PROJECTE FINAL DE CARRERA

Study of gene expression representation with Treelets and hierarchical clustering algorithms

AUTOR: **Pau Bellot Pujalte**

TUTOR: **Philippe Salembier**

Septiembre de 2011

Enginyeria de Telecomunicació

Departament de Teoria del Senyal i Comunicacions

Grup de Processament d'Imatge i Video

Abstract

Since the mid-1990's, the field of genomic signal processing has exploded due to the development of DNA microarray technology, which made possible the measurement of mRNA expression of thousands of genes in parallel. Researchers had developed a vast body of knowledge in classification methods. However, microarray data is characterized by extremely high dimensionality and comparatively small number of data points. This makes microarray data analysis quite unique. In this work we have developed various hierarchical clustering algorthims in order to improve the microarray classification task.

At first, the original feature set of gene expression values are enriched with new features that are linear combinations of the original ones. These new features are called metagenes and are produced by different proposed hierarchical clustering algorithms. In order to prove the utility of this methodology to classify microarray datasets the building of a reliable classifier via feature selection process is introduced. This methodology has been tested on three public cancer datasets: Colon, Leukemia and Lymphoma. The proposed method has obtained better classification results than if this enhancement is not performed. Confirming the utility of the metagenes generation to improve the final classifier.

Secondly, a new technique has been developed in order to use the hierarchical clustering to perform a reduction on the huge microarray datasets, removing the initial genes that will not be relevant for the cancer classification task. The experimental results of this method are also presented and analyzed when it is applied to one public database demonstrating the utility of this new approach.

Resumen

Desde finales de la década de los años 90, el campo de la genómica fue revolucionado debido al desarrollo de la tecnología de los DNA microarrays. Con ésta técnica es posible medir la expresión de los mRNA de miles de genes en paralelo. Los investigadores han desarrollado un vasto conocimiento en los métodos de clasificación. Sin embargo, los microarrays están caracterizados por tener un alto número de genes y un número de muestras comparativamente pequeño. Éste hecho convierte al estudio de los microarrays en único. En éste trabajo se ha desarrollado diversos algoritmos de agrupación jerárquica para mejorar la clasificación de los microarrays.

La primera y gran aplicación ha sido el enriquecimiento de las bases de datos originales mediante la introducción de nuevos elementos que son obtenidos como combinaciones lineales los genes originales. Estos nuevos elementos se han denominado metagenes y son producidos mediante los diferentes algoritmos propuestos de agrupación jerárquica. A fin de demostrar la utilidad de esta metodología para clasificar las bases de datos de microarrays se ha introducido la construcción de un clasificador fiable a través de un proceso de selección de características. Esta metodología ha sido probada en tres bases de datos de cáncer públicas: Colon, Leucemia y Linfoma. El método propuesto ha obtenido mejores resultados en la clasificación que cuando éste enriquecimiento no se ha llevado a cabo. De ésta manera se ha confirmado la utilidad de la generación de los metagenes para mejorar el clasificador.

En segundo lugar, se ha desarrollado una nueva técnica para realizar una reducción inicial en las bases de datos, consistente en eliminar los genes que no son relevantes para realizar la clasificación. Éste método se ha aplicado a una de las bases de datos públicas, y los resultados experimentales se presentan y analizan demostrando la utilidad de éste nuevo enfoque.

Resum

Des de finals de la dècada dels 90, el camp de la genómica va ser revolucionat gràcies al desenvolupament de la tecnología dels DNA microarrays. Amb aquesta tècnica es possible mesurar l'expresió dels mRNA de milers de gens en paralel. Els investigadors han desenvolupat un ample coneixement dels mètodes de classificació. No obstant, els microarrays estàn caracteritzats per tindre una alt nombre de genes i comparativament un nombre petit de mostres. Aquest fet fa que l'estudi dels microarrays sigui únic. Amb aquest treball s' han desenvolupat diversos algoritmes d'agrupació jeràrquica per millorar la classificació dels microarrays.

La primera i gran aplicació ha sigut l'enriqueiment de les bases de dades originals mitjançant l'introducció de nous elements que s'obtenen com combinacions lineals dels gens originals. Aquests nous elements han sigut denominats com metagens i són calculats mitjantçant els diferents algoritmes d'agrupació jerárquica proposats. Per a demostrar l'utilitat d'aquesta metodología per a classificar les bases de dades de microarrays s'ha introduït la construcció d'un classificador fiable mitjantçant un procés de selecció de característiques. Aquesta metodología ha sigut aplicada a tres bases de dades públiques de càncer: Colon, Leucèmia i Limfoma. El métode proposat ha obtenigut millors resultats en la classificació que quan aquest enriqueiment no ha sigut realitzat. D'aquesta manera s'ha confirmat l'utilitat de la generació dels metagens per a millorar els classificadors.

En segon lloc, s'ha desenvolupat una nova técnica per a realitzar una reducció inicial en les bases de dades, aquest mètode consisteix en l'eliminació dels gens que no són relevants a l'hora de realitzar la classificació dels pacients. Aquest mètode ha sigut aplicat a una de les bases de dades públiques. Els resultats experimentals es presenten i analitzen demostrant l'utilitat d'aquesta nova tècnica.

Agradecimientos

Quiero expresar mis más sincero agradecimiento al *Grupo de Imagen* por darme la oportunidad de realizar este proyecto final de carrera, así como a toda la gente que ha colaborado aportando información y su tiempo para ayudarme en el desarrollo de este proyecto.

Especialmente, me gustaría darle las gracias una vez más a mi tutor Philippe Salembier, que me ha proporcionado asesoramiento y entusiasmo a lo largo de todo el proyecto incluido el mes de vacaciones de Agosto. Gracias por el tiempo dedicado a escuchar mis ideas y resolver mis dudas fueran de la temática que fueran. Sin su apoyo y confianza, el resultado no habría sido igual de exitoso.

También me gustaría agradecer a todos los amigos de esta universidad que me han acompañado en este segundo ciclo, sin ellos la estancia en esta universidad y en Barcelona no hubiera sido tan agradable. Mis ánimos a todos aquellos que todavía están acabando la carrera.

Por último y no menos importante, un agradecimiento muy especial a mis padres y mi hermana por todo el apoyo mostrado y su ayuda, y sin olvidar el cariño de mis abuelos.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Traditionally, techniques for the study of gene expression were significantly limited in both breadth and efficiency since these studies typically allowed investigators to study only one or a few genes at a time.

However, the DNA microarray technique is a powerful method that provides researchers with the opportunity to analyze the expression patterns of tens of thousands of genes in a short time. Microarray technology is a powerful approach for genomics research. The multi-step, data-intensive nature of this technology has created an unprecedented informatics and analytical challenge.

Microarray technology has become a standard tool in many genomics research laboratories. The reason for this popularity is that microarrays have revolutionized the approach to biological research [1]. Unfortunately, data analyses are often very complex, daunting and confusing.

1.2 Motivation of this project

Microarray technology allows researchers to analyze patterns of gene expression with the goal of providing useful information for disease diagnosis or prognosis. These datasets tend to have a large number of gene expression values per sample (several thousands to tens of thousands, even millions), and a relatively small number of samples (e.g., a few dozen samples in relatively rare types of cancer) due to the cost of the experiments. While each sample contains expression information for many genes, it is likely that only a small subset of genes are relevant to a

specific diagnosis/prognosis problem. Thus, it could be possible to achieve good discrimination of a sample by using only a small fraction of the original gene vector (which contains all expression data).

Designing such *sparse* classification approaches is likely to be more meaningful from a biological point of view, since mechanisms that lead to specific diseases are thought to involve relatively small numbers of genes. Feature selection and feature transformation are popular tools to design efficient (and sparse) classifiers starting from high dimensional expression data, such as microarray datasets.

This characteristic of sample scarcity makes necessary a feature selection process to produce reliable classifiers [2]. Furthermore, algorithms like Tree Harvesting found useful the feature set expansion via hierarchical clustering [3].

A technique named Treelets [4] has been proposed as a new adaptive multi-scale representation for sparse unordered data, that represents the data in a hierarchical tree, and it has been the starting point for this PFC project.

Moreover, the Department of Signal Theory and Communications (TSC) of the UPC has signed a collaboration agreement with the CELLEX foundation a year ago with the aim to apply Digital Signal Processing tools for analyzing the data generated in the study of cancer. One of the research lines of this project is to analyze the data that come from DNA Microarrays which measure the expression (over or under-expression) of the different genes (around 60000 per patient, depending on the microarray technology), and different kinds of cancer. This is a long-term project in which this PFC is embedded.

At this moment, this collaboration agreement has three specific points:

- Develop an automatic method to classify each new patient sample in terms of the kind or stage of the disease, with a small margin of error.
- Identify what is the minimum number of genes or markers that characterizes a particular disease or its stage.
- Find a way to represent the hierarchy of interactions between genes.

1.3 Project goals

The initial objective of this project is to study the Treelet technique when it is applied to microarray datasets and then, based on this study, to propose some changes in order to optimize or adapt the technique to this kind of data.

The second objective is to study the characteristics of the proposed changes in order to expand the original data with new features that are linear combinations of the original genes and to prove that with those new enriched data it is possible to perform a better classification of the patients (with a lower error rate and greater reliability).

Finally, as technology of Microarrays advances, the number of *probesets* (currently about 60000) per sample analyzed also increases. The analysis time (computational cost) with the tools developed increases exponentially with the number of observed *probesets*. Therefore, the last objective of this PFC project is to use the hierarchical clustering in order to develop a method for removing those initial genes that are not relevant.

1.4 Report Structure

This report is divided in 7 chapters, including this first introduction chapter and the final conclusions and future work lines. The remaining chapters follow the chronological order of the project development, from the analysis of the microarray datasets to the application and validation of the proposed algorithms.

Chapter 2 gives a short introduction to the microarray datasets, starting from the most general concepts of biological issues and the construction of microarrays and then focusing on their problematics.

In chapter 3, the main contributions and improvements to the Treelet technique are presented in order to adapt it to the microarray datasets. At the end, the main characteristics of proposed hierarchical clustering algorithms are explained for a particular cancer dataset.

In chapter 4 a brief review about a reliable classifier based on feature selection process for gene expression classification is done. With this method, a particular gene (or sets of genes) is tested to assess whether it is relevant for classifying a patient with the same error and reliability.

Chapter 5 presents the results of our experiments with different databases. Results are also compared with the *state of the art* in the microarray and cancer classification area. The error classification rates obtained using the techniques described in chapters 3 and 4 are compared with these obtained by other researchers and other techniques using the same publicly available databases.

Chapter 6 presents the development of a new technique to use the hierarchical clustering for performing a reduction on the huge microarray databases, removing the initial genes that will not be relevant for the cancer classification task. At the end of the chapter the results of this method are presented and analyzed when it is applied to one public database.

Finally, chapter 7 presents the conclusions reached with the tools developed during this PFC project.

Additionally, in the appendix A a brief study about the computational cost of this technique is presented in terms of RAM usage and CPU time with different sizes of datasets.

Chapter 2

Genomics review

Genetics is the study of genes, and tries to explain what they represent and how they work. In genetics, a feature of a living thing is called a *trait*. Some traits are part of an organism's physical appearance; such as a person's eye-color, height or weight. Other sorts of traits are not easily seen and include blood types or resistance to diseases. The way our genes and environment interact to produce a trait is very complicated. For example, the chances of somebody dying of cancer or heart disease seems to depend on both their genes and their lifestyle.

Genes are made of a long molecule called DNA, which is copied and inherited across generations. DNA is made of simple units that line up in a particular order within this large molecule. The order of these units carries genetic information, similar to how the order of letters on a text carries information. The language used by DNA is called the genetic code, which lets organisms read the information in the genes.

This information are the instructions for constructing and operating a living organism. Genetic disorders are diseases that are caused by alterations in the genes and are inherited in families. Most of these diseases are inherited in a complex way, with either multiple involved genes, or coming from both genes and the environment. As an example, the risk of breast cancer is 50 times higher in the families most at risk, compared to the families least at risk. Several of the involved genes have been identified but not all of them [5].

2.1 Measuring variation in the levels of RNA expression

When the genes are active, they produce messages (mRNA), used to provide the information needed to make molecules called proteins in the cells, this process is known as protein synthesis. There is a simple division of labor in the cells: genes give instructions and proteins carry out these instructions, for performing tasks like building a new copy of a cell or repairing damage. Each type of protein is a specialist that only does one job, so if a cell needs to do something new, it must make a new protein to do this job. Similarly, if a cell needs to do something faster or slower than before, it makes more or less of the corresponding protein. Genes tell cells what to do by telling them which proteins to make and in what amounts [6]. Since mRNA is an essential product for synthesizing proteins, the mRNA levels can provide a quantification of gene expression levels. Thus, a gene expression level is thought to be correlated with the approximate number copies of mRNA produced in a cell.

The aim of the microarrays is to measure all of the RNA messages that are going on inside of the cell, and therefore get a rich description of the biology of each cell and each disease. Thus providing an insight into which genes are expressed in a particular cell type, at a particular time, under particular conditions. With this analysis a much richer description of the cell is obtained than using a microscope or an enzyme test, for example.

2.2 DNA arrays

Many types of DNA arrays exist. The traditional DNA array is a collection of orderly microscopic *spots*, called features, each with a specific probe attached to a solid surface, such as glass, plastic or silicon biochip. Commonly it is known as a genome chip, DNA chip or gene array, which can be observed in Figure 2.1. Thousands of spots can be placed in known locations on a single DNA chip.

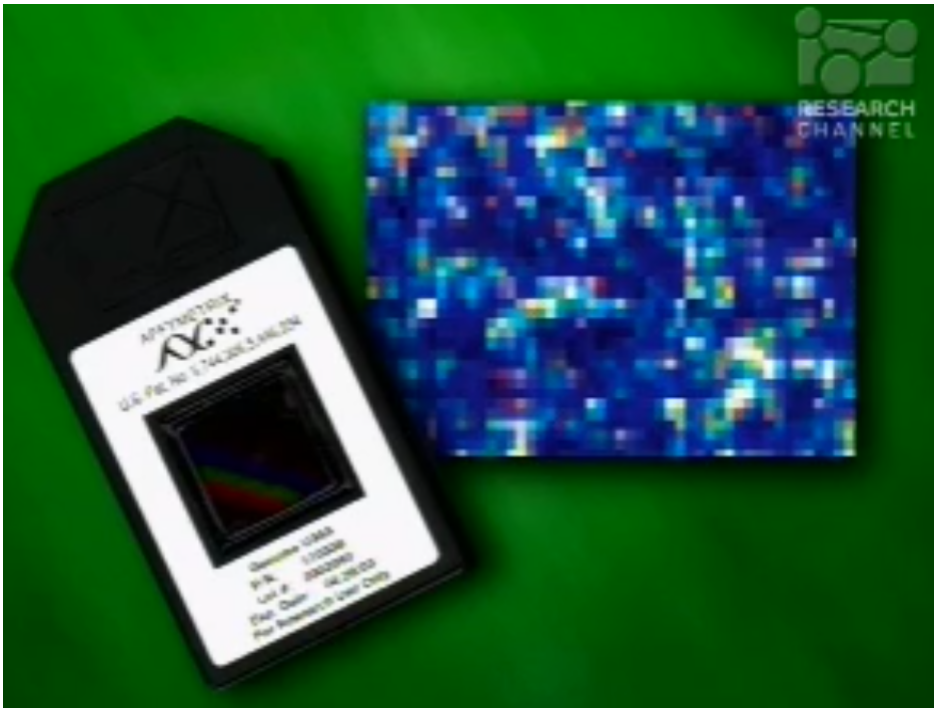
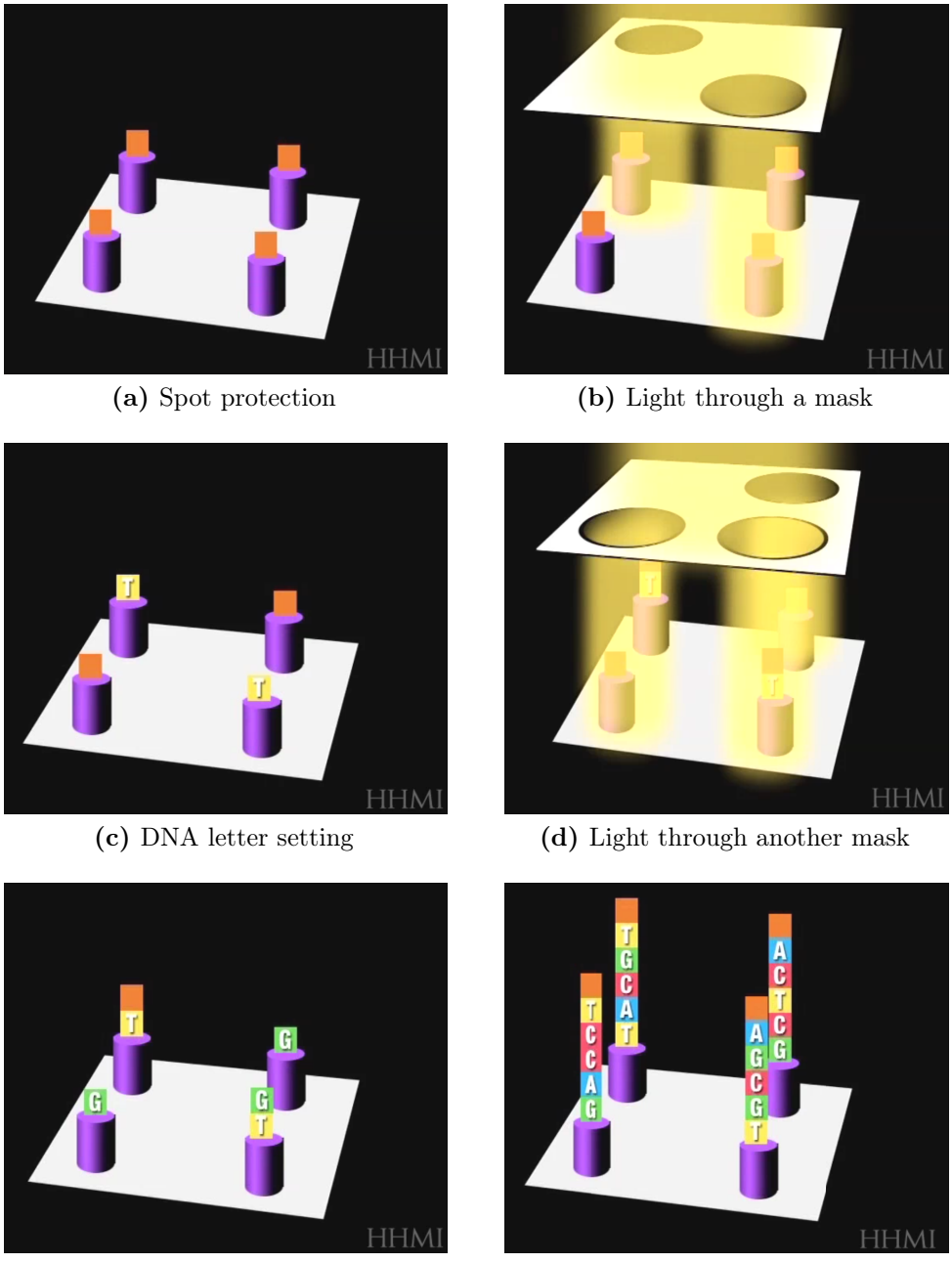


Figure 2.1. *One Affymetrix chip and features example, slide taken from [7].*

Each square of the chip has a different DNA sequence, which is a specific 25-letter DNA sequence in every square. Every one of these has whatever DNA sequence is wanted to be specified. This DNA chip will be used to access to the genomic expression of a single sample.

