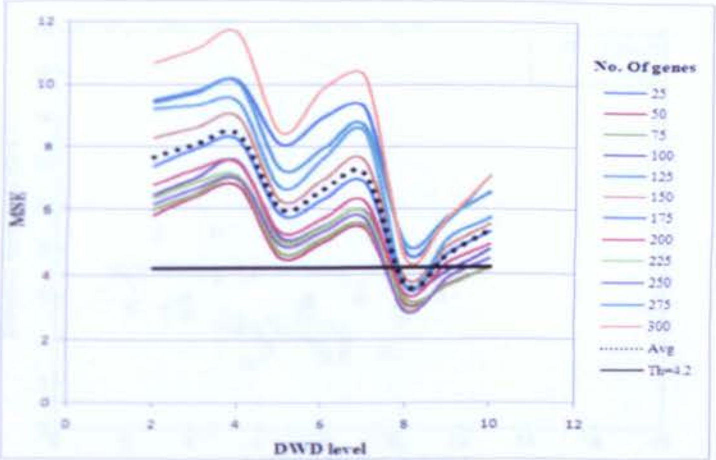
a- Clustering of two classes

b- Voronoi clustering of two classes

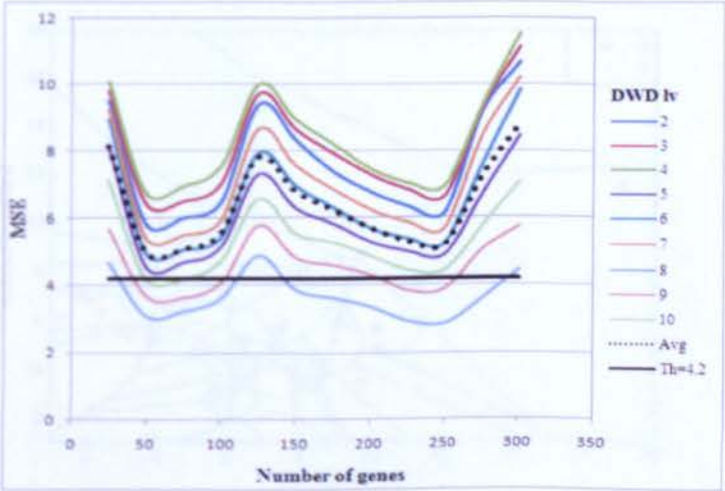
Figure a.19 Two class clustering of microarray samples for Colon dataset

2- Hepatocellular dataset

The miDWD performance analyses concerning Hepatocellular dataset with respect to different number of genes and multiple DWD levels is demonstrate in Figure(a.20). It shows that the minimum MSE identified as (*Th=4.2*) can separate the classes with accurate cluster. Figure (a.21) shows clustering error of different number of genes with respect to different DWD levels. Figure (a.22) shows the result of two classes clustering of the microarray samples accurately with accuracy 90%. It is clear to highlight that there are seven samples unclassified out of 60 samples in the microarray known as {6, 12, 18, 19, 22, 23, 60} which are particularly difficult to classify since they have weak expression levels.



a- MSE of different DWD level



b- MSE of different number of genes

Figure a.20 miDWD analyses with respect to different number of genes and DWD levels for Hepatocellular dataset

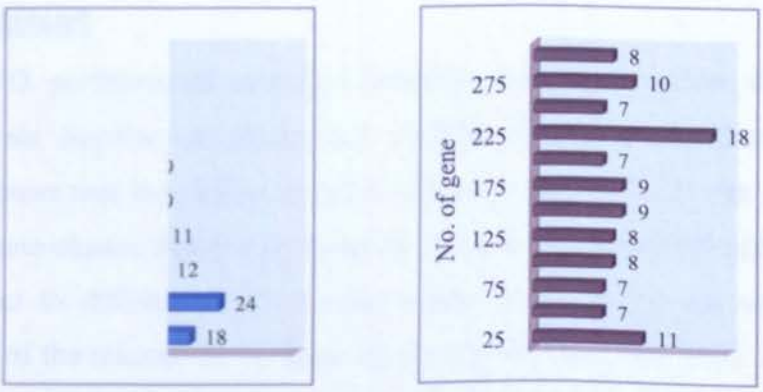
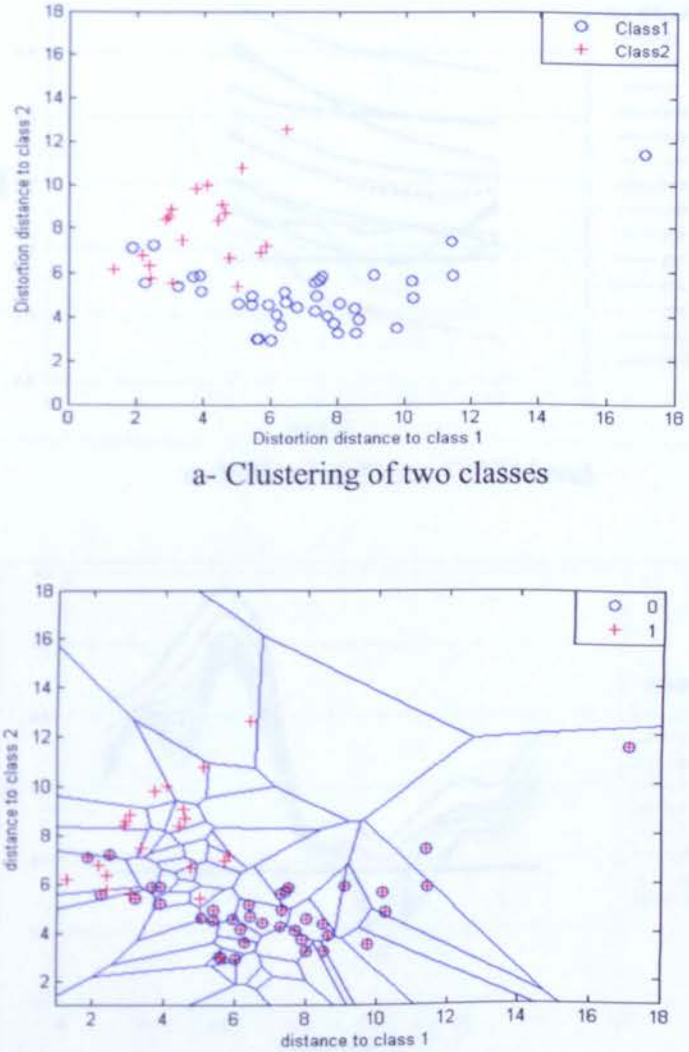


Figure a.21 Clustering error with respect to different number of genes and DWD levels for Hepatocellular dataset



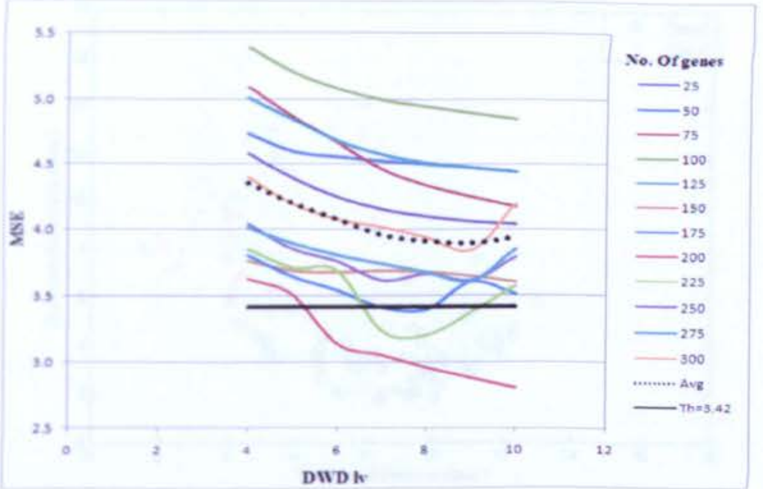
a- Clustering of two classes

b- Voronoi clustering of two classes

Figure a.22 Two class clustering of microarray samples for Hepatocellular dataset

3- Prostate dataset

The miDWD performance analyses concerning Prostate cancer dataset with respect to different number of genes and multiple DWD levels demonstrate in Figure(a.23). It shows that the minimum MSE identified as (*Th=3.42*) can separate the classes with accurate cluster. Figure (a.24) shows clustering error of different number of genes with respect to different DWD levels. Figure (a.25) shows the result of two classes clustering of the microarray samples accurately with accuracy 94%. It is clear to highlight that there are eight samples unclassified out of 136 samples in the microarray known as {32, 33, 47, 57, 68, 81, 84, 92} which are particularly difficult to classify since they have weak expression levels.



a- MSE of different DWD level

b- MSE of different number of genes

Figure a.23 miDWD analyses with respect to different number of genes and DWD levels for Prostate dataset