2.2.1 Construction of DNA chips

The construction of a biochip is done in the same way as the microprocessor chips are built. Every spot is protected (see Figure 2.2a) and a light is shined through a mask (see Figure 2.2b), and the surface is deprotected, then, one of the DNA letters is set (see Figure 2.2c). Then, the surface is re-protected, the light is shined through another mask and deprotect certain spots (see Figure 2.2d) in order to fix the next letter (see Figure 2.2e). After 100 masks it is possible to build up an average of about 25 specific letters in each spot (see Figure 2.2f). In each spot the complementary sequence of every gene in the human genome is built, therefore every spot is a detector for its own gene.



(a) Spot protection **(b)** Light through a mask

(c) DNA letter setting **(d)** Light through another mask

(e) Fixation of another DNA letter **(f)** Results after several DNA letters

Figure 2.2. *Process of construction of a biochip, images taken from [7].*

2.2.2 Use of biochips

The char flow of a typical DNA array experiment is shown in Figure 2.3. As can be observed, the first step is to take the RNA from the cell or the tumor, and inject it into the biochip. Each RNA will stick to its own detector, and then with a scanner the intensity of each spot will be read out. This intensity reflects how much each gene is turned on and off (active or not).

Summing up, each one of those chips converts the tumor or cell into a set of gene sequence, which is a long string of data for each patient, telling us which genes are highly expressed or not.

Finally, several gene sequences (biochips) from differnt patients are grouped in a DNA microarray. The microarrays technique delivers as raw data the information of several patients, stacking all their DNA arrays in a single representation, the various genes are stored in columns and the various patients in rows.

2.3 Learning and classifying Microarray datasets

In some diseases as Leukemia cancer, it has been observed that for different kind of Leukemias, genes expressions may be opposite, while a particular gene may have a high expression for one type of Leukemia, it may have a low expression for an other type [8]. The goal of microarray cancer classification is to identify this kind of distinguishing set of genes.

Thanks to this technique the scientific community is building global cancer maps, trying to get the whole expression patterns of RNA variation in a lot of different tumors. The objective is to learn from those different cancers and to understand the difference between them.

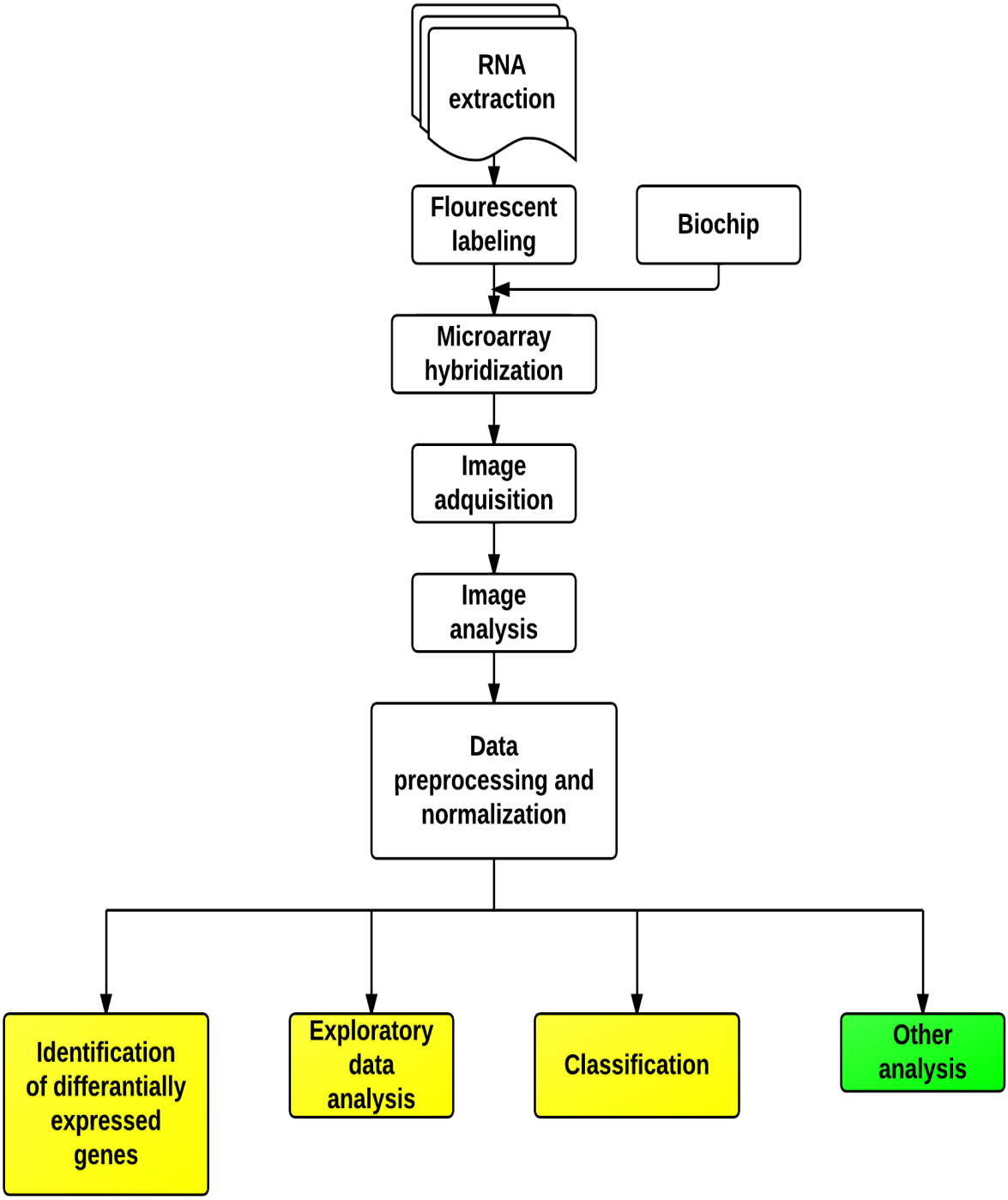


Figure 2.3. *Flow of a typical microarray experiment. The RNAs are extracted and labeled with different fluorescent dyes, and co-hybridized to a microarray. It is scanned to acquire the fluorescent images. An image analysis is performed to obtain the raw signal data for every spot. The poor quality data are filtered out, then the remaining data is normalized. The colored blocks represent the work areas of our study. This figure has been taken and modified partially from [9].*

2.4 Problematics of microarrays technology

Despite all the efforts in the last years, the exact number of genes needed to make a human being is still not known. The computing technologies that recognize transcription units in raw genomic information are not broadly being used, because most the of works only have been appearing in research since the year 2001.

It has been discovered that gene expression patterns differ greatly between genetically identical individuals of the same age, sex, and physiological condition. It is accepted that, for any particular parameter, physiological *normality* is not a strict value but is rather a range of values presented by healthy individuals. Therefore, *normal* gene expression displays similar variability. Such variability between individuals emphasizes the need to join together samples, and to perform a statistical analysis.

Since a lot of genes in the human genome are cancelled by the action of another one, the challenge remains to determine exactly what all of the genes do in terms of the development and physiological functioning of the organism. This fact has motivated an emerging approach, Pathway Analysis that considers that genes work in a cascade of networks and never act alone in a biological system.

Another problem in the microarrays technique is the inherent fluorescence of the surface of the biochip that can contribute greatly to background *noise*. This effect may contaminate samples of patients and make its expression to resemble to be a different class of patient.

Chapter 3

Hierarchical clustering algorithms adapted to microarray datasets

3.1 Motivation and representation

As previously explained in RNA microarray datasets, the expression levels of several thousands of genes are collected. It is possible to access to the expression information of a huge quantity of them. Microarray data is characterized by extremely high dimensionality and comparatively small number of data points. Additionally, many variables are related to each other or are irrelevant for a particular problem. Hierarchical clustering algorithms [10] have been used for organizing and grouping the dataset variables. These methods offer an easily interpretable description of the data structure in terms of a dendrogram which is a tree used to illustrate the arrangement of the clusters produced by the hierarchical clustering. In order to build the dendrogram for gene clustering, the only requirement is to specify a model to represent gene clusters and a measure of similarity between the clusters. In the context of RNA microarray, the hierarchical clustering approach is a bottom up algorithm: the individual genes correspond to the tree leaves and each gene is assigned to its own cluster. They are the starting points of the algorithm, and then iteratively, at each level, the two gene clusters with highest similarity are merged into a larger cluster, and then, the new cluster is represented by a model.

In [11, 12] it is proved that this way to deal with RNA microarray has some benefits for exploring and visualizing microarray data, and it is well recognized in the field. Also, in order to define a classifier in a diagnosis / prognosis problem, feature selection and feature transformation are popular tools to design efficient (and sparse) classifiers starting from high dimensional expression data. Consequently, to define our hierarchical clustering algorithm for RNA microarrays we

must define these two notions: how to measure the similarity between the gene clusters and how the two variables selected for merging should be represented to form a cluster.

3.2 Cluster representation

In a bottom up hierarchical clustering each observation starts in its own cluster, and pairs of clusters are merged as moving up in the hierarchy. In all the section, it is assumed that two gene clusters $\mathbf{gene}_\alpha$ and $\mathbf{gene}_\beta$ have been selected to be merged and generate what in this work has been named as a metagene. The different methods that we have used to represent these clusters are explained in this section.

3.2.1 PCA based on the two child nodes (Rotation)

Principal Component Analysis (PCA) [13] can be seen as a representation of the data and, it is mathematically described as a change of basis in a vectorial space. It has been demonstrated that PCA achieves a compact representation of the data. As originally described, PCA is a global feature transformation (i.e. the new representation is obtained as a linear combination of all components). So, for this work, it is one of the drawbacks of PCA, which creates a linear combination of the whole dataset. In order to solve this problem, in [4] it has been proposed to perform a local pair-wise PCA to create a hierarchical tree of gene clusters. This representation offers an easily interpretable description of the data structure in terms of a dendrogram. To perform the local PCA on the pair of clusters, a rotation is computed [14]:

$$\begin{aligned} \mathbf{s} &= \mathbf{gene}_\alpha \cdot \cos\theta_L + \mathbf{gene}_\beta \cdot \sin\theta_L \\ \mathbf{d} &= \mathbf{gene}_\beta \cdot \cos\theta_L - \mathbf{gene}_\alpha \cdot \sin\theta_L \end{aligned} \tag{3.1}$$

where θ_L is the angle of rotation that decorrelates the two variables $\mathbf{gene}_\alpha$ and $\mathbf{gene}_\beta$, that is, after the rotation the two new variables ($\mathbf{s}$ and $\mathbf{d}$) will have 0 correlation. As in the typical multi-resolution analysis the two gene clusters $\mathbf{gene}_\alpha$ and $\mathbf{gene}_\beta$ are grouped together and replaced by a coarse-grained *approximation variable* ($\mathbf{s}$) and a residual *detail variable* ($\mathbf{d}$). Both variables are computed from a local principal component analysis (perfomed by the rotation of Equation 3.1) in two dimensions. The approximation variable will be used to represent both clusters, if the two merged variables are similar the principal component will be a good summary for both and the detail variable will have little energy. The concept of rotation can be observed in Figure 3.1 (a), where a scatter plot between two similar genes is shown, in red dashed lines the axes after the rotation are shown. In Figure 3.1 (b) the principal components

scatter plot is shown in the new space.

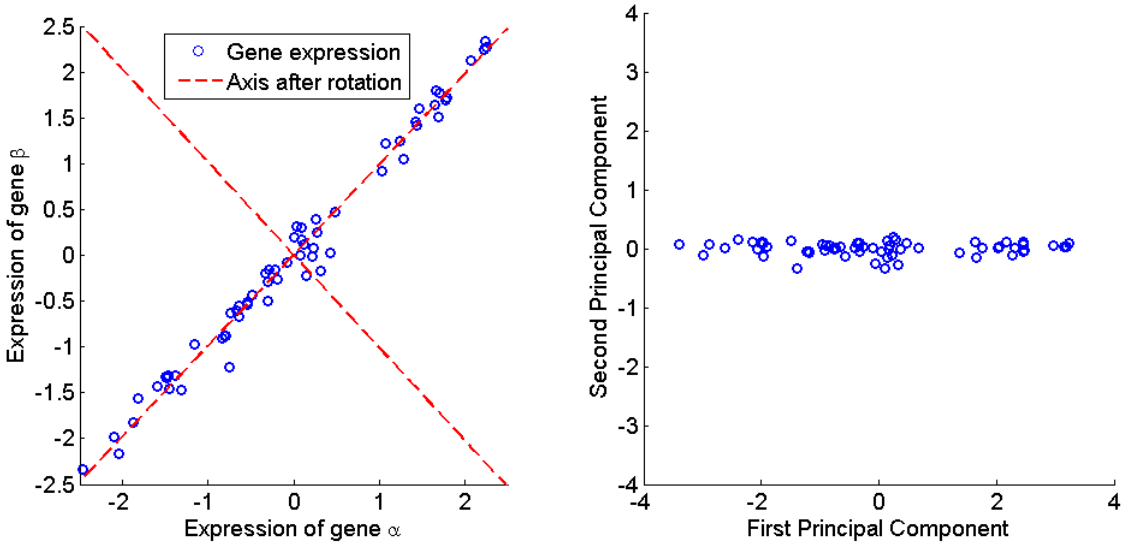


Figure 3.1. *(a) Scatter plot between two similar genes and new rotated axes. (b) Principal Components scatter.*

In this example the *approximation variable* is a linear combination between two genes, but growing up in the hierarchy these approximation variables are a combination between a gene and a previous approximation variable or between two approximation variables. So, the approximation variable would be a linear combination of N genes (see Equation 3.2). In the current work, this concept is named as *metagene*, which is a linear combination of N genes, not an actual gene.

$$\textbf{metagene}_L = \sum_{i=1}^{N} v_L[i] \cdot \textbf{gene}_i \tag{3.2}$$

3.2.2 L1 Normalization based on the child nodes

Using the previous method a metagene is computed as $\textbf{metagene} = \textbf{s} = \textbf{gene}_\alpha \cdot \cos\theta_L + \textbf{gene}_\beta \cdot \sin\theta_L$. This would leave an orthonormal basis with L2 norm due to the local pair-wise PCA, because the following equation is satisfied:

$$1 = \cos^2\theta_L + \sin^2\theta_L \tag{3.3}$$

This norm conserves the energy of the original signal; however the local continuous component of the original variables is not conserved by the *approximation variable*.

If the norm L2 is used (as in the PCA) then the approximation variable has a larger continuous component. Generalizing this example, and moving up in the hierarchy, the metagenes increase their differences with the individual genes they represent, and therefore cannot be compared with them.

In order to avoid this effect, the L1 norm, defined in Equation 3.4, does not conserve the energy but on the other hand preserves the continuous component, so when two identical gene clusters are merged the representative of them will be equal to both.

$$\begin{aligned} 1 &= \frac{\cos\theta_L + \sin\theta_L}{\gamma} \\ \gamma &= \cos\theta_L + \sin|\theta_L| \end{aligned} \tag{3.4}$$

Like in subsection 3.2.1, at each metagene formation the rotation that decorrelates the variables $\mathbf{gene}_\alpha$ and $\mathbf{gene}_\beta$ is calculated, but after that, this rotation is normalized as described by the Equation 3.4:

$$\begin{aligned} \mathbf{s} &= \mathbf{gene}_\alpha \frac{\cos\theta_L}{\cos\theta_L + \sin|\theta_L|} + \mathbf{gene}_\beta \frac{\sin\theta_L}{\cos\theta_L + \sin|\theta_L|} \\ \mathbf{d} &= \mathbf{gene}_\beta \frac{\cos\theta_L}{\cos\theta_L + \sin|\theta_L|} - \mathbf{gene}_\alpha \frac{\sin\theta_L}{\cos\theta_L + \sin|\theta_L|} \end{aligned} \tag{3.5}$$

θ_L is still the angle of rotation that decorrelates the two variables. Mathematically, this transformation is a rotation and a scaling. The angle θ_L for the local pair-wise rotation is defined as $|\theta_L| \leq \frac{\pi}{4}$. For this reason an absolute value is introduced in the sinus in Equations 3.4 and 3.5 in order to avoid a zero division. Consequently, this method to represent the genes clusters has been named as *L1 normalization based on the child nodes* because the normalization takes in account only the rotation at each metagene computation.

3.2.3 L1 Normalization based on the leaf nodes

As it has been explained in the previous section, a L1 norm is useful in order to preserve the continuous component in the representative of gene clusters. In contrast with the previous method, the L1 Normalization based on the leaf nodes leads to a basis with L1 norm taking into account the N genes that are being merged by the metagene, and not only the child nodes.

Therefore, this method relies in performing a PCA pair-wise transform as in the subsection 3.2.1 and at each level the approximation basis that involves all the N genes (see Equation 3.2) and the detail basis are computed and after, the whole basis is scaled to accomplish the

Equation 3.4, as could be seen in the next equation:

$$\begin{aligned} \mathbf{s} &= \sum_{i}^{N} w_L[i]\mathbf{gene}_i \\ w_L &= \frac{v_L}{\sum_i^N |v_L[i]|} \end{aligned} \tag{3.6}$$

These various normalization strategies will be evaluated in the experimental section.

3.3 Similarity Criteria

As it has been mentioned, to build the dendrogram, a similarity criterion must be defined. It should capture the concept of distance or similarity of two variables. This criterion, which could be a metric or a distance function, determines which gene clusters should be merged at each level of the algorithm.

3.3.1 Correlation and absolute value of correlation

One of the classical similarity measures in signal processing relies on the notion of correlation. The correlation criteria (M_{ij}) as a similarity distance between two variables (here cluster of genes) $\mathbf{gene}_i$ and $\mathbf{gene}_j$ is defined in Equation 3.7:

$$M_{ij} = \frac{\Sigma_{ij}}{\sqrt{\Sigma_{ii} \cdot \Sigma_{jj}}} \tag{3.7}$$

Where $\Sigma_{ij} = \mathbb{E}\{(\mathbf{gene}_i - \mathbb{E}\{\mathbf{gene}_i\})(\mathbf{gene}_j - \mathbb{E}\{\mathbf{gene}_j\})\}$ is the usual covariance. With this coefficient, the difference with respect to the shape of the curves representing the gene expressions for all patients are measured. So, the two gene clusters with the most similar shape are clustered together. Also, due to the normalization between autocorrelation of $\mathbf{gene}_i$ and $\mathbf{gene}_j$ variables, this measure does not penalize any difference in dynamic range, and only considers the patterns between the genes. It has been observed that in some RNA microarray data about 45 % of the genes present a negative correlation criterion in Equation 3.7, which means that these genes present a similar pattern if we invert one of them. In order to take advantage of this situation we have analyzed another criterion that would merge the two gene clusters with the maximum absolute value of correlation criteria ($|M_{ij}|$).

3.3.2 Squared error, Euclidean distance

The Euclidean distance or squared error is the ordinary distance in mathematics between two points and is given by the Pythagorean formula. Expanding this concept to multidimensional vectors, this distance between two vectors is computed as the squared root of the sum of the squared differences over all coordinates as seen in the next equation:

$$\begin{aligned} d_{ij}^2 &= (\mathbf{gene}_i - \mathbf{gene}_j)^T(\mathbf{gene}_i - \mathbf{gene}_j) \\ d_{ij}^2 &= \sum_{k=1}^{N}(\mathbf{gene}_i[k] - \mathbf{gene}_j[k])^2 \end{aligned} \tag{3.8}$$

In other words, the Euclidean distance measures the average difference across all the coordinates of $\mathbf{gene}_i$ and $\mathbf{gene}_j$ variables. This distance is used in order to place progressively bigger weights on vectors that are further apart. Therefore, in contrast to the correlation criteria, two gene clusters with a different continuous component will have a big distance (and would be merged in the lasts levels of the dendrogram). In order to use this criterion, the L1 norm is needed to represent the clusters. If a L2 norm is used, three identical genes could not be merged due to the continuous component that is not preserved in the representation in the first clustering. The use of this method is motivated by the belief that not only the shape of the genes provides information; also, differences in the continuous component of the gene expression could offer some useful information.

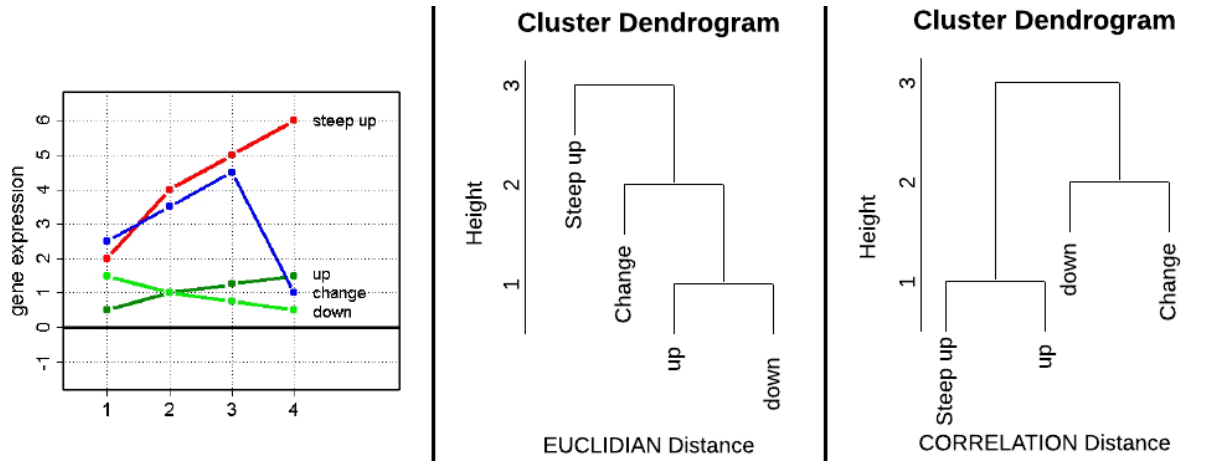


Figure 3.2. *(Left) Four different examples of gene expressions in time evolution, with different patterns and dynamic range. (Middle) Result of clustering of the four genes using the Euclidean distance as the distance criterion. (Right) Result of clustering of the four genes using the correlation distance as the distance criterion. These examples have been taken from [15].*

In Figure 3.2, the repercussion of using either the covariance or the Euclidean distance as a criterion is illustrated when constructing the dendrogram between the four synthetic genes shown on the left of the Figure 3.2, these examples have been taken from [15]. As it has been mentioned, using the Euclidean distance implies to merge first the genes with a similar shape and a similar continuous component. This phenomenon could be observed in the middle of Figure 3.2 where the genes *up* and *down* are merged in the first level. However, if the correlation distance is used as a similarity criterion, the first clustering would be between the genes *steep up* and *up* because they have the most similar pattern.

3.3.3 Classification error as a similarity criterion

In the two previous subsections, two criteria merging the two nearest gene clusters or the most similar ones have been presented, but this clustering is done independently of the classification task. Thus, combining genes may potentially reduce the effect of noise, but there is no guarantee that these clusters will lead to a better classification performance, because they are built from different but similar genes and represent them by a metagene (this step has been explained in the section 3.2).

Instead of merging similar genes, we have made an experiment trying to merge the pair of genes that would provide the best classification error. To this end, a monodimensional classifier has been used to evaluate the classification error resulting from the potential merging of all pairs of genes (or metagenes). The classification error was estimated with the bolstered resubstitution method [16, 17], as the criterion to build the hierarchical structure. The bolstered resubstitution method is a general method for error estimation that displays low variance and generally low bias as well. This method is based on *bolstering* the original empirical distribution of the data and generates a test set from the original data following simple rules. This method can be used to improve the performance of any error-counting estimation method as it does not imply any splitting of the sample set like in a cross-validation operation. This characteristic makes it very suitable for a small-sample context like the microarray analysis. It is a faster method than cross validation since it needs to build only one classifier instead of k in a k-Fold cross validation error estimation.

At each level of the tree the goal is to find the best two gene clusters that, after performing a pair-wise PCA between them, will achieve the best classification error in the approximation variable. If more than one pair of gene clusters give the same classification error, a *reliability* value is used to break the ties in terms of classification error. This metric gives a notion of the distance between the two classes and the threshold, and the dispersion inside of each class.

The biggest drawback of this method is that it needs more information from the dataset. In particular, this method uses the classification ground truth of the database to construct the hierarchical clustering (and not only the genes expression as with the other similarities). Moreover, the exhaustive search to find the best possible gene clustering at every level takes much more computation time; a further analysis on this point is performed in the appendix A. In average, this method takes 50 times more time as the others methods proposed in this work for performing the hierarchical clustering. So, in the chapter 5, it is only applied to the colon dataset, which is the smallest dataset.

3.4 Presentation of proposed hierarchical clustering algorithms

In order to illustrate the previous notions, a toy example is analyzed in this section. This example consists of 29 selected genes of the colon microarray dataset, more details of this dataset are presented in chapter 5. These genes were chosen so as the whole dataset is represented: 14 genes provide a good individually classification error when used individually and are also similar to each other, and the other 15 genes are very noisy genes with the worst classification errors. This dataset is shown in the Figure 3.3.

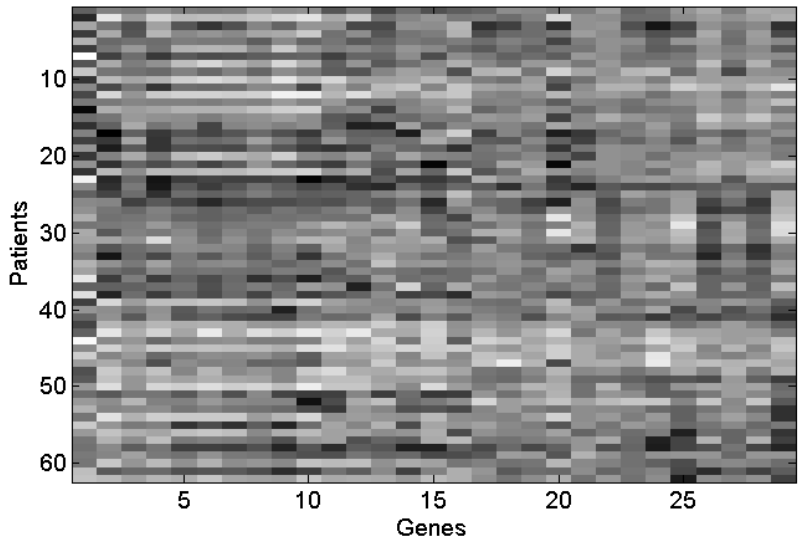


Figure 3.3. *Toy dataset of 29 genes of colon microarray.*

In order to illustrate and compare the different methods for hierarchical clustering, the following conventions are used. The dendrogram figures consist of many U-shaped lines connecting objects in a hierarchical tree. The height of each U represents the level of the tree, that is the iteration at which the cluster has been created. Each individual gene has a color that represents a subgroup inside of 29 selected genes: cyan and green ones are the best genes for classifying in one dimension, and the components of each group are similar between them; black and red ones are two subgroups with similarities between them selected from the noisiest genes with the worst classification error. The numbers appearing close to each node of the dendrogram represent the classification error of a mono dimensional classifier. In order to identify easily the best node for classifying, the best result is written in red.

As explained in the previous section, depending on how to define similarity between genes or metagenes and how to represent a cluster, different hierarchical clustering algorithms are possible, in the present work we have studied the following ones:

1. PCA based on the two child nodes and correlation value as similarity measure. These conditions for hierarchical clustering are proposed in Treelets [4], and was the starting point of the current work.
2. PCA based on the two child nodes and absolute value of correlation as similarity measure ("Anticorrelation Treelet").
3. L1 Normalization based on the child nodes and Euclidean distance as similarity measure ("L1 child nodes ").
4. L1 Normalization based on the leaf nodes and Euclidean distance as similarity measure ("L1 leaf nodes").
5. PCA based on the two child nodes and classification error as criterion to merge genes ("Classification clustering").

3.4.1 Treelet

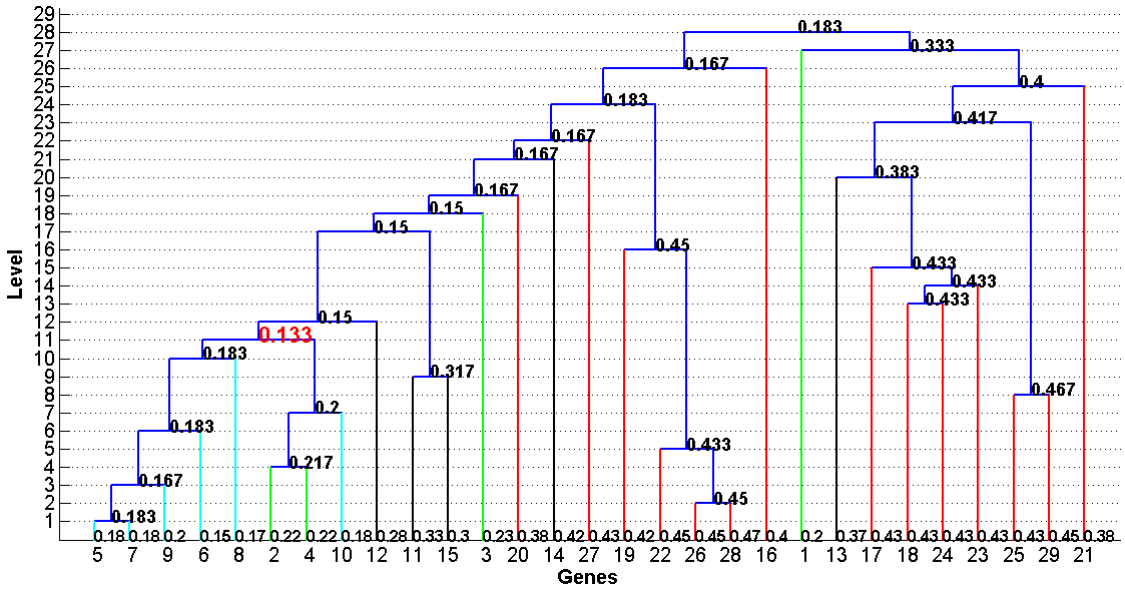


Figure 3.4. *Dendrogram corresponding to the Treelet algorithm (Cluster representation: Pairwise PCA and Similarity: correlation).*

In the Figure 3.4, it can be observed that the dendrogram merges some of the best genes in the first eleven levels. The level eleven produces the best metagene for classification (error 13%) and the approximation basis at this level is shown in the Figure 3.5. The figure shows the actual weights used to compute the linear combination creating the metagene. Here we can see that the metagene at level 11 involve a combination of 8 genes. It can be also observed that gene number one which has a good classification error (20%) has been merged with the worst genes and belongs to the right subtree. The reason is because this gene presents a negative correlation compared with the majority of other genes.

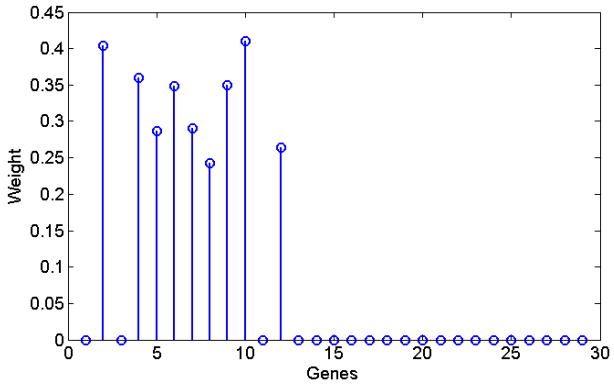


Figure 3.5. *Basis of Treelet at level 11.*

3.4.2 PCA based on the two child nodes and absolute value of correlation as similarity measure

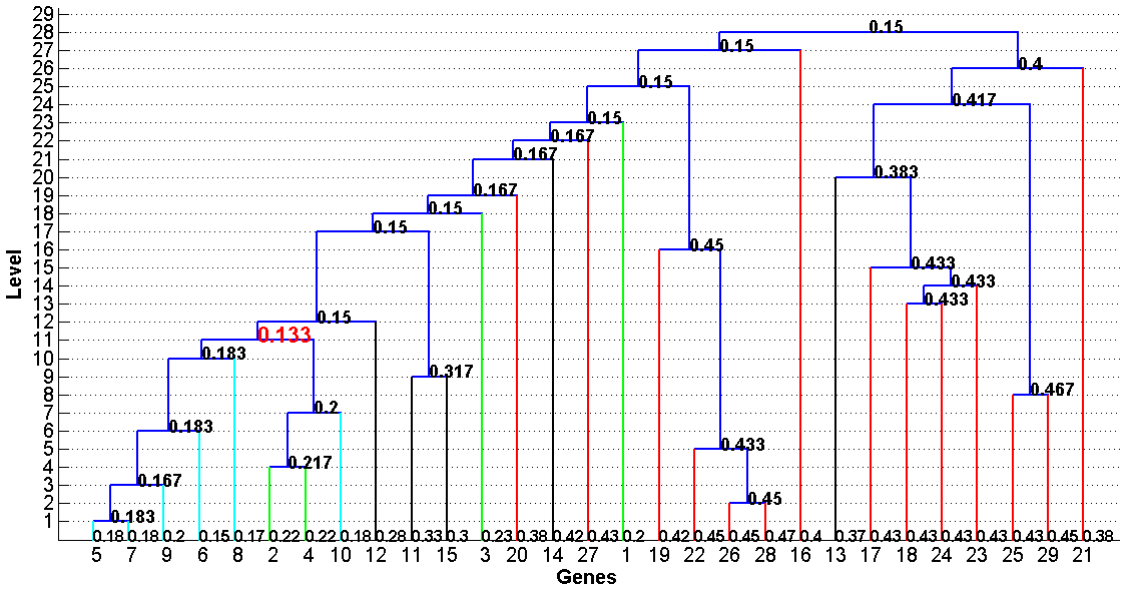


Figure 3.6. *Dendrogram resulting from the following conditions: Cluster representation: Pairwise PCA and Similarity: absolute value of correlation.*

In the Figure 3.6, it could be observed that the dendrogram has many elements in common with the one obtained with Treelet, and, in both, the level eleven generates the best metagene for classifying and the approximation basis at this level is the same and, for this reason it is not showed again. But in this case, the gene one has been merged with the left subtree because the similarity criterion considers that two genes with negative correlation (for example, if one is the opposite of the other) are very similar.