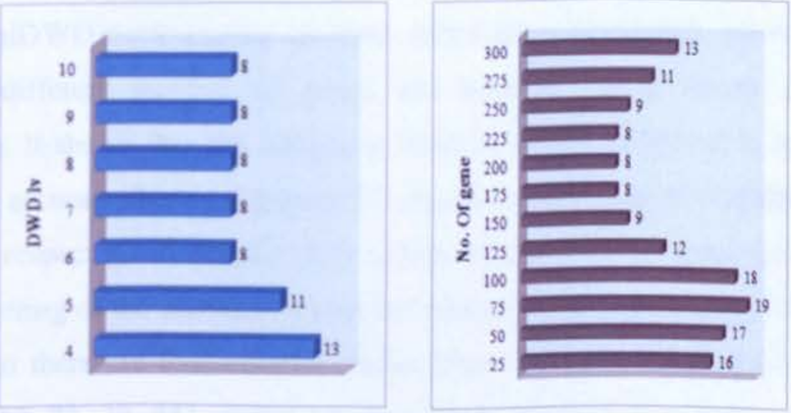
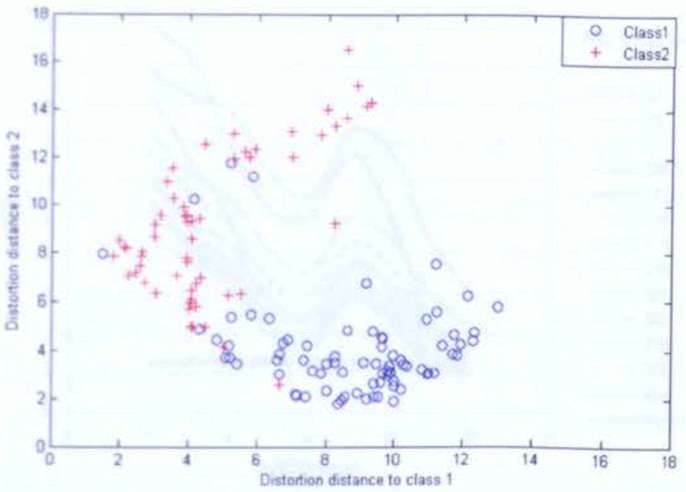
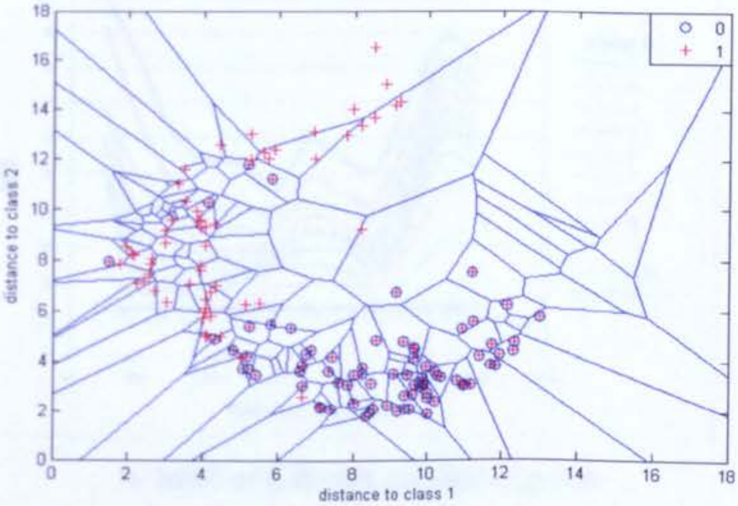


Figure a.24 Clustering error with respect to different number of genes and DWD levels for Prostate dataset



a- Clustering of two classes

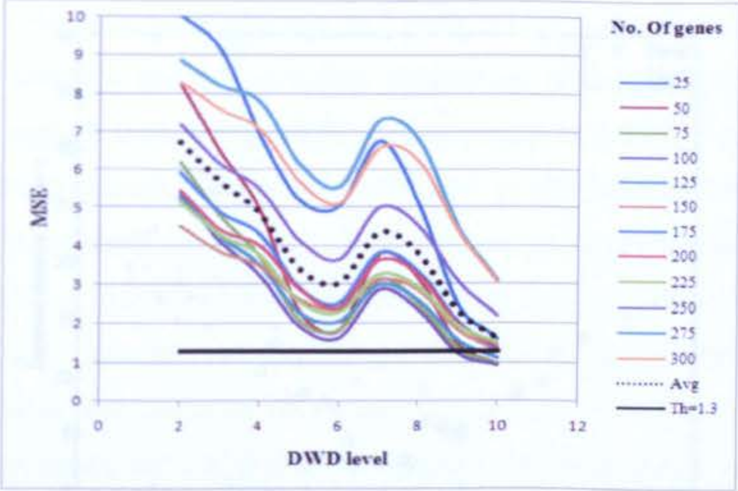


b- Voronoi clustering of two classes

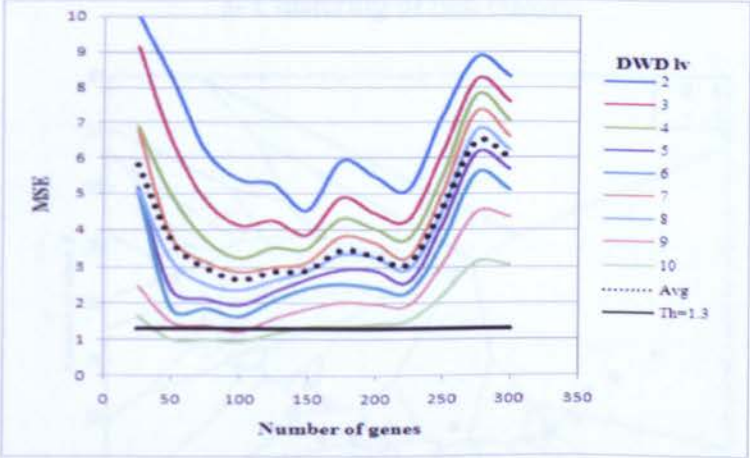
Figure a.25 Two class clustering of microarray samples for Prostate dataset

4- High-grade glioma dataset

The miDWD performance analyses concerning High-grade glioma dataset with respect to different number of genes and multiple DWD levels demonstrate in Figure(a.26). It shows that the minimum MSE identified as (*Th=1.3*) can separate the classes with accurate cluster. Figure (a.27) shows clustering error of different number of genes with respect to different DWD levels. Figure (a.28) shows the result of two classes clustering of the microarray samples accurately with accuracy 92%. It is clear to highlight that there are four samples unclassified out of 50 samples in the microarray known as {12, 22, 28, 35} which are particularly difficult to classify since they have weak expression levels.



a- MSE of different DWD level



b- MSE of different number of genes

Figure a.26 miDWD analyses with respect to different number of genes and DWD levels for High-grade glioma dataset

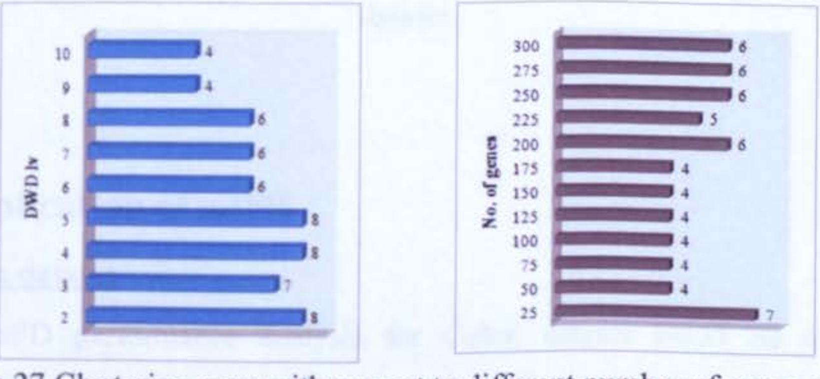
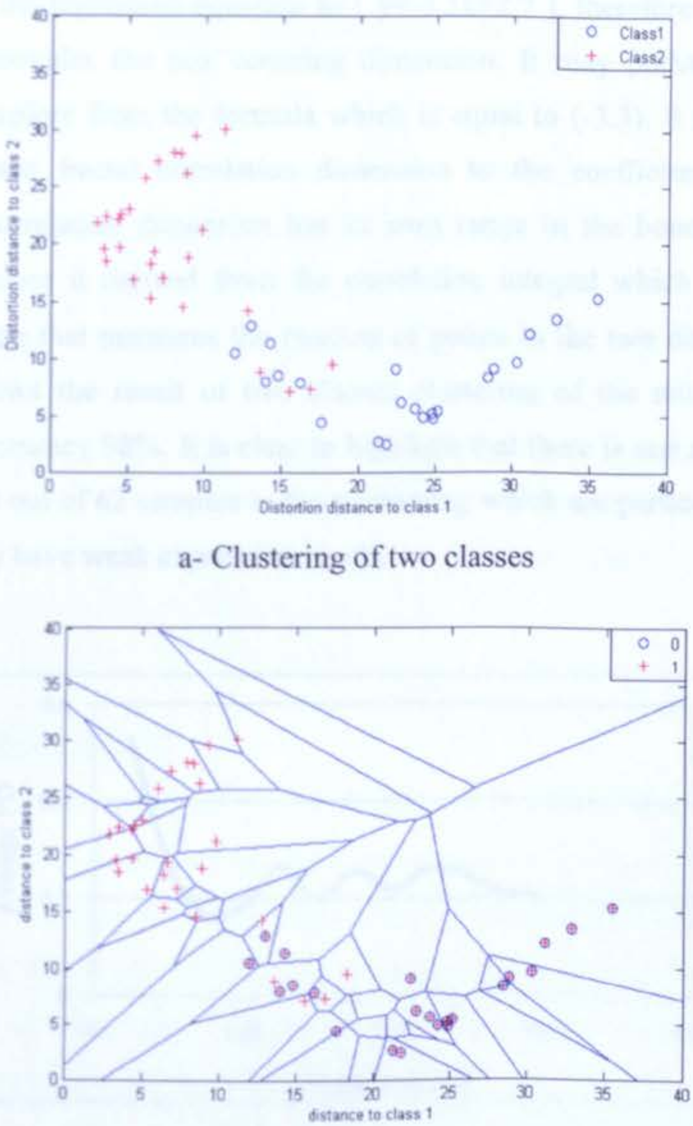