3.4.3 L1 Normalization based on the child nodes and Euclidean distance as similarity measure

With this method, the same classification error (13%) is obtained at two levels, the first one (level 12) involves fewer genes in the cluster, and the second one (level 28) involves all the genes. This result at level 28 is caused by the L1 normalization based on the child nodes leads to a basis, where in most cases; the last element merged to the cluster has a bigger weight in the basis, so the metagene obtained is more related with it. This phenomenon could be observed in Figure 3.8 (b) where the last gene merged (gene one) has one of the biggest weights in the basis (in terms of absolute value).

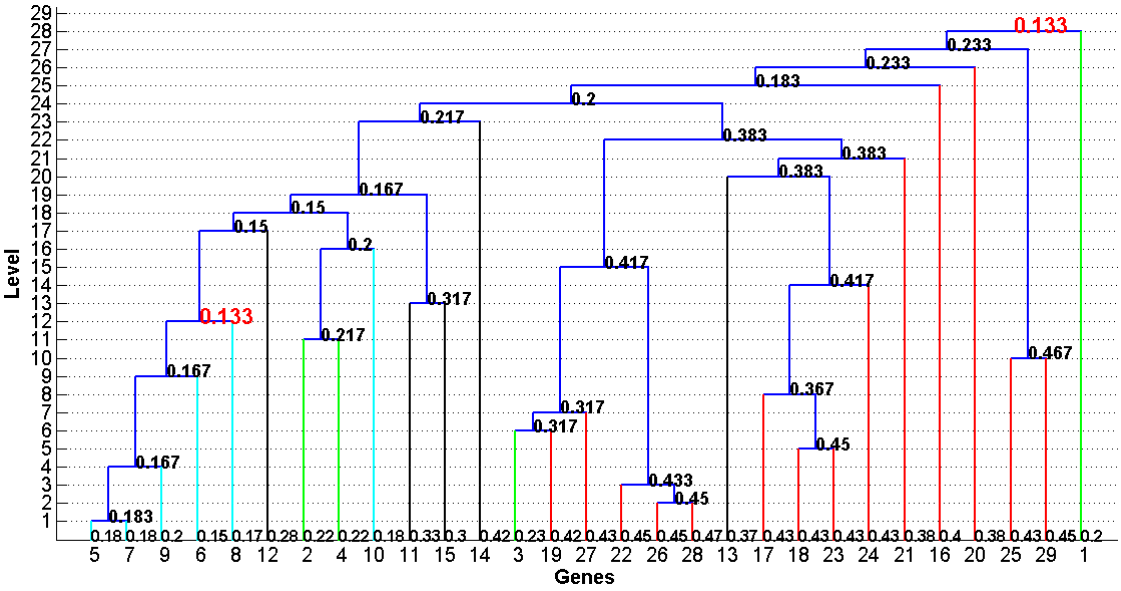
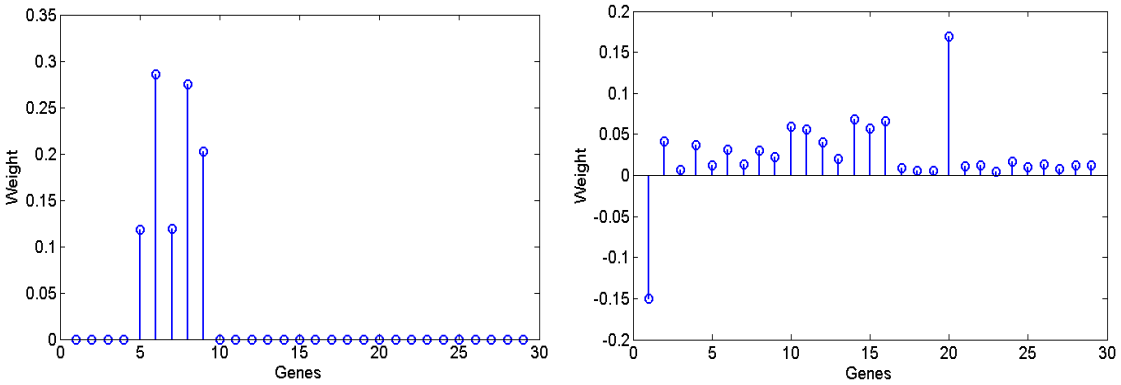


Figure 3.7. *Dendrogram resulting from the following conditions: Cluster representation: Pairwise PCA with L1 normalization based on the child nodes and Similarity: Euclidean distance.*

As has been explained, the most similar genes are merged in the first levels, so if several genes are grouped together the last incorporated one is the most different between them. So this effect could be not desirable, but the microarrays analyzed at chapter 5, shows a good behavior in term of outcomes.



(a) Level 12 **(b)** Level 28

Figure 3.8. *Approximation basis of the hierarchical clustering.*

3.4.4 L1 Normalization based on the leaf nodes and Euclidean distance as similarity measure

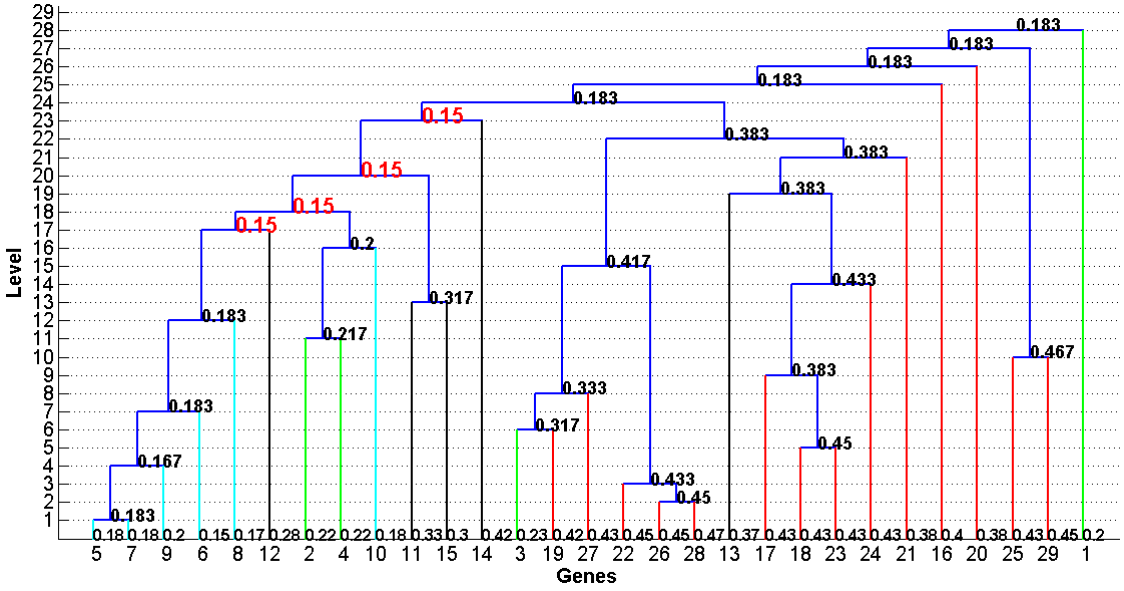
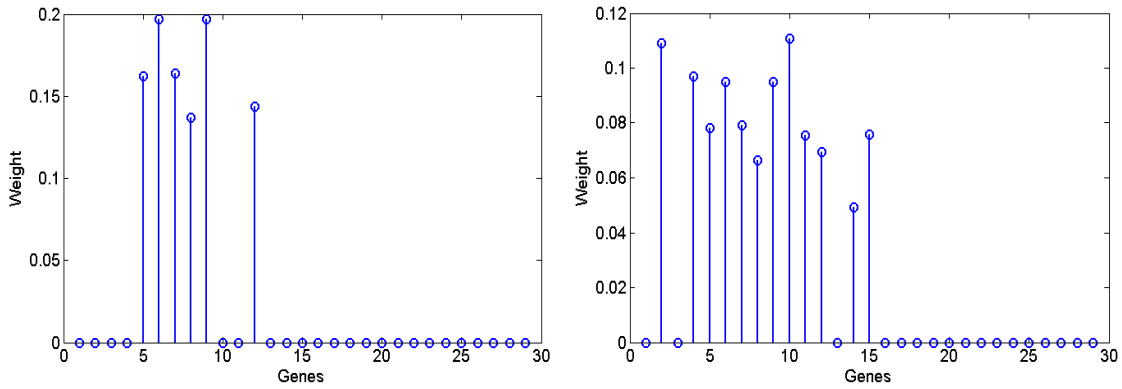


Figure 3.9. *Dendrogram resulting from the following conditions: Cluster representation: Pairwise PCA with L1 normalization of leaves and Similarity: Euclidean distance.*

With this hierarchical clustering algorithm, four metagenes get the best and the same classification error (15%); however it is worse than the previous methods. In the Figure 3.10 the approximation basis of level 17 (a) and level 23 (b) are showed. It could be observed that, in this case, the approximation basis does not assign a very large weight to the last clustered gene because the normalization is performed on the leaves.



(a) Level 17 **(b)** Level 23

Figure 3.10. *Approximation basis of the hierarchical clustering, Pairwise PCA with L1 normalization of leaves and Similarity: Euclidean distance.*

3.4.5 PCA based on the two child nodes and classification error as criterion to merge genes

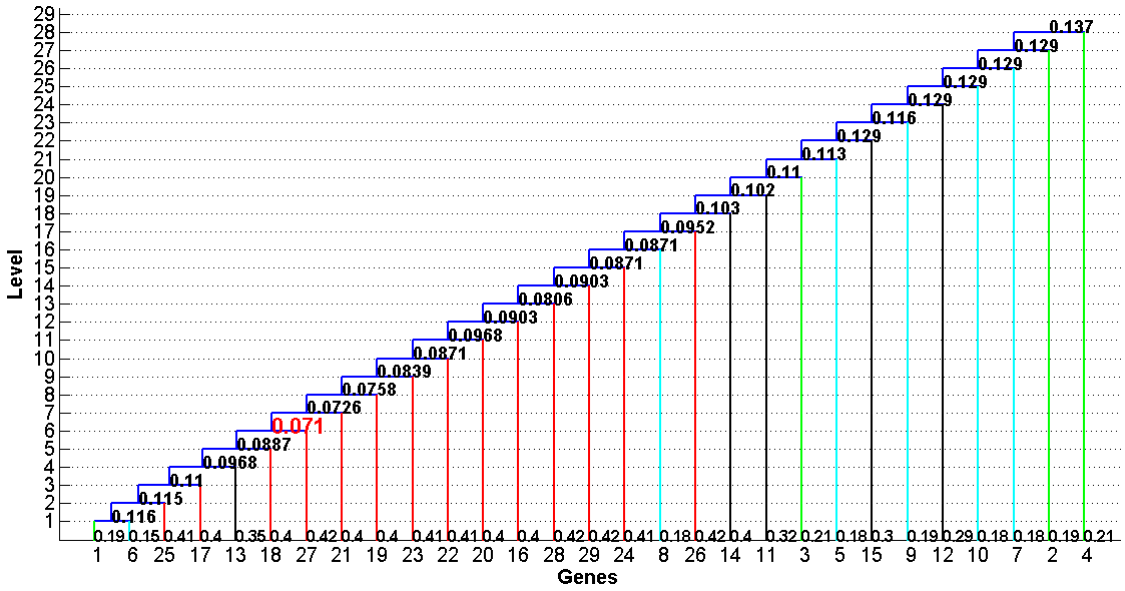


Figure 3.11. *Dendogram resulting from the following conditions: Cluster representation: Pairwise PCA and Similarity: Lowest classification error.*

This hierarchical clustering gets the best classification error (7,1 %), at level 6, of all of the methods presented in this section. This method tends to generate dendrograms with a stair shape (individual genes are progressively merged to the metagene). It is thought that this behavior is caused by the fact that the first metagene was obtained searching all the possible clustering between individual genes and it is the best clustering with the original genes. So, the next levels only add some genes to this representative that eliminate the noise and separate more the two classes. This phenomenon could be observed in the next figure where most of elements of the basis have a small weight.

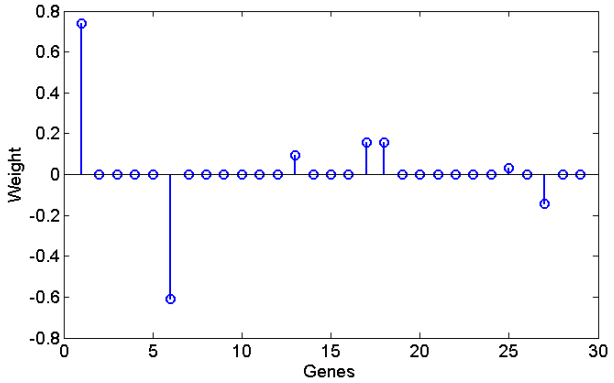


Figure 3.12. *Metagene basis of level 6.*

Unlike the previous methods which use only the dataset of gene expression for training, this method uses extra information as the classification ground truth. This clustering always reaches the best classification error and very high *reliability* value applying it to the ground truth. This happens because it has been built to achieve this goal. Therefore, in contrast to the previous methods, this hierarchical clustering cannot be applied to the same set for which has been built. The method to perform the building and the validation is explained at chapter 4. Nevertheless, in this example it is applied to the same set because it is only for exemplifying the differences between the methods.

Chapter 4

Introduction to LDA classifier to evaluate the hierarchical clustering algorithms

As mentioned in the chapter 1, this PFC is included in a bigger research project. Thus, the method presented in this chapter has been developed by Mattia Bosio (who is a PhD student of the TSC) in order to perform microarray data classification. The goal of his work is to find a suitable algorithm to identify the best subset, in terms of predictive ability, from an initial feature set.

In Figure 4.1, the procedure flowchart of the whole project is shown. As can be seen, the original microarray data matrix $S_{n,p}$ is the input data composed by probe set values in logaritmic scale. As mentioned in chapter 2, the data matrix has a structure n x p in which rows represent the patients and each one of the columns represents a gene profile. The microarray data matrix is the input data for one of the hierarchical clustering algorithms explained in chapter 3, where it is transformed into the feature set matrix $S^*_{n,2p-1}$. This procedure generates an agglomerative combination of genes with a tree structure, so, $p-1$ metagenes are generated in this step.

After that, the core block of the feature selection process scans the whole feature set, composed by the union of both metagenes and original genes, to choose the best subset in terms of error rate and reliability.

Figure 4.1. *Procedure flowchart of the whole project. The white blocks represent external proceedings that have not been developed in this PFC project.*

4.1 Statistical classification

Statistical classification is the problem of identifying the sub-population to which new observations belong. To perform this task the identity characteristics of each sub-population on the training set of data should be ascertained. In the training set the sub-population belonging of each sample is known. Thus the requirement is that new individual items are placed into groups based on quantitative information on one or more measurements, traits or characteristics, etc. and based on the training set in which previously decided groupings are already established.

4.1.1 Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) is a statistical technique for classifying samples into mutually exclusive groups and is based on a linear combination of a feature set.

The approach assumes that a training set is available, that is, for each sample x of the training set the corresponding membership y is known. In order to simplify the explanation it is assumed that only two sub-classes exist. These two classes of observations with N_0 and N_1 sample sizes, which have means $\mu_{y=0}$ and $\mu_{y=1}$ and the following covariances $\Sigma_{y=0}$ and $\Sigma_{y=1}$.

Then, applying a linear combination w on the feature set, will produce a change on the mean of each class $w^T\mu_{y=i}$ and their variance $w^T\Sigma_{y=i}w$. The separation between these two distributions is defined as the ratio of the variance between the classes (inter-class) with the variance within the classes (intra-class), as can be seen in Equation 4.1.

$$S = \frac{\sigma^2_{inter}}{\sigma^2_{intra}} = \frac{(w^T\mu_{y=1} - w^T\mu_{y=0})^2}{w^T N_1 \Sigma_{y=1} w + w^T N_0 \Sigma_{y=0} w} \tag{4.1}$$

LDA assumes that posteriori class probabilities are normally distributed and have the same covariance matrix, in order to simplify the classification operations. LDA attempts to recover the direction along which data are best discriminated. It is done trying to simultaneously maximize the inter-class distance and minimize the intra-class distance, maximizing S of Equation 4.1 with respect to w.

The choice of this classifier follows the idea of simplicity highligthed in [18] and [17]. In this kind of microarray analysis problems, very complex classifiers with many parameters usually lead to over-fitting and poor generalization characteristics. For this reason a fairly simple classifier has been chosen as it should lead to more reliable results.

4.2 Feature selection

In this section, a brief explanation of the feature selection block in Figure 4.1 is presented. The objective is of this block is to select from the expanded feature set (initial genes and metagenes obtained through hierarchical clustering) the smallest subset to build a good classifier. Let us first discuss the criterion used to perform the feature selection.

4.2.1 Feature ranking criterion

Cross Validation

The error estimation is performed through a 10 fold cross validation phase (CV), which is an iterative validation process performed in 10 iterations. In each iteration, the first step is to split the sample set, the n samples are divided in two complementary sets: the training set, containing approximately 90% of the samples, and the test set, containing approximately 10% of the samples. Both sets are selected in order to be balanced, they have samples from both classes, and also representative, with similar proportions as in the original dataset.

1. The i^{th} feature of validation set and training set is respectively selected.
2. Training set data are used in order to train the classifier whose performance is evaluated using test set data.
3. Test set data are classified using the classifier produced. The outputs are two values, error rate end *reliability*, relative to ith feature in k^{th} iteration.
4. These values are averaged along iterations to obtain mean values of error rate and *reliability* for each one of the features. The final output is the best feature out of the

ranked feature list.

Ranking protocol

It is common to have a group of features with the same error rate when only one feature has to be selected. To solve this situation, a two level criterion is introduced. The ranking protocol to select the best feature consists in two steps:

1. Features are first sorted in terms of error rate from smallest to largest.
2. Features with equal error rate are then sorted from largest to smallest reliability parameter.

The reliability metric gives a notion of the distance between the two classes and the threshold, and the dispersion inside of each class.

4.3 Feature selection algorithm

The algorithm is a modification of the Sequential Floating Forward Selection algorithm (SFFS) [19] with the introduction of a replacing step when backtracking does not work. It is called Improved sequential Floating Forward Selection (IFFS) [20].

SFFS is a sequential algorithm that allows backtracking after each sequential step to locate a better subset: after adding a feature to the subset, the algorithm looks for the possible benefits of eliminating one or more features.

IFFS adds more flexibility to SFFS introducing a replacing stage in case that backtracking does not improve the classification performance. The price to pay is a sensible increase of execution time in the replacing phase that does not grow linearly with the feature subset dimension. In Figure 4.2, the flowchart of IFFS technique is presented.

Starting with an empty set and ending when a threshold value is reached. The threshold is either the maximum accepted number of features or a maximum number of iterations in order to avoid an infinite loop of the algorithm. The steps of the algorithm are the following:

1. The add phase, for each features that have not been tested, the current feature set is expanded by adding it.
   (a) The classifier is trained and then validated producing a classification score $J(\cdot)$.
   (b) The feature whose classification score is the best is selected to be added to the current set.
2. Backtracking phase, which is done if the threshold has not been reached.
   (a) The weakest feature in the current subset is marked as candidate to be eliminated.
   (b) If the elimination improves the classification score, the weak feature is removed and a new backtracking phase is performed
3. Substitution phase, the algorithm tries to substitute one feature in the current set, the substitute is chosen via a similar analysis at the add phase.
   (a) If some substitution improves the classification score, this feature is marked.
   (b) If more than one substitution are possible, only the feature which achieves the best improvement in $J(\cdot)$ is selected.
   (c) The current set is updated and a new backtracking phase is performed.

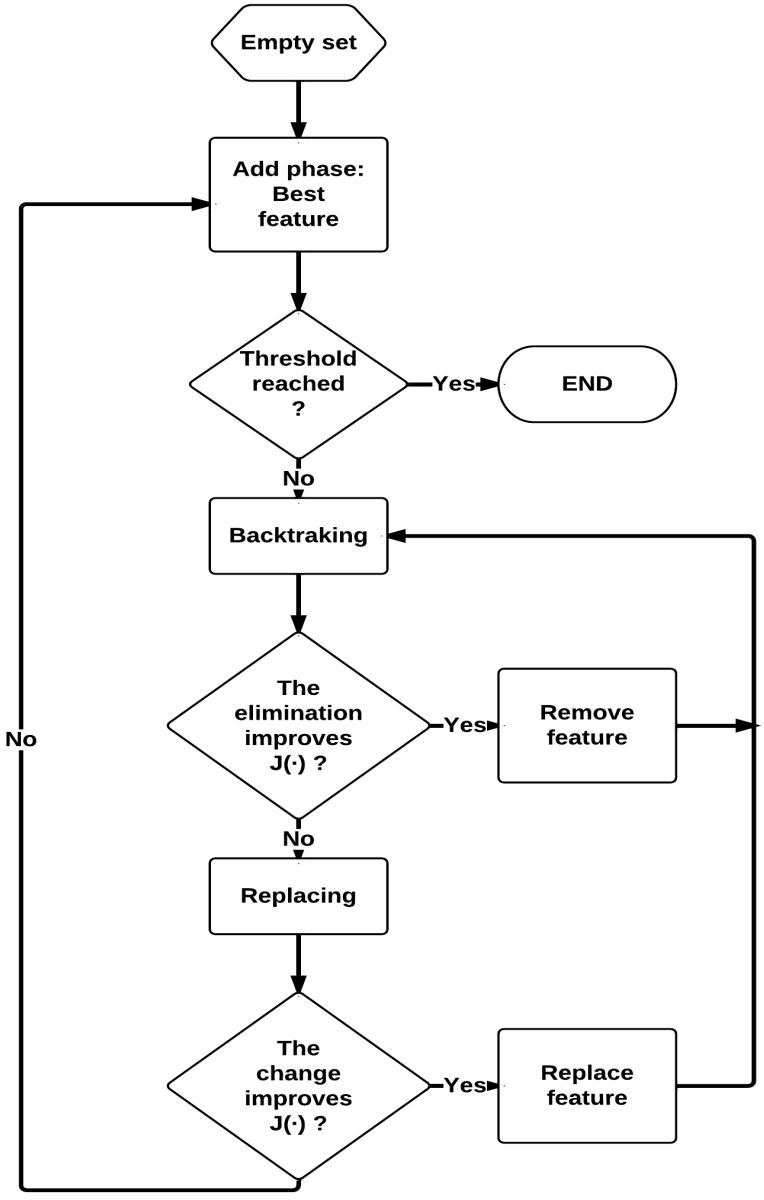


Figure 4.2. IFFS feature selection algorithm.

Chapter 5

Results of feature enhancement via hierarchical clustering algorithms for cancer classification

In this chapter, the hierarchical clustering algorithms have been used to classify different microarray datasets. The goal of the experiments is to analize whether the enrichment of the microarrays via the introduction of the metagenes improves the classification performance of microarray datasets. The metagenes generation process will be considered useful if it allows classifying better, or with fewer features, than using only the original probe set values. Also, the obtained results with the proposed method are compared with the *state of the art* .

Organization of results

The results are presented in three sections, each one corresponding to a different cancer dataset. In each dataset a monodimensional and multi-dimensional classification analysis is performed. The monodimensional results involve the expression level of the best metagene for classifying, and also the approximation basis to generate this metagene. The best method for classifying is the one that achieves the smallest error rate and the highest reliability. The results are presented in tables including error rate and reliability values for each method of enrichment.

5.1 Selected microarray datasets

Experiments have been conducted on three different public datasets which characteristics are explained in the following list and are summarized at table 5.1.

- Colon Dataset: it is a Colon cancer dataset consisting of 62 patients (40 with Colon cancer and 22 without) of 2000 genes each one [21]. This is a commonly used dataset in the literature to evaluate classification algorithms and can be downloaded at http://genomics-pubs.princeton.edu/oncology/.
- Leukemia Dataset: it is an acute Leukemia dataset [8] and it is very common dataset in the literature. It consists in 72 patients (47 with acute lymphoblastic Leukemia ("ALL") and 25 with acute myeloblastic Leukemia ("AML")) each having 7129 probe sets. In this dataset the training set and the validation set are already defined and consists respectively in 38 samples (27 ALL and 11 AML) and 34 samples (20 ALL and 14 AML). It can be downloaded from http://www.broadinstitute.org/cgi-bin/cancer/publications/pub_paper.cgi?mode=view&paper_id=43.
- Lymphoma Dataset: is a collection of expression measurements from 96 normal and malignant lymphocyte samples [22]. It contains 42 samples of diffused large B-cell Lymphoma (DLBCL) and 54 samples of other types, each one including 4026 genes. And it is available at http://llmpp.nih.gov/lymphoma/data/figure1.cdt.

Dataset Name	**Colon**	**Leukemia**	**Lymphoma**
Number of Genes	2000	7129	4026
Total of Patients	62	72	96
Number of classes	Cancer/ No Cancer	ALL/ AML	DLBCL/ No DLBCL
Class 1 patients	40 (Cancer)	47 (ALL)	42 (DLBCL)
Class 2 patients	22 (No Cancer)	25 (AML)	54 (No DLBCL)

Table 5.1. *Table with the main characteristics of the used Datasets.*

5.2 Colon dataset

This dataset has already been filtered, preserving the 2000 genes with highest mean intensity value from an original set of 6500 genes. These 2000 genes are processed to generate 1999 metagenes, with the different hierarchical clustering methods (see chapter 3).

The feature selection phase is performed through a 10 fold cross validation. The two complementary sets (training and test) reflects the original dataset, and the test set is made of six samples composed by two samples of one class and four of the other. This test only involves 60 elements, therefore two samples are not evaluated in the cross validation process.

Since this database only involves 2000 genes, it has been possible to apply all the hierachical clustering algorithms proposed in the chapter 3. However, the application of anticorrelation clustering has revealed not to be useful. This is, the produced metagene set is almost equal to the one produced by original Treelet, except in the last two levels when all genes are combined. Thus, the anticorrelation treelet has been removed from the analysis.

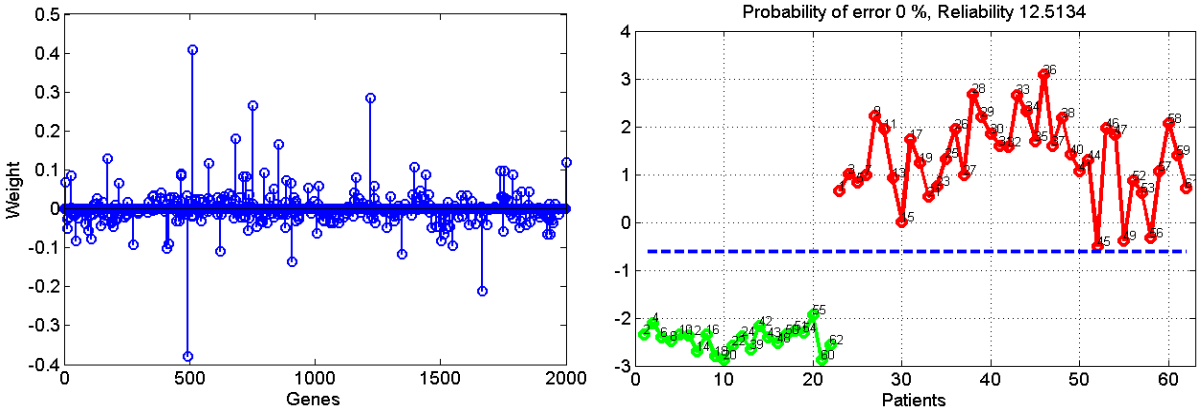
5.2.1 Monodimensional classification analysis

In this subsection, the monodimensional classification, with a 10 fold cross validation (CV) in order to estimate the error, is compared along the different clustering methods and the original data (without enrichering via the metagenes). The analysis consists in performing an exhaustive monodimensional classification in all the original genes and metagenes computed with the following clustering methods: Treelet, L1 child nodes, L1 leaf nodes and the classification clustering. In the figures of this section, the gene or metagenes expressions for all the patients are shown: patients with (without) colon cancer are shown in red (green), the blue line represents the mean threshold of the classifier computed via 10 fold CV.

	Original Data	**Treelet**	**L1 child nodes**	**L1 leaf nodes**	**Classification clustering**
Error %	15	13	15	15	0
Reliability	2.37	2.47	2.37	2.37	12.51

Table 5.2. *Monodimensional results (error % and reliability) for Colon dataset and the different hierarchical clustering methods.*

In Table 5.2, the classification outcomes are compared. With this database the best monodimensional classification is performed by the classification clustering, because this method uses extra information to learn from a ground truth of classified patients. In Figure 5.1b the best metagene is shown, and as can be observed it allows to get 0 % classification error and very high reliability value. This result is not surprising, because the building of the classification clustering allows to search the best combination to classify a ground truth data. This combination involves 299 genes in the approximation basis (see Figure 5.1a), but only 15 of these genes present a significant weight in the basis. Thus, this method must be tested to prove if good results are maintained when this hierarchical clustering is applied to a different set for which has not been trained. To test it, a new protocol will be defined later in this section (see subsection 5.2.3).

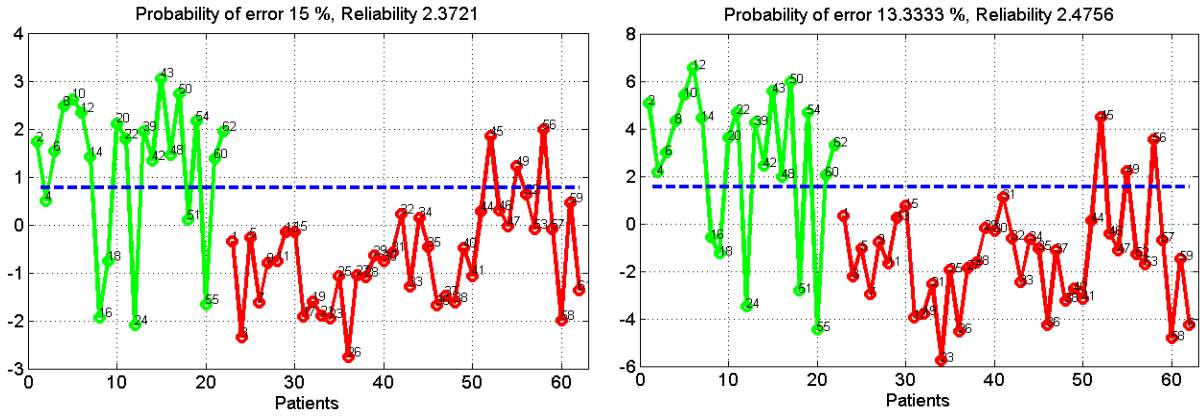


(a) Approximation basis to obtain the metagene with classification clustering method.

(b) Metagene obtained with classification clustering method.

Figure 5.1. *Metagene obtained with classification clustering method and its approximation basis.*

In Table 5.2 could be seen that the two L1 hierarchical clustering methods do not help to improve the monodimensional classification. In fact, the presented results are the same as the one of the Original genes column, that is the best result is obtained with an initial gene (not a metagene) that is shown in Figure 5.2a. On the other hand the original Treelet algorithm achieve a small improvement in the classification task, the metagene generated by Treelet algorithm is presented in Figure 5.2b. In both cases it can be observed that the classification is not perfect, because some samples of the same class are above and below of the threshold.



(a) Best individual gene for monodimensional classification using L1 norm.

(b) Metagene obtained with original Treelet algorithm.

Figure 5.2. *Best monodimensional gene and metagene for classification using L1 norm and Treelet as hierarchical clustring algorithms.*

5.2.2 Multi-dimensional classification analysis

Now, the multi-dimensional classification is analyzed and the results are shown in Table 5.3, in order to prove the benefits of introducing a feature set expansion by the inclusion of metagenes from different hierarchical clustering algorithms. The best method is still the classification clustering, but also it can be observed that the inclusion of the metagenes allows to reach a zero error using fewer dimensions than using the original set only.

In general term it can be observed that when the dimensionalty grows, better results are obtained. There is an improvement in the reliability and a reduction of the classification error % in all cases, but there are important differences between the clustering algorithms. With the orginal dataset and Treelet enrichment, the IFFS algorithm needs 6 dimensions to achive the zero error. In the case of L1 norm this goal is accomplished with 5 dimensions. And, in the classification clustering method the IFFS achieves a huge reliabilty with the increase of the dimensionality.

	Original Data		Treelet		L1 child nodes		L1 leaf nodes		Classification clustering	
N	Error %	Reliability	Error %	Reliability	Error %	Reliability	Error %	Reliability	Error %	Reliability
1	15	2.37	13	2.47	15	2.37	15	2.37	0	12.51
2	8.3	4.22	8.3	4.83	8.3	4.46	8.3	4.46	0	15.39
3	6.3	4.81	6.3	4.92	5.0	4.97	5.0	4.97	0	16.65
4	5.0	5.97	3.3	5.29	3.3	6.192	3.3	6.192	0	17.89
5	3.3	6.68	1.5	5.78	0	6.78	0	6.78	0	17.11
6	0	6.57	0	6.95	0	8.26	0	6.31	0	18.42
7	0	7.54	0	7.15	0	9.24	0	7.73	0	18.95
8	0	8.68	0	7.91	0	10.27	0	8.55	0	19.86
9	0	11.22	0	8.79	0	12.21	0	9.64	0	20.79
10	0	12.84	0	10.06	0	13.95	0	10.77	0	22.3

Table 5.3. *Multi-dimensional results for Colon dataset using IFFS as feature selection method, and the different techniques to generate metagenes.*

5.2.3 New experimental protocol to evaluate the data knowledge effect

As mentioned in Chapter 3, the classification clustering takes into account the class information to decide which features are clustered at each level. This fact has motivated an experiment with a different protocol to evaluate if this result derives from an excessive knowledge of the data.

This hierarchical clustering cannot be applied to the same set that it has been built. Thus, in order to compare this hierarchical clustering with the others methods it must be applied to a different set, so we use a cross-validation method to build and test it. The patients are split in 10 groups, in this case, inside of each group there are four patients of one class and two of the other. The hierarchical clustering algorithm is aplied only to 9 of these groups that conforms the training set. Then, the best features in terms classification are searched perfoming IFFS algorithm only in the data from the training set. After that, this classifier is applied to the other set (test set) in order to obtain an blind classification error and reliability coefficient, this results are named as validation results. Like in the standard cross-validation, in order to reduce variability, multiple rounds are performed switching between the different sets, and then, the validation results are averaged over the rounds. So with this protocol 10 different metagenes collection are obtained for each hierarchical clustering algorithm.

This experiment simulates the application of a trained classifier on an independent set for ten times. It has been adopted only for a comparison purpose to estimate how the benefit of using classification clustering is related to the sample class knowledge. This kind of experiment, as remarked in [17], is not the best procedure for error estimation when the number of sample is

small, but the results could be useful to evaluate if the classification clustering is a generalizable methodology to classify independent data.

The obtained results of the monodimensional analysis are showed in Table 5.4. They are obtained for all the clustering methods, by an exhaustive search of the best gene or metagene for classifying in the training set. Then, it is applied to perform a blind classification into the test set, and this analysis is repeated in the 10 different iterations involving all the patients in the different roles (training or test).

	Original Data	**Treelet**	**L1 child nodes**	**L1 leaf nodes**	**Classification clustering**
Error (%) on Test set	21.7	20	25	26.7	18.3
Error (%) at Training set	14.3	13.9	13.8	14.1	0

Table 5.4. *Error % results for 10 CV building mondodimensional classification with different metagen agregation procedures.*

As it has been explained, the previous errors estimations shown on the first raw only take into account the 10 results of blind classification and an average between their parameters is performed. It can be observed that the Treelet, L1 child nodes and L1 leaf nodes algorithms obtain in mean, worst classification errors on the test set that the ones shown in Table 5.2. It can be seen that the two hierarchical clustering with L1 norm have a worse results on the test set than the one obtained with the original genes. But, it is thought that these results are caused by the random factor and the scarcity of the patients involved at both sets, because their results in the training set are always better. The classification clustering achieves the best classification error, and it is perfect for the training set.