Figure a.27 Clustering error with respect to different number of genes and DWD levels for High-grade glioma dataset



a- Clustering of two classes

b- Voronoi clustering of two classes

Figure a.28 Two class clustering of microarray samples for High-grade glioma dataset

A.3- Application of miFD

1- Colon dataset

The miFD performance analysis for Colon dataset based on multiple run concerning different number of involved genes is shown in Figure (a.29). It presents an observation that the lowest average fractal dimension demonstrated with 75 genes involved which give the lower clustering error. Figure (a.30) plots the local representation of the sample 9. Logarithmic values of box number and box sizes were used to compute the regression equation as [y=-3.3x+2.7], therefore the slope of this regression line provides the box counting dimension. It may perhaps identify as a coefficient and explore from the formula which is equal to (-3.3). It is more likely to establish significant fractal correlation dimension to the coefficient, based on the possibility that correlation dimension has its own range in the boundaries of fractal dimension. Therefore it derived from the correlation integral which is a cumulative correlation function that measures the fraction of points in the two dimensional space. Figure (a.31) shows the result of two classes clustering of the microarray samples accurately with accuracy 98%. It is clear to highlight that there is one sample identified as 51 unclassified out of 62 samples in the microarray which are particularly difficult to classify since they have weak expression levels.

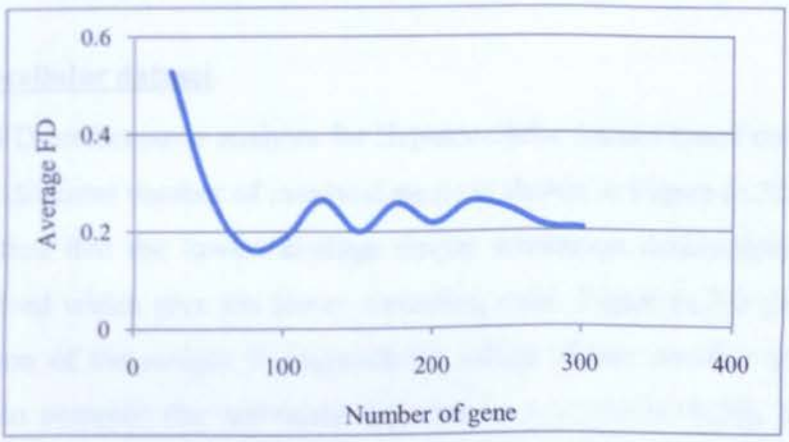


Figure a.29 miFD analyses for Colon dataset

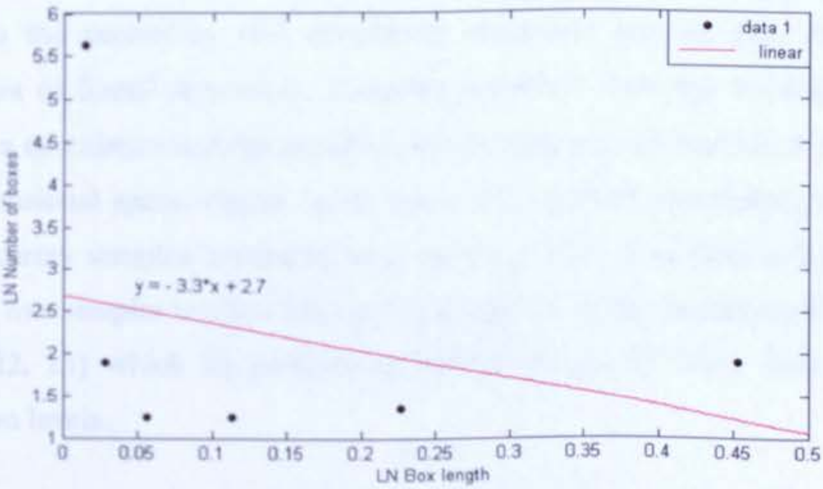


Figure a.30 Local representation of sample 9 of Colon dataset

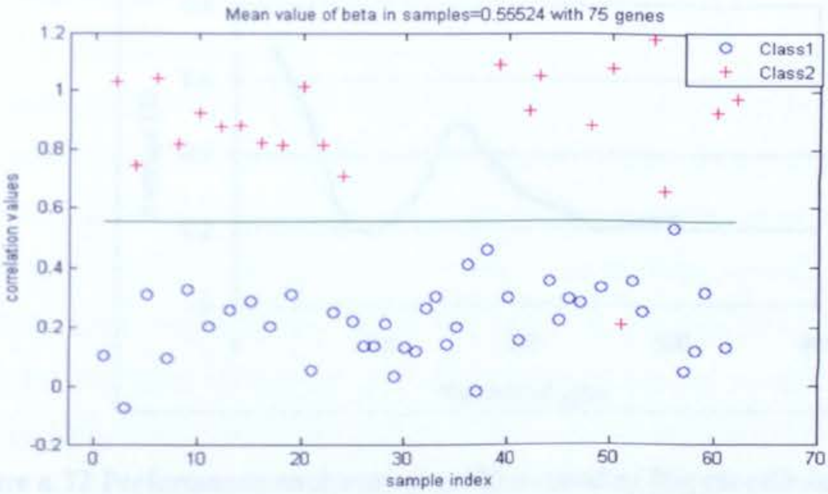


Figure a.31 miFD clustering of Colon dataset

2- Hepatocellular dataset

The miFD performance analysis for Hepatocellular dataset based on multiple run concerning different number of involved genes is shown in Figure (a.32). It presents an observation that the lowest average fractal dimension demonstrated with 100 genes involved which give the lower clustering error. Figure (a.33) plots the local representation of the sample 9. Logarithmic values of box number and box sizes were used to compute the regression equation as [y=-0.15x+0.29], therefore the slope of this regression line provides the box counting dimension. It may perhaps identify as a coefficient and explore from the formula which is equal to (-0.15). It is more likely to establish significant fractal correlation dimension to the coefficient, based on the possibility that correlation dimension has its own range in the boundaries of fractal dimension. Therefore it derived from the correlation integral which is a cumulative correlation function that measures the fraction of points in the two dimensional space. Figure (a.34) shows the result of two classes clustering of the microarray samples accurately with accuracy 92%. It is clear to highlight that there are five samples unclassified out of 60 samples in the microarray known as {6, 18, 19, 22, 23} which are particularly difficult to classify since they have weak expression levels.