Otherwise, the multi-dimensional analysis has been performed in this experiment too, and the results are presented in Table 5.5. It can be observed how the classification clustering is not good when it is applied on an independent set. It is thought that these results are due to the reduced number of samples. And also it may be due to the fact that the feature selection process is performed without an internal cross validation phase (the features that best fit the training set are chosen regardless of their strength in classifying unknown samples).

About the error rates on the test set, no regularity can be observed in the error rate progression as the dimension number grows and this could be due to an overfit in the training data. However, the best error value is obtained using metagenes from L1 normalization on leaves with 10 features: it reaches 8.3% of error rate on the test set and a 0.2% on the training set. In the multi-dimensional analysis, the classification clustering does not present the best outcomes. Moreover, the error rate value is constant. It is thought, that this behavior is

due to the high level of adaptation that can be obtained using this clustering method. Many metagenes perfectly classify the training set. Indeed, they are all highly correlated with each other, due to the stair-pattern of the clustering.

	Original Data		Treelet		L1 child nodes		L1 leaf nodes		Classification clustering	
N	Error % Test	Error % Training	Error % Test	Error % Training	Error % Test	Error % Training	Error % Test	Error % Training	Error % Test	Error % Training
1	21.7	14.3	20.0	13.9	25	13.8	26.7	14.1	18.3	0.0
2	20.0	7.1	18.3	7.3	26.7	6.6	18.3	6.4	18.3	0.0
3	20.0	4.5	20.0	5.5	21.7	4.5	20.0	4.1	16.7	0.0
4	21.7	3.0	21.7	3.2	23.3	3.0	16.7	2.5	16.7	0.0
5	21.7	2.5	23.3	2.7	21.7	1.8	16.7	1.6	16.7	0.0
6	18.3	1.8	20.0	1.4	21.7	1.4	15.0	0.7	16.7	0.0
7	20.0	0.9	21.7	1.1	16.7	0.5	13.3	0.2	16.7	0.0
8	15.0	0.5	21.7	0.7	18.3	0.2	16.7	0.2	16.7	0.0
9	20.0	0.5	25.0	0.4	26.7	0.0	16.7	0.2	16.7	0.0
10	10.0	0.5	18.3	0.4	13.3	0.0	8.3	0.2	16.7	0.0

Table 5.5. *Multi-dimensional analysis for Colon dataset with new experimental protocol.*

The problem of the classification clustering is the high dependence of the classification accuracy from the dataset used in the construction phase, evidenced in Table 5.4 and Table 5.5. This limitation makes it less robust in the case of a reduced training set than other clustering methods that simply cluster using a distance measure not related to the sample class.

After this experiment it is difficult to reach a clear conclusion about the knowledge effect of the data. Some interesting results have been obtained and should be studied with a deeper analysis in order to learn from the data and at the same time avoiding a high dependence in the building set process. The classification clustering is probably the best solution if the training set is large enough and the number of dimension is low. But if the training set is very small, other clustering algorithms are probably better.

5.2.4 Comparison with the *state of the art*

For this dataset, the *state of the art* (see Table 5.6) proceeds from different techniques in feature processing, feature selection and classification, and are extracted from [23, 24, 25, 26]. All the values are relative to the test set classification after a cross validation phase. The comparison shows that our results are equal to the best methods in terms of error rate. Furthermore there is an improvement in terms of classifier dimension. So, expanding the feature set with metagenes generated by a clustering process is beneficial to the classification, by using less

dimensions and obtaining little error classification.

Algorithm	Error %	Number of dimensions
NSGAA II [26]	0	6
Genetic programming [23]	0	6
Top scoring pair [24]	5.4	2
Partial LS logistic discriminant [23]	6.5	–
Discrete wavelet transformation & Neural network [23]	8	–
ICA & LDA [24]	14.04	–
kPCA linear kernel + FDA [23]	19.7	–
PCA + Fisher Discriminant Analysis [23]	19.7	–
IFFS original data	0	6
IFFS L1 norm	0	5
Classification clustering	0	1

Table 5.6. *Comparision with* state of the art *for Colon dataset. Our methods are highlighted in a grey cell.*

5.3 Leukemia dataset

In order to compare our results with the *state of the art* the feature selection protocol has been changed to a 20 fold cross validation. Also, since the database already has two independent sets for training and testing the classifiers, they have been adopted.

The classification clustering technique has not been analyzed for this database due to computational cost (see Appendix A). Thus, it is not included in the results. Instead, the Treelet with anticorrelation method is analyzed. With this dataset, this method generates a metagene collection significantly different from the one generated by the original Treelet method.

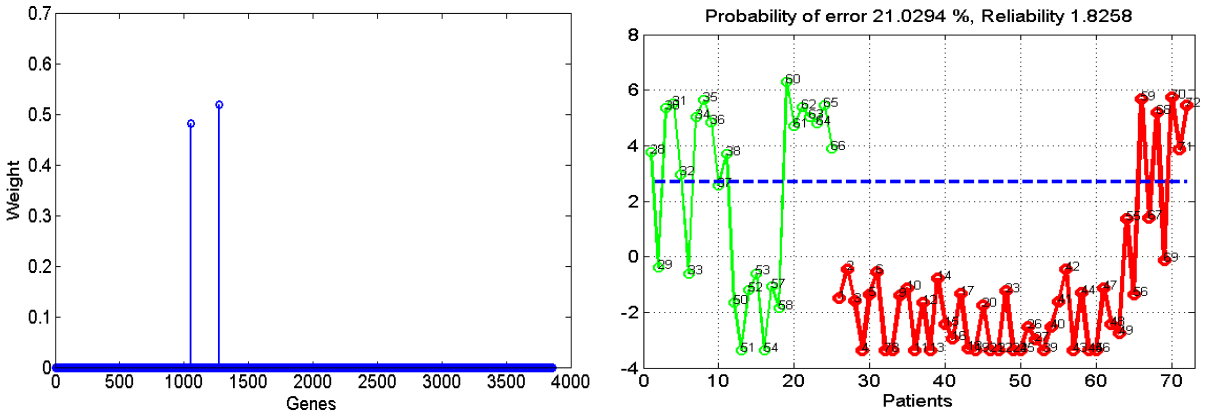
5.3.1 Monodimensional classification analysis

In this subsection, the monodimensional classification is analyzed. As can be observed in Table 5.7, with a monodimensional classifier the introduction of the metagenes does not improve the outcomes. This analysis consists in performing an exhaustive monodimensional classification with a 20 fold cross validation only in all the metagenes computed with the following clustering methods: Treelet, Anticorrelation Treelet, L1 child nodes and L1 leaf nodes, in order to compare their results. In the figures of genes or metagenes expressions the two classes are shown, in green color AML patients and in red the ALL ones, the blue line represents the mean threshold of the classifier computed via 20 fold CV.

	Original Data	**Treelet**	**Anticorrelation Treelet**	**L1 child nodes**	**L1 leaf nodes**
Error %	19.7	25.88	23.82	21	21
Reliability	1.99	1.96	1.73	1.83	1.83

Table 5.7. *Results for 20 CV mondodimensional classification.*

In Table 5.7, the classification errors are compared. Considering only the metagenes and searching the best of them, it could be observed that the classification success is very similar for all the different methods being the best method the L1 norm with the Euclidean distance. The metagene of L1 rotation and the one of L1 leaves are the same linear combination, because the metagene only contains two genes as could be observed in Figure 5.3a, therefore the two ways to calculate the L1 normalization produce the same results.



(a) L1 Norm basis **(b)** Metagene expression

Figure 5.3. *Metagene with L1 norm, in green AML patients and in red ALL ones, the blue line represents the mean threshold of 20 fold CV.*

At figure 5.3a, the two genes involved in the approximation basis are: "GB DEF = M5 muscarinic acetylcholine receptor gene M80333_at" and "N–methyl–D–aspartate receptor modulatory subunit 2A (hNR2A) mRNA U09002_at". As could be observed the two weights of both gens are almost equal, so they are very similar and also near in terms of Euclidean distance. Thus, with this linear combination the metagene (with 21 % error) has filtered the noise of both genes, because individually they have 26.76 % and 22.79 % classification error respectively, so they are noisier. Nevertheless, considering only one dimension to perform the classification task, the best solution is the individual gen "BAGE B melanoma antigen U19180_at" with a 19.7 % error as can be seen on Table 5.7 in the original data column.

It is thought that these results are due to nature of this dataset, which comes from two different machines. Also, the training of the classification is performed only in 50 % of the patients in order to compare the results with the *state of the art*.

5.3.2 Multi-dimensional classification analysis

In this subsection the multi-dimensional classification is analyzed and presented in Table 5.8. In this analysis and in contrast to the monodimensional analysis whole enriched sets (genes and metagenes) have been used to perform the IFFS algorithm. For this reason in the first row all the values are the same, because the best feature is not a metagene. As can be observed in this database it is very difficult to reach a zero error. It is thought that this is caused by the fact that the training phase is performed on a smaller percentage of the whole samples. Also, it is thought that due to the randomness of the process, some samples acquire more weight than others as they appear in different test sets (data not shown).

	Original Data		Treelet		Anticorrelation Treelet		L1 child nodes		L1 leaf nodes	
N	Error %	Reliability	Error %	Reliability	Error %	Reliability	Error %	Reliability	Error %	Reliability
1	19.7	1.99	19.7	1.99	19.7	1.99	19.7	1.99	19.7	1.99
2	13.8	2.80	13.7	2.66	13.8	2.80	13.8	2.80	13.8	2.80
3	10.6	2.96	10.3	2.93	10.0	3.19	10.5	2.96	10.5	2.96
4	9.3	3.26	8.8	3.45	8.4	3.77	7.4	3.12	8.4	3.11
5	6.9	3.13	5.9	4.87	6.3	3.99	6.6	3.24	6.9	3.47
6	6.3	3.25	3.1	5.90	5.1	4.27	5.9	3.22	5.7	3.73
7	5.4	3.77	1.2	6.90	4.4	4.20	5.4	3.20	4.7	3.88
8	4.9	4.13	0.3	7.8	4.1	4.41	5.4	3.43	4.6	3.94
9	4.7	4.44	0.0	8.37	3.5	4.41	5.6	3.59	4.3	4.10
10	4.4	4.45	0.0	9.43	3.4	4.39	–	–	3.9	4.55

Table 5.8. *Multi-dimensional results for Leukemia dataset using IFFS as feature selection method.*

It can be observed that with 3 dimensions or more, the introduction of metagenes improves the outcomes in terms of error classification and reliability. But, only the original Treelet algorithm method has been able to produce a feature set reaching zero error rate. However, for the feature set produced by L1 normalization based on the child nodes the algorithm is not able to improve the result obtained with 9 features. For this reason the 10^{th} row of Table 5.8 does not have any value in the corresponding column; this phenomenon indicates that IFFS has encountered a local minimum from which it is not able to escape.

5.3.3 Comparison with the *state of the art*

Only using the original Treelet it has been possible to reach zero error. Nevertheless, the introduction of the metagenes allows IFFS to reach the perfect classification. Once again, the enrichment of the datasets demonstrates to be useful, in this case even necessary.

Observing the *state of the art* results showed in Table 5.9, only the NSGAA II [26] method reaches zero error rate with only four features. None of the other methods can reach the 100% of correct test set classification. Therefore, our method consisting in expanding the feature set with metagenes and multi-dimensional feature selection process is a good alternative, improving other results from the *state of the art*.

Algorithm	**Error %**	**Number of dimensions**
NSGAA II [26]	0	4
RFE, FLD [25]	3.5	–
PAM [24]	5	–
ICA & LDA [24]	5.35	–
kPCA linear kernel + FDA [23]	5.6	–
PCA + Fisher Discriminant Analysis [23]	5.6	–
LS-SVM RBF kernel [23]	6.44	–
LS-SVM linear kernel [23]	7.14	–
Top scoring pair [24]	10.6	2
IFFS original data	4.4	10
IFFS Treelet	0	9

Table 5.9. *Comparision with* state of the art *for Leukemia dataset. Our methods are highlighted in a grey cell.*

5.4 Lymphoma dataset

The Lymphoma dataset contains the three most prevalent adult lymphoid malignancies and it is available at http://genome-www.stanford.edu/lymphoma. The dataset was first studied in [22]. At this stage of the UPC-CELLEX collaboration project, we only have tackled two classes cancers. For this reason, and as in [26], we have adopted this database as a collection of expression measurements from 96 normal and malignant lymphocyte samples. In other words, considering it as a two classes problem; 42 samples correspond to diffused lymphocyte samples reported large B-cell Lymphoma (DLBCL) and 54 samples of other types considered as only one class. The Lymphoma data contains 4026 genes per sample, and therefore the hierarchical clustering methods generating 4025 metagenes in each method.

5.4.1 Monodimensional classification analysis

In this subsection, the monodimensional classification is analyzed and compared between the different clustering methods and the original data (without enriching via the metagenes). The analysis consists in performing an exhaustive monodimensional classification with a 10 fold CV in the enriched datasets (original genes with the corresponding metagenes). The different methods to build the metagenes are the following clustering methods: Treelet, Anticorrelation Treelet, L1 child nodes, L1 leaf nodes.

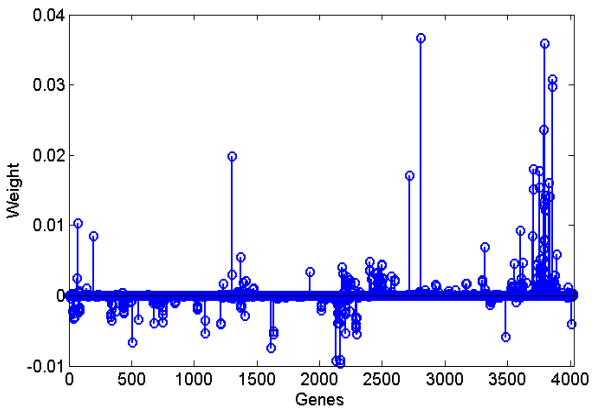
	Original Data	**Treelet**	**Anticorrelation Treelet**	**L1 child nodes**	**L1 leaf nodes**
Error %	5.56	4.44	4.44	3.33	5.56
Reliability	6.80	7.93	7.93	8.42	6.80

Table 5.10. *Comparison results for 10 CV mondodimensional classification for Lymphoma database.*

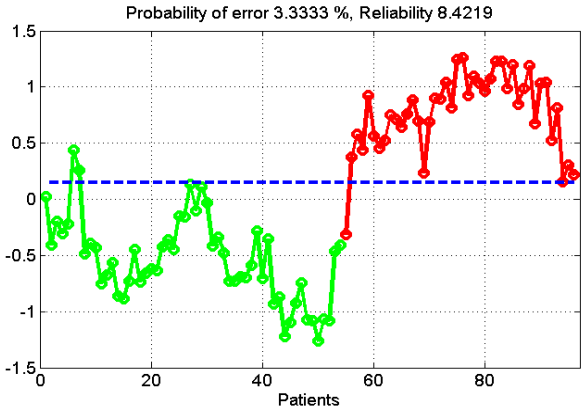
In the Table 5.10, the classification outcomes are compared. It can be seen that the method L1 leaf nodes has the same values than the original data, because it is the same original gene that has been selected as the best feature. Therefore, in one dimension this hierarchical clustering does not help. In the same way, the Anticorrelation Treelet and Treelet algorithms present the same result. In contrast, the incorporation of the metagenes generated by the method L1 child nodes allows to obtain a better outcomes in terms of classification error and reliability, even considering only the best feature to perform a monodimensional classification. This metagene is shown at Figure 5.4b, and its construction basis at Figure 5.4a.

As can be observed, the approximation basis involves the linear combination of 3998 individual genes. As explained in subsection 3.2.2, the L1 normalization in the child nodes generates an approximation basis where a lot of its components have a very small weight. In this example, only 19 genes have a relative big weight in this basis.

As previously, in the Figure 5.4b the metagene expression show the two patient classes, in green color No DLBCL patients and in red color the ones with DLBCL, the blue line represents the mean threshold of 10 fold cross validation. As can be observed this metagene is sufficiently good because only 4 of 96 samples are misclassified.



(a) Approximation basis of the metagene generated by L1 child nodes algorithm.



(b) Metagene obtained with L1 child nodes algorithm.

Figure 5.4. *Best monodimensional metagene for classification and its basis.*

5.4.2 Multi-dimensional classification analysis

Now, the multi-dimensional classification is analyzed and shown at Table 5.11. In order to prove the benefits of introducing a feature set expansion by the inclusion of metagenes from different hierarchical clustering algorithms. The best method is still the L1 child nodes. Also it can be observed that the inclusion of the L1 normalization metagenes allows to reach a zero error using fewer dimensions than using the original Treelet algorithm or using only the original set.

	Original Data		Treelet		Anticorrelation Treelet		L1 child nodes		L1 leaf nodes	
N	Error %	Reliability	Error %	Reliability	Error %	Reliability	Error %	Reliability	Error %	Reliability
1	5.56	6.80	4.44	7.93	4.44	7.93	3.33	8.42	5.56	6.80
2	3.33	8.65	3.33	9.35	3.33	9.35	1.11	8.70	2.22	9.00
3	2.22	9.78	1.11	10.85	1.11	10.85	0	10.27	1.11	9.39
4	1.11	10.90	1.11	12.35	1.11	12.35	0	11.40	0	10.30
5	1.11	12.12	1.11	13.30	1.11	13.30	0	12.64	0	11.54
6	0	14.21	0	12.65	0	12.58	0	13.94	0	12.53
7	0	15.60	0	14.43	0	14.22	0	15.01	0	13.52
8	0	16.67	0	15.59	0	15.68	0	15.84	0	16.18
9	0	18.22	0	17.02	0	17.41	0	16.83	0	17.56
10	0	19.45	0	18.64	0	19.44	0	18.31	0	19.44

Table 5.11. *Multi-dimensional results for Lymphoma dataset using IFFS as feature selection method.*

5.4.3 Comparison with the *state of the art*

For this dataset it only has been possible to obtain two alternative methods that conform the *state of the art*. The Support vector machines (SVM) 1 vs. all with information gain as feature selection method [27] reaches zero error limiting SVM with random coding to the top 150 genes. In the other hand, the NSGAA II algorithm reaches also zero error but involving 12 genes in the classifier. So, thanks to the feature set enhancement by metagenes via hierarchical clustering, it is possible to use fewer features reducing the needed dimensions up to three.

Algorithm	**Error %**	**Number of dimensions**
NSGAA II [26]	0	12
SVM 1 vs. all + information gain [27]	0	< 150
IFFS original data	0	6
IFFS L1 child nodes	0	3

Table 5.12. *Comparision with* state of the art *for Lymphoma dataset. Our methods are highlighted in a grey cell.*

Chapter 6

Study of Treelet as a dataset reduction tool

As Microarrays technology advances, the number of genes per sample analyzed also increases, currently about 60000 genes per sample can be accessed. The analysis time and RAM (computational cost) used by many processing tools developed increase exponentially with the number of observed genes, see Figure A.1 at the appendix.

In order to avoid this problem the raw data extracted from image analysis need to be pre-processed to exclude poor-quality spots. Also this data need to be normalized in order to remove many systematic errors as possible before the downstream analysis is performed [9].

It is possible to achieve good discrimination using only a small fraction of the original expression data because only a small subset of genes are relevant to a specific diagnosis/prognosis problem. Therefore, the last objective of this PFC project is to develop a method that uses the hierarchical clustering structure in order to remove those initial genes that are not relevant.

6.1 Classical approach to reduce the datasets

Inside of raw microarray datasets a lot of genes present a small variation in their profile. It is assumed that these genes will not help to distinguish between the different patients classes. To exemplify it, imagine a gene that presents the same value for all the samples, this kind of gene can not help to classify the kind of patient since it is equal for all of them. The same principle can be applied if the gene is almost flat, that is, it has a small variation in its profile. These genes are removed from the dataset even thought they are not noisy or their lecture does not suffer from a poor-quality reading.

In order to limit the number of genes, the classical approach consists in sorting the genes based on their variance. Then, the gene filtering selects the N genes with highest variance.

6.2 Proposed reduction methodology: *Treelet Pruning*

Instead of performing a filtering on the dataset based on the variance, our proposal aims to *summarize* the original dataset in order to reflect it to the maximum. If the dataset needs to be reduced to a small quantity of features, it is thought that the use of only high variance genes does not guarantee the best classification results.

The proposed reduction uses the tree structure that builds the Treelet algorithm. Each node or metagene represents a sparse set of genes that are linearly combined to obtain this Treelet representative. If this node represents a similar set of p genes each metagene would be similar to this set. As growing up in the tree structure, each metagene represents more individual genes and the similirity with these genes is decreasing. We could represent the p genes by a metagene, when it represents them with a good accuracy. So in order to reduce the dataset we can perform an intelligent pruning of the tree. The pruning can be based on a similarity between the metagenes and the individual genes that each one represents. The similarity function that has been chosen is the Euclidean distance defined in Equation 3.8. But since each metagene represents p individual genes each metagene will have p different distance values. So, two functions that collapse all these different values in a single one are shown in Equations 6.1 and 6.2.

$$d_m = \max d_{m_i} \tag{6.1}$$

$$d'_m = \frac{\sum_i^p d_{m_i}}{p} \tag{6.2}$$

Function 6.1 seeks to ensure that all the individual genes have a similarity less than or equal to the value assigned by this function. On the other hand, Function 6.2 aims to reflect how similar is the metagene to the whole set of genes, performing a mean between all the distances. Any of this two functions can be chosen. Then, the tree structure is gone over in a top down approach to find the firsts nodes that presents a minimum similarity value. An example of this kind of pruning applied to the same tiny dataset presented at Figure 3.3 is shown at Figure 6.1. The pruning has been performed with the Equation 6.1 and selecting the first nodes with a coefficient less than or equal to seven.

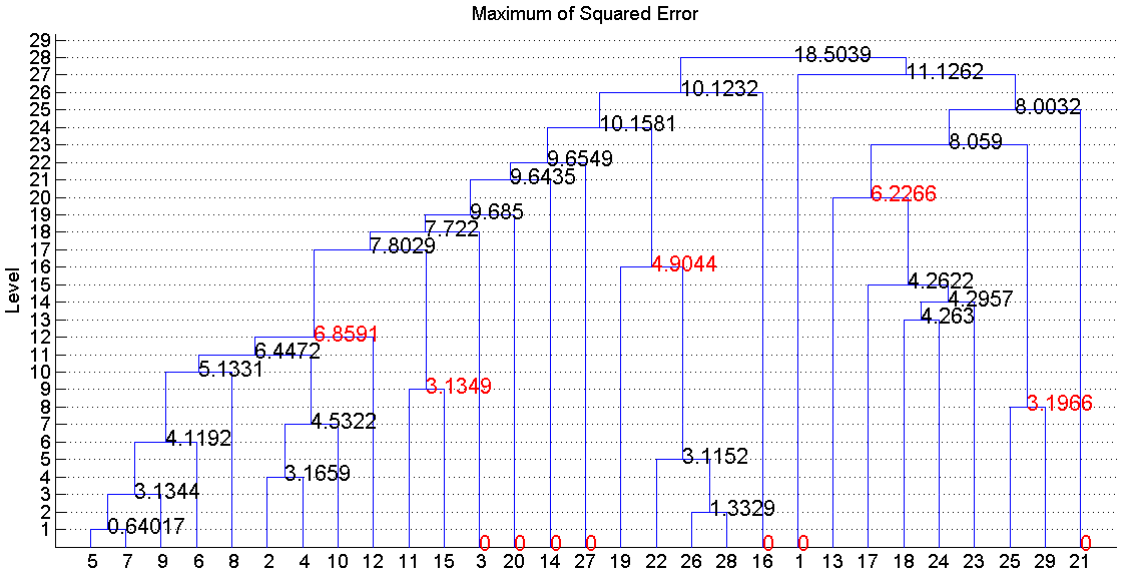


Figure 6.1. *Tiny dendrogram corresponding to the Treelet algorithm with the maximum similarity value overhead to each node. In red the selected features to be part of the reduced set.*

6.2.1 Implementation

To implement this pruning the tree structure should be traversed starting at the top (the root node) and going deeper in the structure until a node accomplishes the desired condition. Also, it should be ensured that all the original feature set is represented. The natural idea is to implement it with a recursive algorithm. But when the original feature set includes a huge quantity of genes, an recursive algorithm will produce an overflow in the memory stack, which has a maximum recursive limit. Therefore, in order to handle with this situation an iterative approach based on virtual stack have been adopted [28]. Two pruning have been developed:

1. Selecting a maximum distance allowed. This number is used as the threshold to perform the pruning. With this method the *quality* of the summary of filtering is ensured, but not the number of features after the pruning.

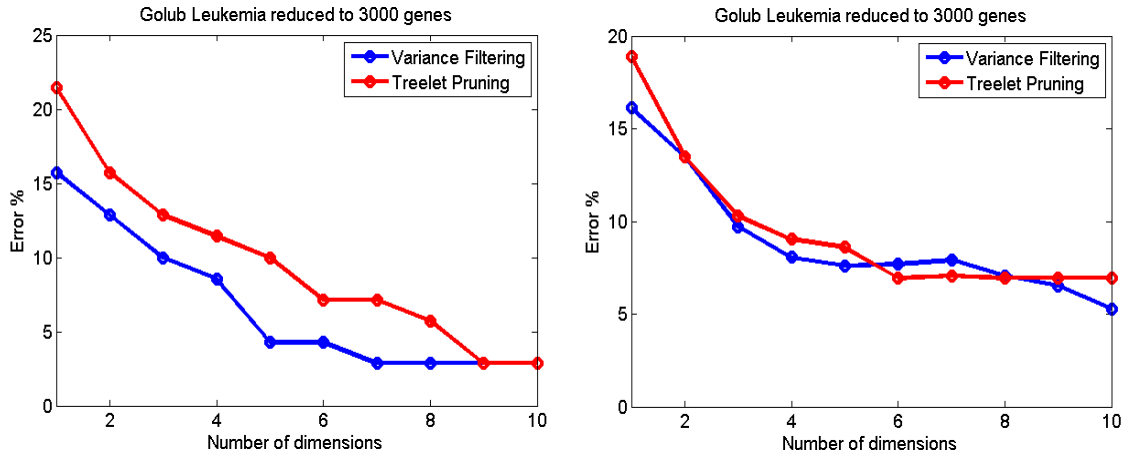
2. Selecting a maximum number of features desired to be part of the summary. In this case the *quality* is not determined a priori, instead the threshold *quality* is searched by a dichotomic search [29].

6.3 Results of Leukemia dataset reductions

In order to evaluate this method to reduce the datasets a criterion should be defined. Since our aim is to use the microarrays to classify the patients, this application has been selected in order to evaluate the dataset reduction.

For this proposal the Leukemia dataset has been selected, it is the same which has been used in section 5.3 but without performing any gene reduction, so every sample has 7129 genes. This dataset is selected because it has the largest number of genes of all the dataset used in this work. Two methods to estimate the error have been adopted. The first is a classical 10 fold cross validation (see section 4.2.1). The second method is the bolstered resubstitution method [16, 17] because this kind of reduction leads to a small-sample scenario. The results show in a compact form the multidimensional analysis up to 10 dimensions (if it is possible). Three different reductions are studied here with the following maximum number of allowed features: 3000, 2000 and 1000. The prunings in this section have been performed ensuring a maximum number of features, and the function to estimate the quality of the metagene is the Equation 6.1.

6.3.1 Data reduction to 3000 features



(a) Comparison for reduction up to 3000 features, error estimated with 10 cross validation.

(b) Comparison for reduction up to 3000 features, error estimated with bolstered resubstitution method.

Figure 6.2. *Comparison for reduction up to 3000 features for Leukemia dataset.*

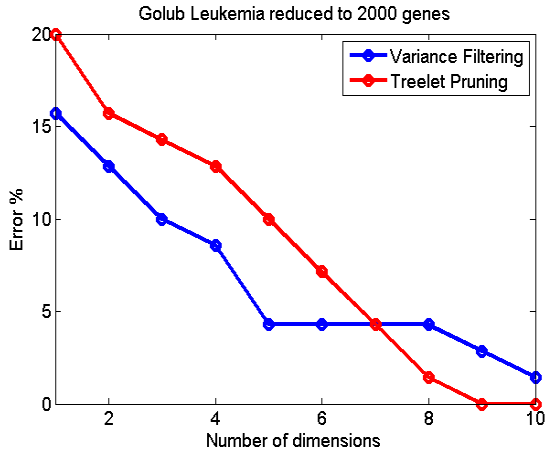
	Error estimated with cross validation				Error estimated with bolstered resubstitution			
	Variance Filtering		Treelet Pruning		Variance Filtering		Treelet Pruning	
N	Error %	Reliability	Error %	Reliability	Error %	Reliability	Error %	Reliability
1	15.71	2.4	21.43	2.32	16.11	2.66	18.89	1.57
2	12.86	2.6	15.71	2.67	13.47	3.36	13.47	2.62
3	10	2.75	12.86	2.58	9.72	3.42	10.28	2.82
4	8.57	3.45	11.43	3.33	8.06	3.44	9.03	2.96
5	4.29	3.79	10	3.54	7.6	4.01	8.61	2.86
6	4.27	4.07	7.14	3.73	7.7	4.86	6.94	3.20
7	2.85	4.4	7.14	4.08	7.9	5.65	7.08	3.06
8	2.85	4.72	5.71	4.71	7.08	6.06	6.94	4.71
9	2.85	5.36	2.86	5.36	6.53	6.99	6.94	5.36
10	2.85	5.98	2.86	6.17	5.28	7.53	6.94	6.17

Table 6.1. *Comparison for reduction up to 3000 features.*

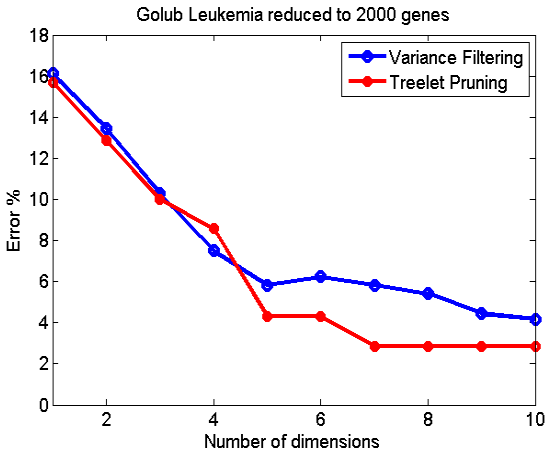
In Table 6.1 the multi-dimensional analysis for the reduction up to 3000 features is shown. The results of estimated error (%) are also shown graphically at Figure 6.2. For this reduction the two methods obtain similar results, but the ones obtained by the variation filtering process are slightly better than the ones obtained by our method. This benefit is bigger when the error of the classifier is estimated with a cross validation. Alternatively, if the bolstered resubstitution method is used, the observed behavior is an initial decrease in classification error in both reduction methods, then reaching a similar value.

6.3.2 Data reduction to 2000 features

In Table 6.2 the multi-dimensional analysis for the reduction up to 2000 features is shown. Otherwise, the results of estimated error (%) are shown graphically at Figure 6.3. It can be observed that for this stronger reduction the two methods to estimate the error rate offer different behaviours. With the cross validation estimation, the variance filtering method to reduce the dataset obtains best results until the 7^{th} dimension, after this point our method achieves better error outcomes, even reaching zero error with 10 dimensions. In contrast, when bolstered resubstitution method is used to estimate the error, the two methods to reduce the dataset obtain similar results until the 4^{th} dimension. When more dimensions are used our method to reduce the dataset allows the classifier to reach lower error rates.



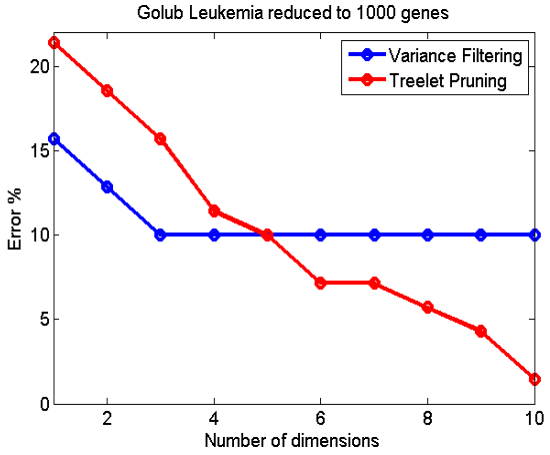
(a) Comparison for reduction up to 2000 features, error estimated with cross validation.



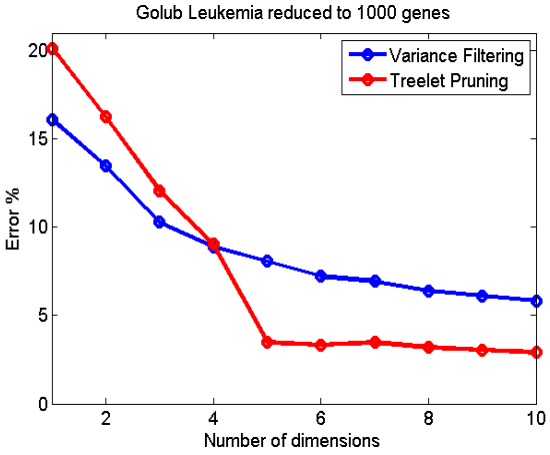
(b) Comparison for reduction up to 2000 features, error estimated with bolstered resubstitution method.

Figure 6.3. *Comparison for reduction up to 2000 features for Leukemia dataset.*

6.3.3 Data reduction to 1000 features



(a) Comparison for reduction up to 1000 features, error estimated with cross validation.



(b) Comparison for reduction up to 1000 features, error estimated with bolstered resubstitution method.

Figure 6.4. *Comparison for reduction up to 1000 features for Leukemia dataset.*

In Table 6.3 the multi-dimensional analysis for the reduction up to 1000 features are shown. The results of estimated error (%) are shown graphically at Figure 6.4. It can be observed that for the heavy reduction the classical reduction method obtains best results until the 4^{th} dimension. Whatever the method used to estimate the error, when more dimensions are used our method to reduce the dataset allows the classifier to reach lower error rates. The error obtained by the variance filtering method suffers a saturation, but our method also suffer this saturation with a low error rate (3 %).