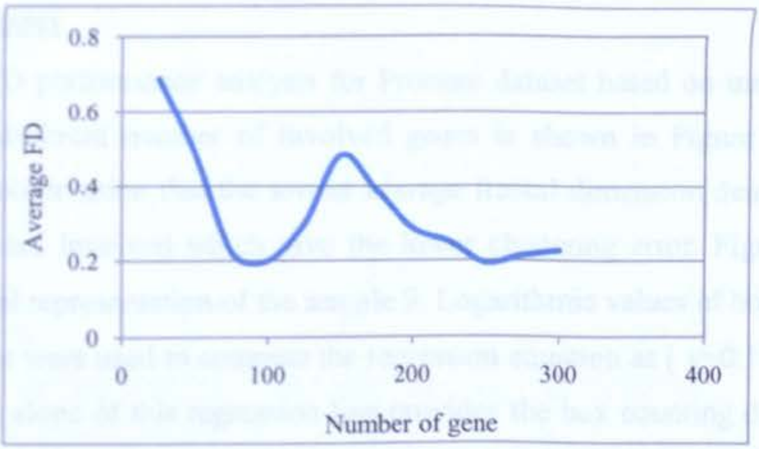


Figure a.32 Performance analyses of miFD method of Hepatocellular dataset

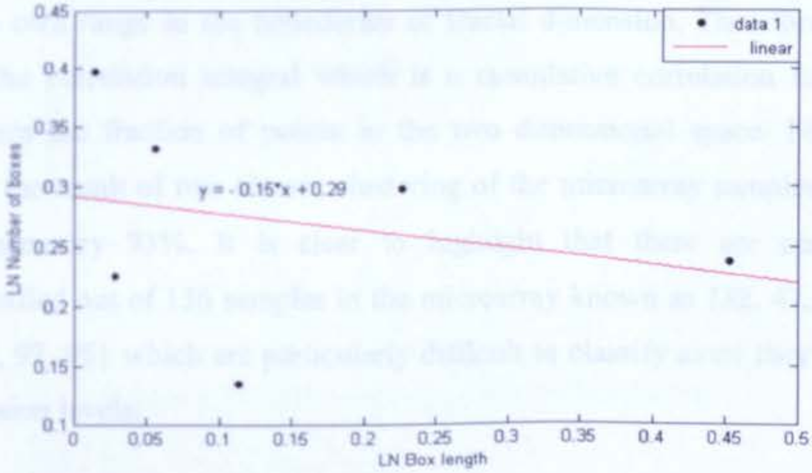


Figure a.33 Local representation of sample 9 of Hepatocellular dataset

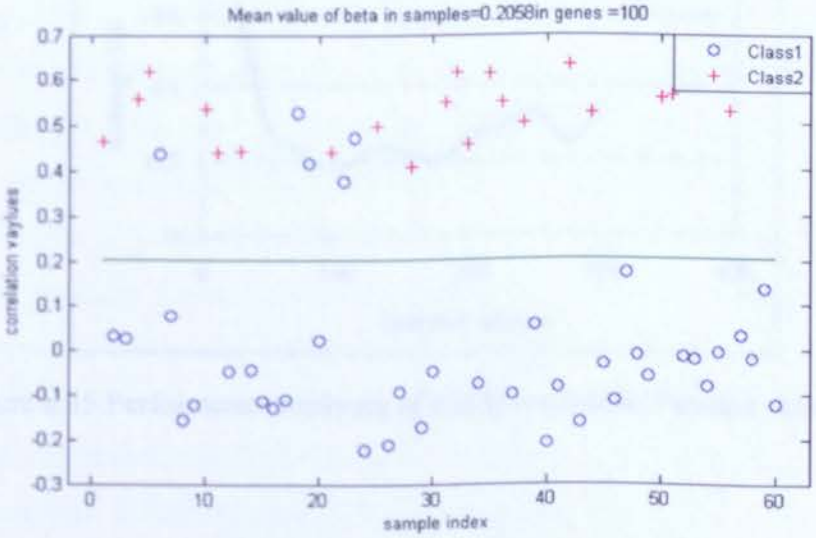


Figure a.34 miFD clustering of Hepatocellular dataset

3- Prostate dataset

The miFD performance analysis for Prostate dataset based on multiple run concerning different number of involved genes is shown in Figure (a.35). It presents an observation that the lowest average fractal dimension demonstrated with 100 genes involved which give the lower clustering error. Figure (a.36) plots the local representation of the sample 9. Logarithmic values of box number and box sizes were used to compute the regression equation as [y=0.17x+0.56], therefore the slope of this regression line provides the box counting dimension. It may perhaps identify as a coefficient and explore from the formula which is equal to (0.17). It is more likely to establish significant fractal correlation dimension to the coefficient, based on the possibility that correlation dimension has its own range in the boundaries of fractal dimension. Therefore it derived from the correlation integral which is a cumulative correlation function that measures the fraction of points in the two dimensional space. Figure (a.37) shows the result of two classes clustering of the microarray samples accurately with accuracy 93%. It is clear to highlight that there are nine samples unclassified out of 136 samples in the microarray known as {32, 47, 57, 59, 68, 81, 84, 92, 95} which are particularly difficult to classify since they have weak expression levels.

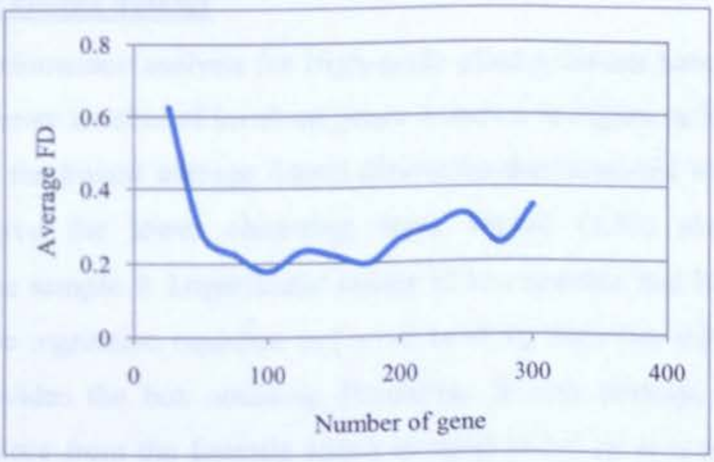


Figure a.35 Performance analyses of miFD method of Prostate dataset

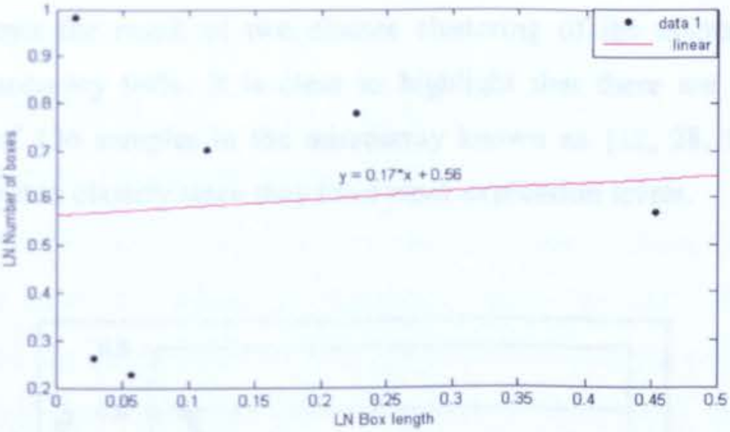


Figure a.36 Local representation of sample 9 of Prostate dataset

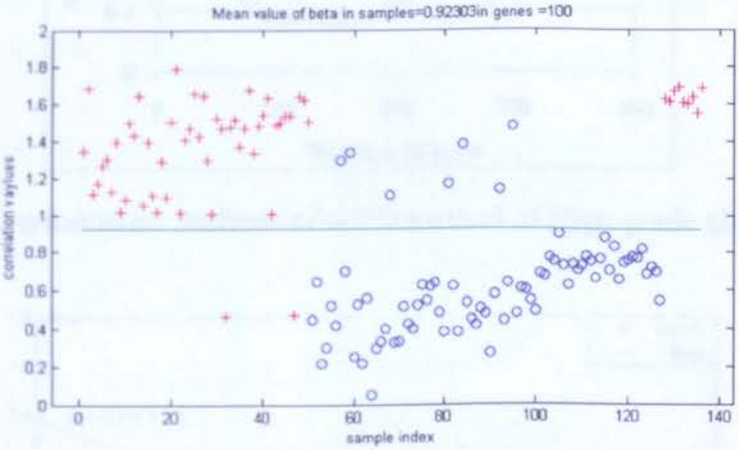


Figure a.37 miFD clustering of Prostate dataset

4- **High-grade glioma dataset**

The miFD performance analysis for High-grade glioma dataset based on multiple run concerning different number of involved genes is shown in Figure (a.38). It presents an observation that the lowest average fractal dimension demonstrated with 100 genes involved which give the lower clustering error. Figure (a.39) plots the local representation of the sample 9. Logarithmic values of box number and box sizes were used to compute the regression equation as [y=-2.1x+1.4], therefore the slope of this regression line provides the box counting dimension. It may perhaps identify as a coefficient and explore from the formula which is equal to (-2.1). It is more likely to establish significant fractal correlation dimension to the coefficient, based on the possibility that correlation dimension has its own range in the boundaries of fractal dimension. Therefore it derived from the correlation integral which is a cumulative correlation function that measures the fraction of points in the two dimensional space.

Figure (a.40) shows the result of two classes clustering of the microarray samples accurately with accuracy 94%. It is clear to highlight that there are three samples unclassified out of 136 samples in the microarray known as {12, 28, 35} which are particularly difficult to classify since they have weak expression levels.

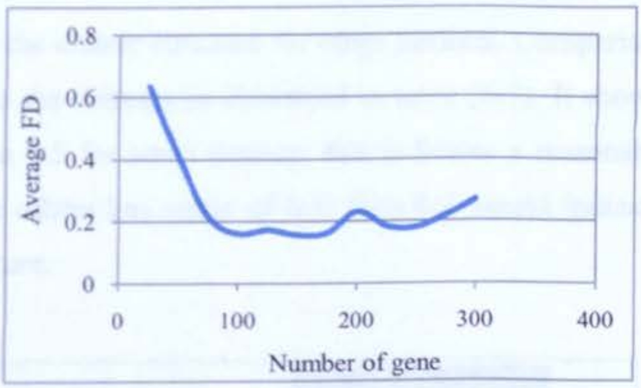


Figure a.38 Performance analyses of miFD method of High-grade glioma dataset

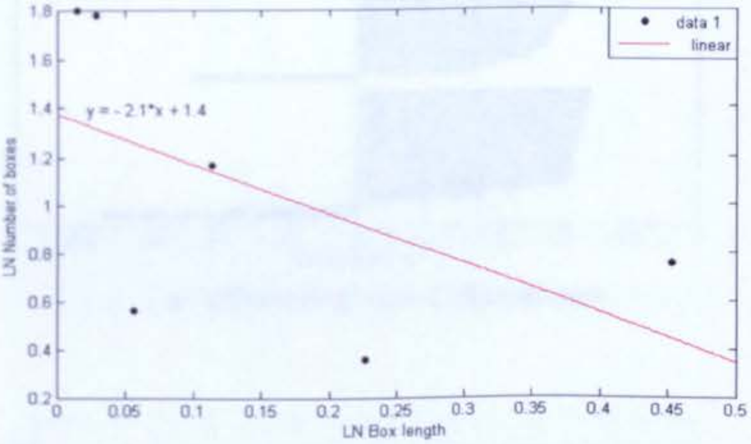


Figure a.39 Local representation of sample 9 of High-grade glioma dataset

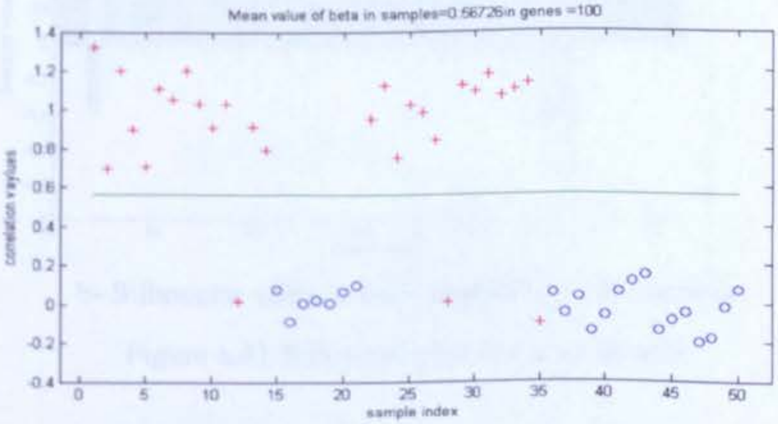


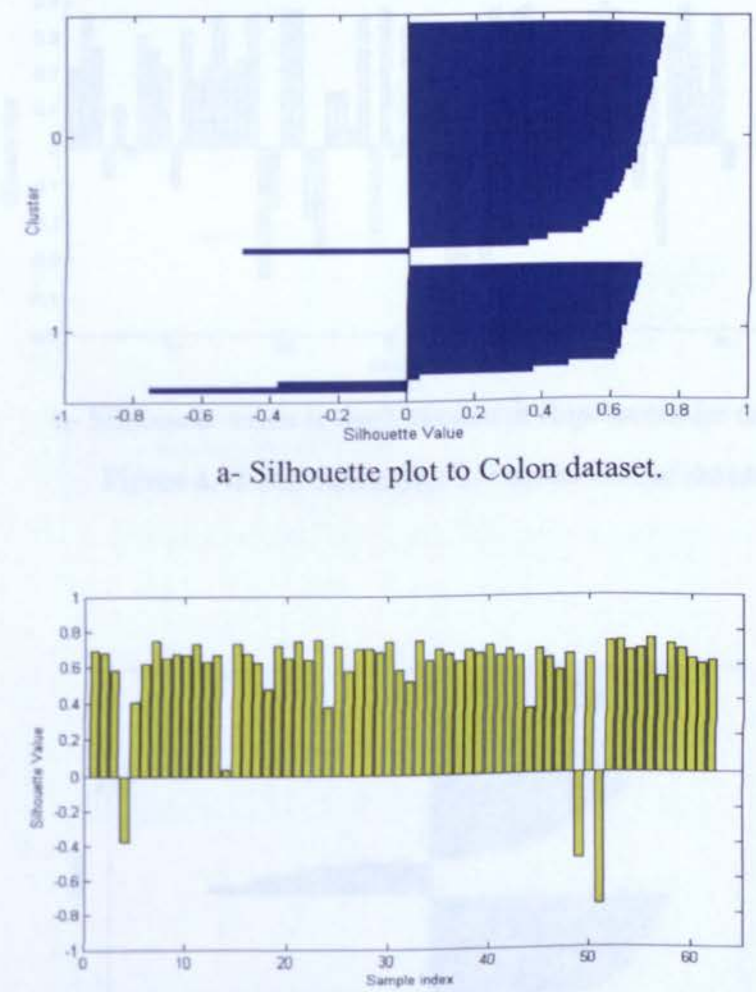
Figure a.40 miFD clustering of High-grade glioma dataset

A.4- Cluster validation methods

A.4.1- Silhouette index

A.4.1.1- Validation of miLPC approach

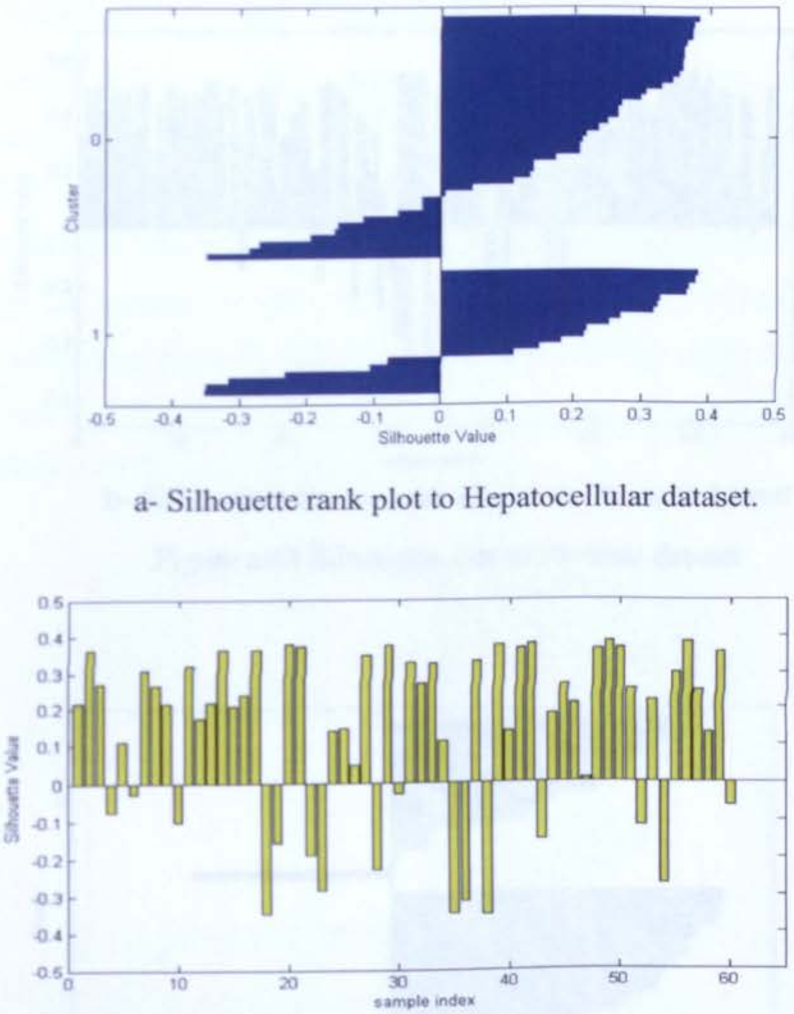
The evaluation of miLPC approach to other datasets is illustrated as follows. Figures (a.41-44) show the silhouette plot index values for each cluster that can visualise and assess the cluster structure for other datasets. Comparison between Global silhouette indexes to the datasets is illustrated in table (6-7). It shows that the average width is greater than 0.5 for some datasets that indicates a reasonable partition of the data samples, while others has value of less than 0.2 would indicate the data do not exhibit cluster structure.



a- Silhouette plot to Colon dataset.

b- Silhouette value to each sample in Colon dataset

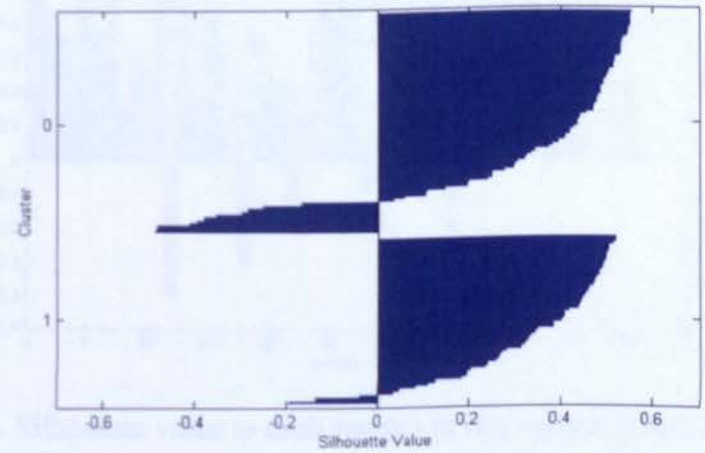
Figure a.41 Silhouette plot to Colon dataset



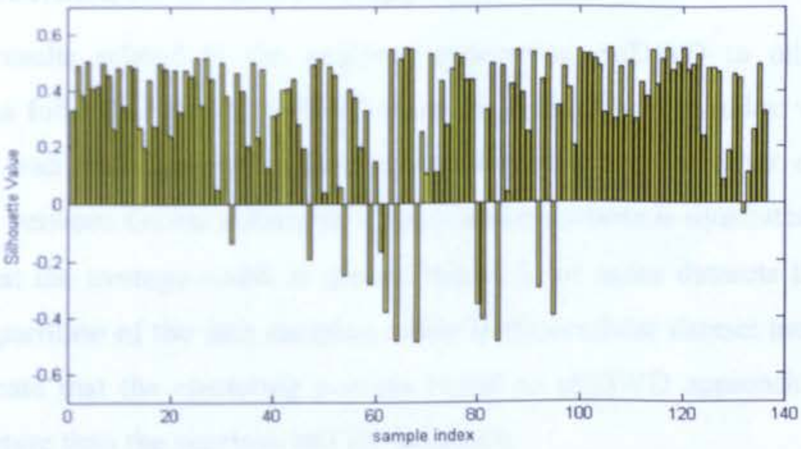
a- Silhouette rank plot to Hepatocellular dataset.

b- Silhouette value to each sample in Hepatocellular dataset

Figure a.42 Silhouette plot to Hepatocellular dataset

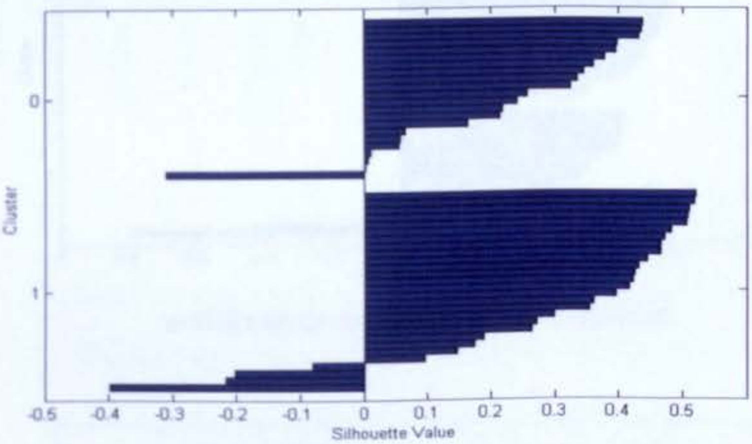


a- Silhouette rank plot to Prostate dataset.

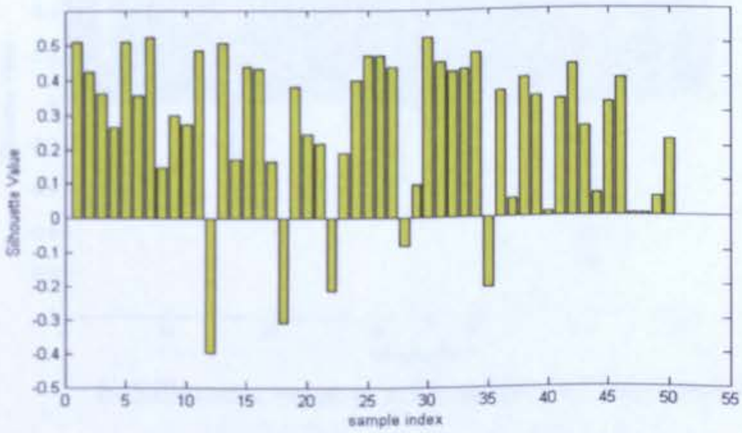


b- Silhouette value to each sample in Prostate dataset

Figure a.43 Silhouette plot to Prostate dataset



a- Silhouette rank plot to High-grade gliom dataset.

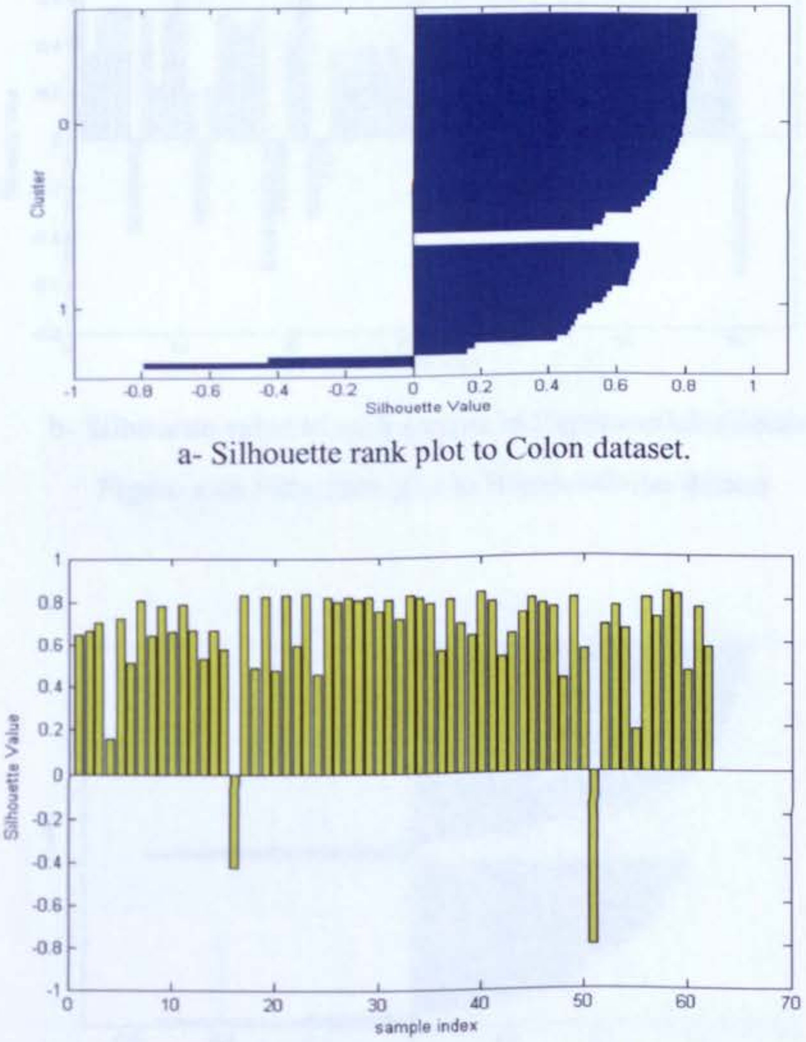


b- Silhouette value to each sample in High-grade gliom dataset

Figure a.44 Silhouette plot to High-grade gliom dataset

A.4.1.2- Validation of miDWD approach

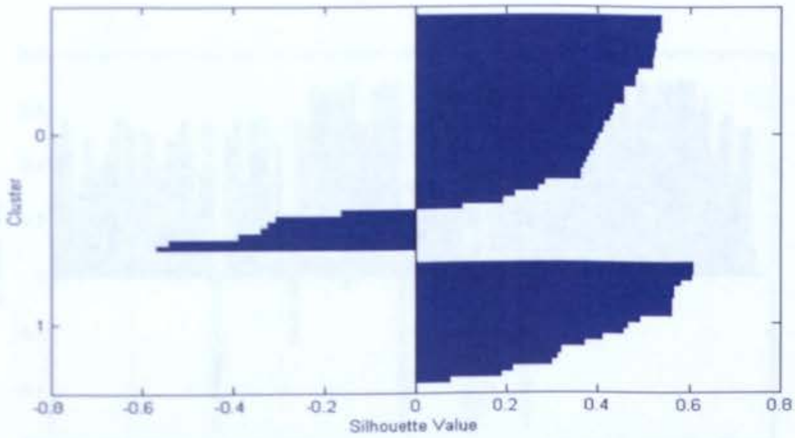
The results related to the analyses concerning miDWD to other datasets is illustrated as follows. Figures (a.45-48) show the silhouette plot index values for each cluster that can visualise and assess the cluster structure for other datasets, while comparison between Global silhouette indexes to the datasets is illustrated in table (a.5). It shows that the average width is greater than 0.5 for some datasets that indicates a reasonable partition of the data samples, while Hepatocellular dataset has value of 0.34 would indicate that the clustering process based on miDWD approach is almost has proper structure than the previous miLPC approach.



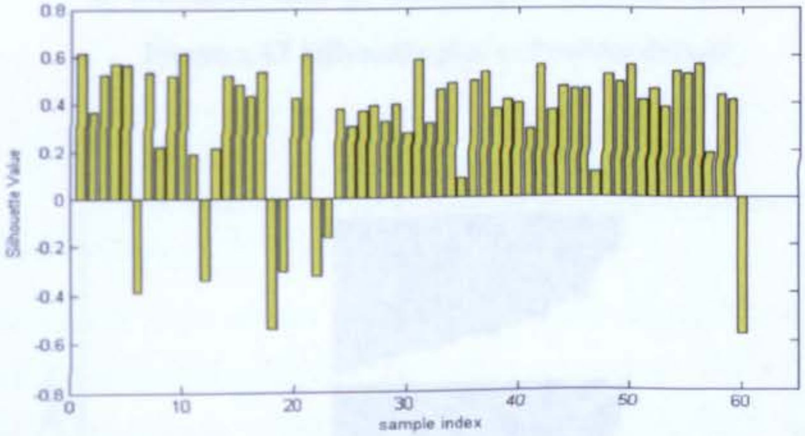
a- Silhouette rank plot to Colon dataset.

b- Silhouette value to each sample in Colon dataset

Figure a.45 Silhouette plot to Colon dataset

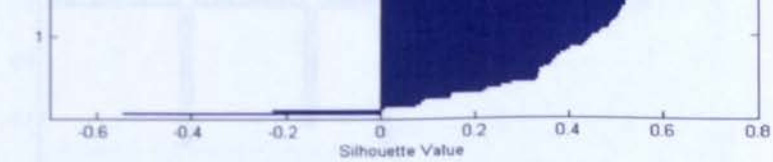


a- Silhouette rank plot to Hepatocellular dataset.

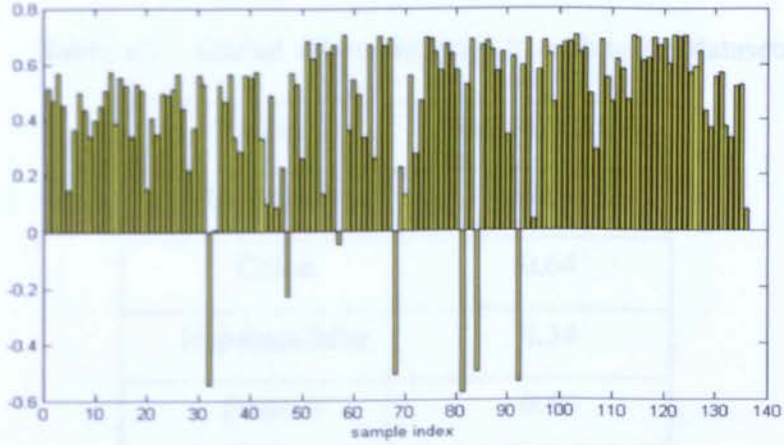


b- Silhouette value to each sample in Hepatocellular dataset

Figure a.46 Silhouette plot to Hepatocellular dataset

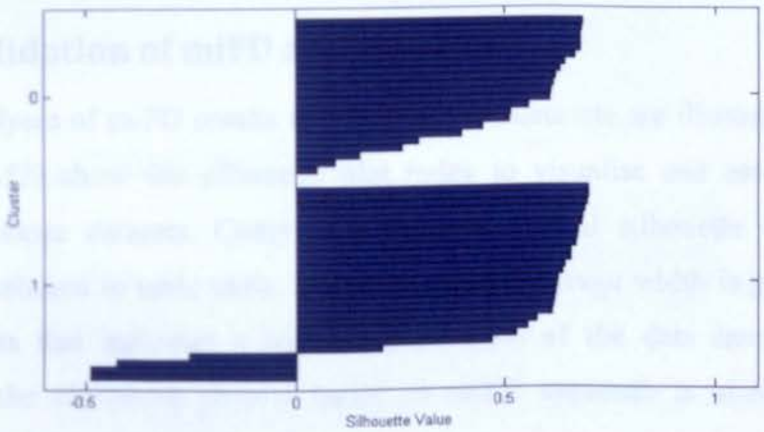


a- Silhouette rank plot to Prostate dataset.

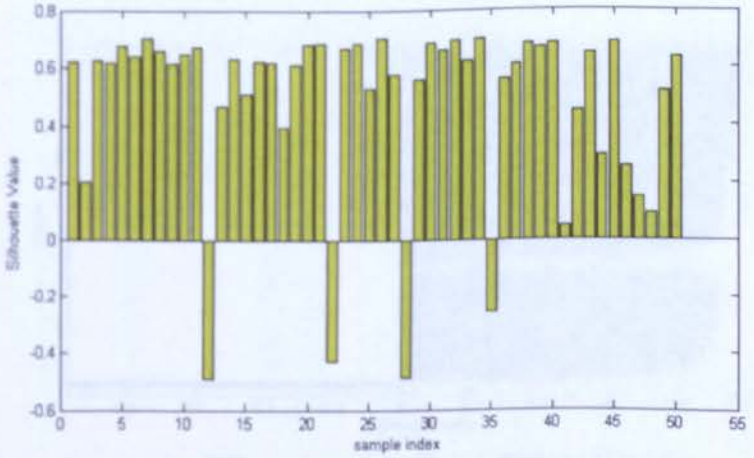


b- Silhouette value to each sample in Prostate dataset

Figure a.47 Silhouette plot to Prostate dataset



a- Silhouette rank plot to High-grade gliom dataset.



b- Silhouette value to each sample in High-grade gliom dataset

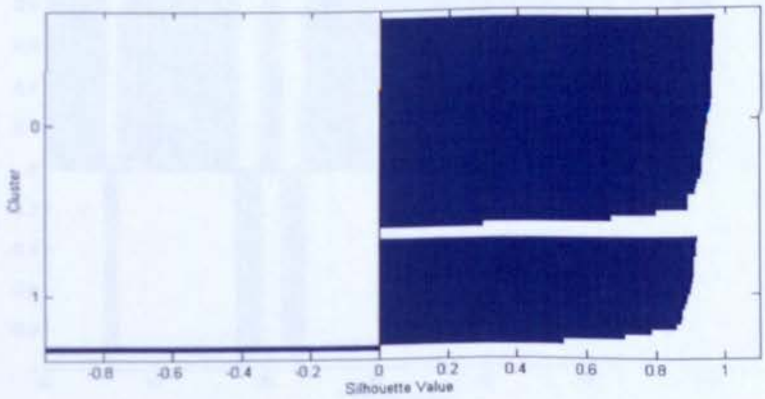
Figure a.48 Silhouette plot to High-grade gliom dataset

Table a.5 Global silhouette index to each tested datasets

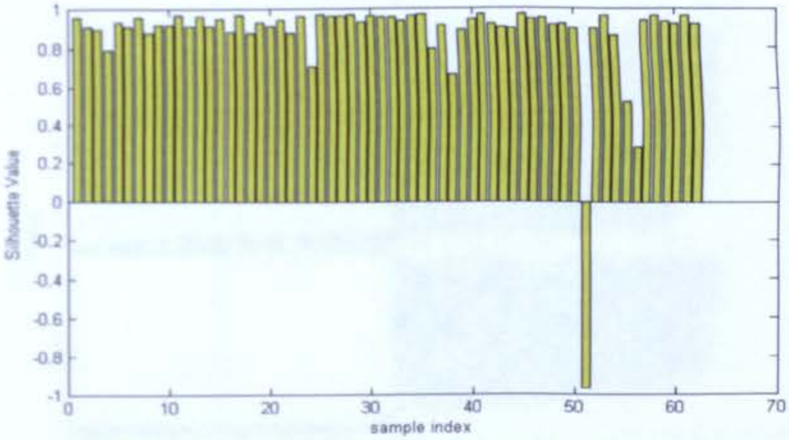
Dataset	miDWD ASW
Leukaemia	0.63
Colon	0.64
Hepatocellular	0.34
Prostate	0.46
Gliomas	0.485

A.4.1.3- Validation of miFD approach

The analyses of miFD results to other selected datasets are illustrated as follows. Figures (a.49-52) show the silhouette plot index to visualise and assess the cluster structure for those datasets. Comparison between Global silhouette indexes to the datasets is illustrated in table (a.6). It shows that the average width is greater than 0.5 for all datasets that indicates a reasonable partition of the data samples. It would indicate that the clustering process based on miFD approach is almost has proper structure than others approaches.

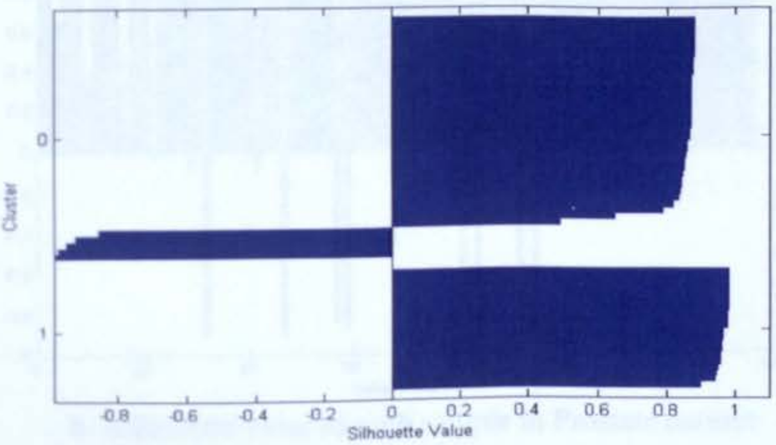


a- Silhouette rank plot to Colon dataset.

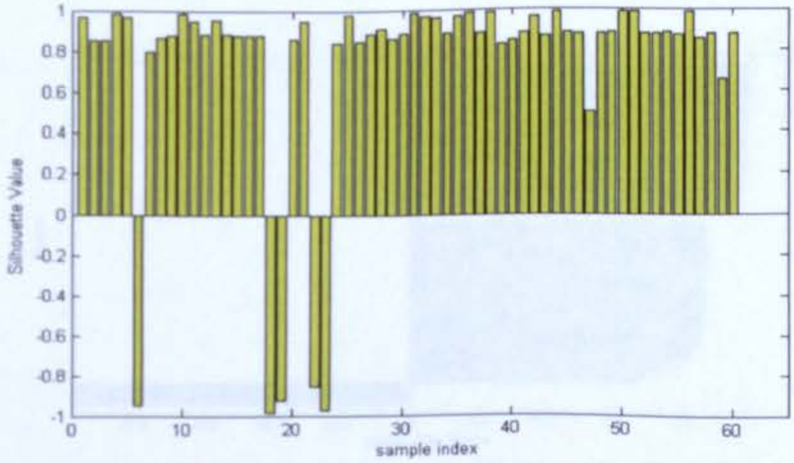


b- Silhouette value to each sample in Colon dataset

Figure a.49 Silhouette plot to Colon dataset

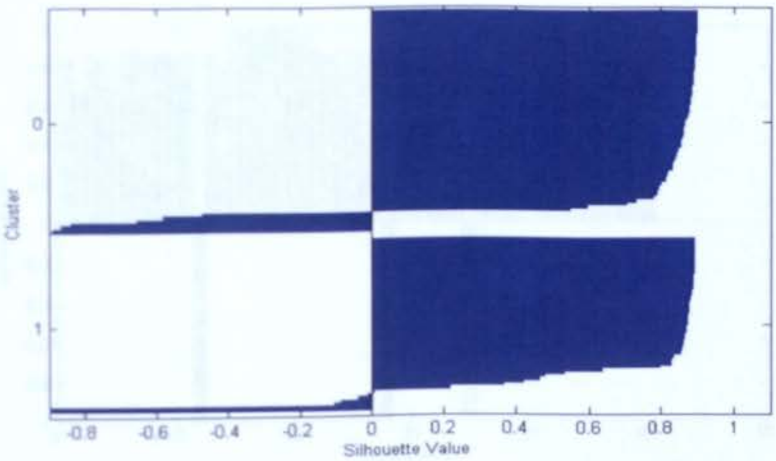


a- Silhouette rank plot to Hepatocellular dataset.

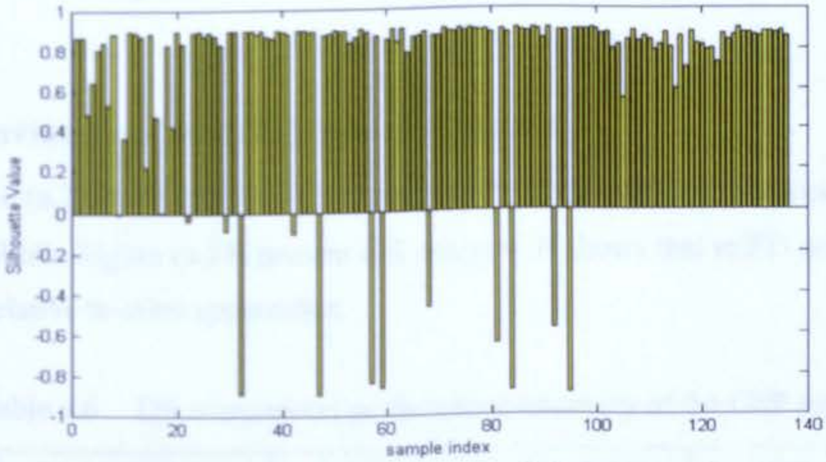


b- Silhouette value to each sample in Hepatocellular dataset

Figure a.50 Silhouette plot to Hepatocellular dataset

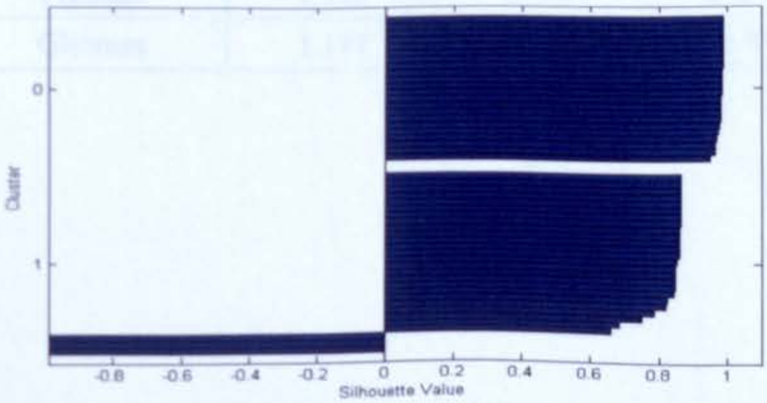


a- Silhouette rank plot to Prostate dataset.

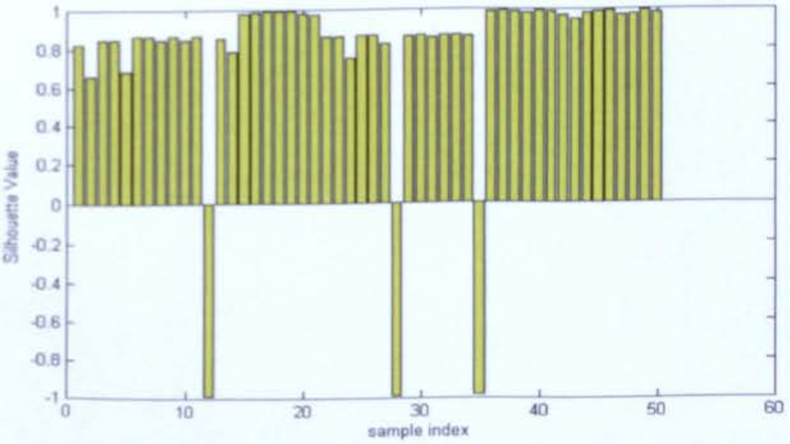


b- Silhouette value to each sample in Prostate dataset

Figure a.51 Silhouette plot to Prostate dataset



a- Silhouette rank plot to High-grade gliom dataset.



b- Silhouette value to each sample in High-grade gliom dataset

Figure a.52 Silhouette plot to High-grade gliom dataset

A.4.2- Davies-Bouldin (DB) index Validation,

Table (a.7) shows the DB-index as a result of the evaluation process to the tested datasets. While Figure (a.54) present DB analysis. It shows that miFD achieve promise result in relative to other approaches.

Table a.6 DB comparison performance summary of the GSP approaches

Dataset	miLPC DB	miDWD DB	miFD DB
Leukemia	0.865	0.574	0.292
Colon	0.605	0.553	0.359
Hepato-cellular	2.14	1.03	0.366
Prostate	1.214	0.885	0.538
Gliomas	1.119	0.706	0.36