	Error estimated with cross validation				Error estimated with bolstered resubstitution			
	Variance Filtering		Treelet Pruning		Variance Filtering		Treelet Pruning	
N	**Error %**	**Reliability**	**Error %**	**Reliability**	**Error %**	**Reliability**	**Error %**	**Reliability**
1	15.71	2.4	20	2.13	16.11	2.66	15.71	2.4
2	12.86	2.6	15.71	2	13.47	3.36	12.86	2.6
3	10	2.7	14.29	1.99	10.28	3.85	10	2.75
4	8.57	4.29	12.86	2.03	7.5	3.52	8.57	3.45
5	4.29	3.75	10	3.1	5.83	4.03	4.29	3.79
6	4.29	3.99	7.14	3.5	6.25	4.01	4.29	4.07
7	4.29	4.15	4.29	4.98	5.83	4.01	2.86	4.4
8	4.29	4.36	1.43	6.01	5.42	3.99	2.86	4.72
9	2.86	4.86	0	7.38	4.44	4.17	2.86	5.36
10	1.43	6.1	0	8.76	4.17	4.25	2.86	5.98

Table 6.2. *Comparison for reduction up to 2000 features.*

	Error estimated with cross validation				Error estimated with bolstered resubstitution			
	Variance Filtering		Treelet Pruning		Variance Filtering		Treelet Pruning	
N	**Error %**	**Reliability**	**Error %**	**Reliability**	**Error %**	**Reliability**	**Error %**	**Reliability**
1	15.71	2.4	21.43	2.14	16.11	2.66	20.14	2.1
2	12.86	2.6	18.57	2.53	13.47	3.36	16.25	3.24
3	10	2.74	15.71	2.83	10.28	3.85	12.08	3.73
4	10	2.74	11.43	3.31	8.89	3.73	9.03	4.44
5	–	–	10	3.52	8.06	4.15	3.47	5.15
6	–	–	7.14	3.74	7.22	4.24	3.33	5.08
7	–	–	7.14	3.9	6.94	4.25	3.47	5.49
8	–	–	5.71	4.36	6.39	4.3	3.19	5.61
9	–	–	4.29	6.14	6.11	4.43	3.06	5.68
10	–	–	1.43	7.08	5.83	4.52	2.92	5.94

Table 6.3. *Comparison for reduction up to 1000 features.*

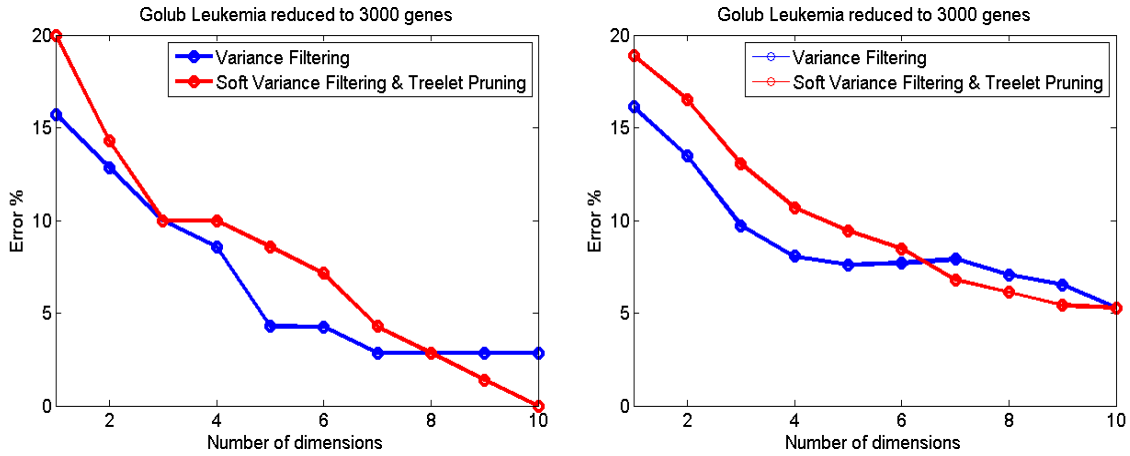
6.4 Analysis of the results

Observing the results previously exposed in section 6.3 our proposed method for reducing the dataset obtains in general better results than the classical method based on the variation filtering. In addition, the method demonstrates its usefulness when a strong reduction in the dataset is needed. As an exemple see the reduction up to 1000 features in our study, wich is a reduction of 85.97 % in the number of genes. But on the other hand, performing a classical filtering reduction is good enough if our computer can handle to work with more features. In our study this situation is exemplified with the reduction up to 3000 features that represents a reduction of 57.91 % in the number of features.When the pruning involves so many features the Treelet Pruning can reflect the original dataset with too much fidelity. Therefore, the reduced dataset starts to contain a lot of almost flat features that are not useful to the classification task.

In order to prove this hypothesis, an initial variance filtering (at percentile 10) is performed to the original dataset in order to remove the almost flat-shape genes. Then, with this modified dataset a Treelet hierarchical clustering is performed and also the intelligent pruning up to the desired number of genes. This solution has been teste for the example involving 3000 features.

The results from this experiment are shown in Figure 6.5 and Table 6.4. It can be observed that when the genes with lowest variance are first removed the outcomes in terms of error classification improves with respect to the experiment shown at Table 6.1. So, it can be a clue to confirm this hypothesis. If the error estimation is perfomed with bolstered resubstitution method the proposed method still obtains low error rates at the higher dimensions. Thus, this new method becomes an interesting technique for microarray reduction.

Nevertheless, to prove it a deeper analysis should be performed with more datasets. Also this analysis should explore the implications of applying the Equation 6.2 to compute the mean distance of the set to the metagene and compare the results. This kind of analysis would imply a lot of time and effort. Therefore, it is beyond of the limits of this PFC project. For this reason this study is proposed as a future work line.



(a) Comparison for reduction up to 3000 features, error estimated with cross validation.

(b) Comparison for reduction up to 3000 features, error estimated with bolstered resubstitution method.

Figure 6.5. *Comparison for reduction up to 3000 features for Leukemia dataset.*

	Error estimated with cross validation				Error estimated with bolstered resubstitution			
	Variance Filtering		Treelet Pruning		Variance Filtering		Treelet Pruning	
N	**Error %**	**Reliability**	**Error %**	**Reliability**	**Error %**	**Reliability**	**Error %**	**Reliability**
1	15.71	2.4	20.00	1.87	16.11	2.66	18.89	1.57
2	12.85	2.6	14.28	2.82	13.47	3.36	16.53	2.34
3	10	2.75	10	2.78	9.72	3.42	13.06	2.76
4	8.57	3.45	10	2	8.06	3.44	10.69	2.99
5	4.28	3.79	8.57	3.75	7.6	4.01	9.44	3.32
6	4.27	4.07	7.14	4.42	7.7	4.86	8.47	3.58
7	2.85	4.4	4.28	5.39	7.9	5.65	6.81	3.5
8	2.85	4.72	2.85	5.72	7.08	6.06	6.11	3.49
9	2.85	5.36	1.4	6.22	6.53	6.99	5.42	3.46
10	2.85	5.98	0	7.14	5.28	7.53	5.28	3.46

Table 6.4. *Comparison for reduction up to 3000 features.*

Chapter 7

Conclusions and future work

This final chapter reviews the project achievements and the obtained conclusions. Also, some future work lines are presented based on the open issues detected during the project development.

7.1 Conclusions

In this work, the characteristics of the microarray datasets and their utility for classifying patients have been studied in order to adapt the Treelet algorithm (an existing technique) to this kind of data.

Other works have only applied Treelets as feature transformation tool [30]. We have used it to expand the original feature set with the metagenes generated as linear combination of individual genes. Starting form the Treelet, various new clustering algorithms have been proposed and explained in chapter 3. Each algorithm produces a different clustering with different properties and therefore a different metagene set.

In order to prove the utility of performing this expansion of the feature set, in chapter 4, the feature selection task is introduced for choosing a specific classifier for performing the experiments.

The different alternatives have been tested individually in order to evaluate the classification performance, and the results are presented in chapter 5. Moreover, three different public datasets have been used in order to evaluate how the classification ability depends on the feature expansion.

Experiments have showed how performance depends on the adopted feature set. When the error estimation is performed via cross validation, one of our proposed technique has allowed to obtain a better classifier for two of the three analyzed databases that the ones proposed in the *state of the art*.

Also a new approach to cluster the genes has generated metagenes that are perfectly adapted to the training data. This method is the classification clustering which, for Colon dataset, allowed to perform a perfect monodimensional classification. But on the other hand, this method implies a hight computational load and a high dependence on the training set data. When it has been applied to an independent set, the method is not able to classify samples that are different from those used in the training phase, making its generalization more difficult. Nevertheless, if the training data were more abundant and would represent the sample variability, then the classification clustering could become the best alternative.

A summary of the best metagene generation for each dataset follows here. For Colon dataset, the best alternative is the classifier clustering using one feature only. But, ignoring this specific method, both the hierarchical clustering techniques that use a L1 normalization reach zero error with 5 features. If only the original genes without metagenes are used the classifiers need an extra dimension to reach the same error, but with a smaller reliability. For Leukemia dataset, the best metagene generation is the Treelet clustering, which obtains zero error using 9 features. If only the original genes are used, then the classifiers are not able to reach zero error, indeed they only can reach 4.4 % error using 10 dimensions. Finally, for Lymphoma dataset, the best metagene generation is the one generated by the hierarchical clustering with a L1 normalization based on the child nodes. With this metagenes the classifier obtains zero error using 3 features. When only the original dataset is used, the classifier needs 6 features to reach the same classification error. So, for two of the three studied datasets, the introduced L1 normalization has achieved the best outcomes.

This results demonstrates how the feature set expansion with metagenes produces better classifiers than using only the original data. This is, when the feature set expansion with metagenes is adopted, it makes possible to obtain smaller or equal error rates with fewer features than using only the original features, in all three studied datasets.

Finally, a new application for the hierarchical clustering has been studied. With it, it is possible to generate a summary of the original dataset in order to reduce it. Using the tree structure created by the Treelet algorithm and defining a loss of quality between the metagene representative and the individual genes that it represents a pruning has been developed. The method has proved its usefulness when a strong reduction is needed. Alternatively, when the needed reduction is moderate, it has been shown that creating a summary with the Treelet

pruning introduces some features that do not help the classifier. At last, a brief study has been performed in order to prevent this effect on the summarization implementing a pre-filtering step. This alternative has shown better results in the performed experiment.

7.2 Future work lines

At the end of this work, some future work lines are presented as possible continuation paths from the current stage.

A possible work line is to explore the usefulness of Anticorrelation Treelet. In our experiments the Anticorrelation Treelet and the original Treelet algorithms have presented an almost similar hierarchical structure, therefore a lot metagenes are the same for both methods. It is possible that this behaviour is caused by the definition of the angle for the local pair-wise rotation in Equation 3.1, which is defined as $|\theta_L| \leq \frac{\pi}{4}$.

Another work line is to perform a deeper analysis with the classification clustering. This analysis should apply this method to a dataset with more samples in order to define a representative training set and then validate if the method is not overfitted to the training set. If under this condition the classification clustering continues presenting an overfitting, then a new construction method should be proposed.

In spite of only taking into account a better reliability value for the classifiers, a new criterion may be studied. This criterion could privilege the robustness in one of the classes. If the metagenes behaves differently in the two classes, the classification could be more robust for a specific class. Additionally, this different robustness for different classes that present different metagenes can be used in order to build a classifier for a specific class.

An additional work line is to perform a deeper analysis in the study of Treelet as a dataset reduction tool. As has been mentioned, the current results are promissing. But in order to prove the method, it should be applied to different datasets. Also the implications of applying the Equation 6.2 to compute the mean distance of the set to the metagene should be studied and compared. Finally, the pre-filtering step needs a deeper analysis to figure out if it resolves the explained problematics of this method.

Additionally, the big challenge is to use the hierarchical structure and not only the metagenes. In order to adapt the technique to a pathway analysis. Since genes never act alone in a biological system – they are working in a cascade of networks. So, analyzing the microarray data in a pathway perspective could lead to a higher level of understanding of the system. If several genes are assigned to the same group by cluster analysis, they might be co-regulated or

involved in the same signaling pathway. Analyzing the promoters of this group of genes can often reveal common regulatory motifs and unveil a higher level of network organization in the biological system [31]. This topic has been suggested by the CELLEX foundation as an future line work in order to confirm some discovered pathways.

Appendix A

Study of computational cost

This appendix will discuss the computational complexity and cost involved in the calculation of the metagenes with different hierarchical algorithms presented in chapter 3. This analysis is performed in terms of used RAM and the needed time versus differents sizes of datasets.

A.1 Algorithm

The presented algorithms are developed in $Matlab^{®}$, because this is a research project and it needs a lot of debugging and interaction with plots and representations. These requeriments make $Matlab^{®}$ the selected platform for this stage of the project. $Matlab^{®}$ is reasonably fast, extremely intuitive environment and a lot of documentation can be found. But $Matlab^{®}$ is an interpreted language, and it can be slow, indeed some programming practices like loops can make it very slow.

Almost of the methods proposed in this PFC are based on the Treelet algorithm. Therefore in this section the pseudocode of the orginal Treelet algorithm is presented at it was developed in [4].

Algorithm:

1. Compute the sample covariance and correlation matrix from original data.
   (a) Initialize the set of indices δ. Each index represents a tree branch that can be combined into a higher level. Initially, the set of indices contains every original feature.

Repeat for $1, \ldots, \#_{features} - 1$:

2. Find the two most similar variables from δ according to the correlation matrix.
3. Perform a local PCA on the selected pair throught a rotation that decorrelates the pair via the covariance matrix.
4. Update the covariance and correlation matrices.
5. Retain the principal component to the pair in δ to represent both features.

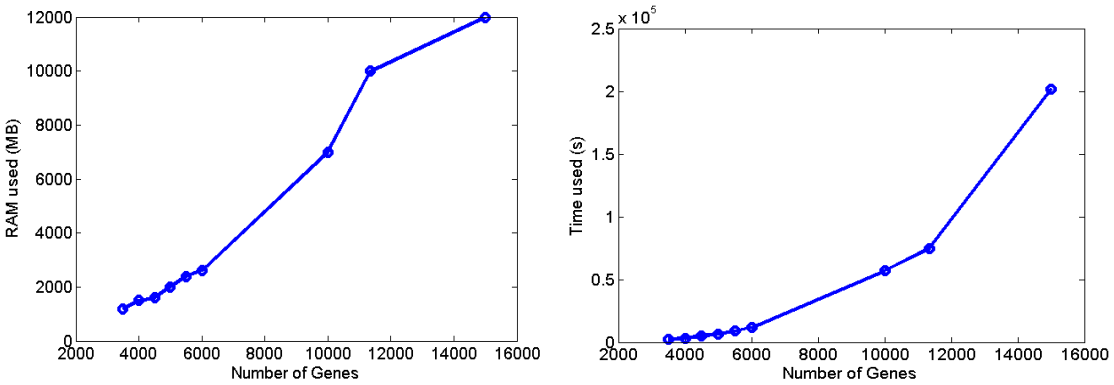
A.2 Cost in terms of RAM and time

In this section a brief analysis of the computational cost in terms of RAM and time used by the algorithm described in section A.1 is performed. To perform this analysis a new dataset have been used. This is a restricted dataset wich has not been presented due its nature. However, for this kind of analysis it is the ideal one, because it represents well the microarray specificity of small sample number and gene abundance. The original dataset is composed of 54675 features to classify only 22 samples.

The reults of this analysis are presented in Table A.1 and Figure A.1, and both are structured as follows: The time needed and the RAM used by the algorithm have been measured for this dataset filtered out with different number of genes.

Number of genes	RAM (MB)	Time	
		hours	minutes
3500	1200	0	38
4000	1500	0	56
4500	1600	1	20
5000	2000	1	49
5500	2400	2	27
6000	2600	3	15
10000	7000	15	50
11349	10000	20	48
15000	12000	56	5

Table A.1. *RAM (measured in MB) and time (hours and minutes) used by the Treelet algorithm.*



(a) RAM (MB) **(b)** Time (s)

Figure A.1. *RAM and time used by the Treelet algorithm.*

As can be observed, the time needed by the algorithm increases exponentially with the number of genes in the database as also does the RAM used. For every different dataset four differents hierarchical clustering algorithms have be applied. So in order to obtain the results in a resonable time, at this stage of the development the maximum number of gene set has been fixed to 11349 genes.

A.3 Classifcation clustering requeriments

As has been explained in the previous section the *classical* hierarchical clusterings (Treelet, Anticorrelation Treelet, L1 child nodes and L1 leaf nodes) have been used to process dataset up to 11349 genes. The RAM used by them increases in order to have a local storage for the matrices of simmilarity and covariance. Then, these matrices can be updated only tracking the local changes. However, the classification clustering should perfom a classification between all the genes or metagenes in δ. And, even though most of the results of the classifiers can be reused the average time used in the Colon dataset is 13 hours and a half. The Colon dataset is the smallest dataset used in this work and it involves 62 patients with 2000 genes for each patient. Also the time used increases exponentially, and therefore this algorithm has only been applied to the Colon dataset. But in contrast this algorithm uses less than 2 GB of RAM for the Colon dataset.

Bibliography

[1] H. Curtis, S. Barnes, and A. Schnek, *Biologia/ Biology*, Editorial Medica Panamericana Sa de, 7th edition, 2008.

[2] S. Dudoit and J. Fridlyand, "Classification in microarray experiments", *Statistical analysis of gene expression microarray data*, pp. 93–158, 2003.

[3] T. Hastie et al., "Supervised harvesting of expression trees", *Genome Biology*, vol. 2, no. 1, pp. research0003.1–research0003.12, 2001.

[4] A. B. Lee B. Nadler and L. Wasserman, "Treelets – an adaptive multi–scale basis for sparse unordered data", *Annals of Applied Statistics*, vol. 2, no. 2, 2008.

[5] J. Peto, "Breast cancer susceptibility–a new look at an old model", *Cancer Cell*, vol. 1, no. 5, pp. 411 – 412, 2002.

[6] E. P. Solomon et al., *Biology*, Saunders College Publishing, 4th edition, 1996.

[7] S. Schreiber and E. Lander, "Scanning life's matrix: genes, proteins, and small mollecules", in *The 2002 Holiday Lectures on Science*, 2002.

[8] T. R. Golub et al., "Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring", *Science*, vol. 286, no. 5439, pp. 531–537, 1999.

[9] Yuk Fai Leung and Duccio Cavalieri, "Fundamentals of cDNA microarray data analysis", *Trends in Genetics*, vol. 19, no. 11, pp. 649 – 659, 2003.

[10] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: A review", 1999.

[11] M. B. Eisen et al., "Cluster analysis and display of genome-wide expression patterns", *Proc. Natl Acad. Sci. USA*, vol. 95, pp. 14863–14868, 1998.

[12] R. Tibshirani, "Cluster analysis and display of genome-wide expression patterns", *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[13] I. T. Jolliffe, *Principal Component Analysis*, Springer, 2002.

[14] G. H. Golub and C. F. van Loan, *Matrix Computations*, Johns Hopkins University Press, 1996.

[15] J. Rahnemführer, "Exploratory data analysis for microarrays", Tech. Rep., Computational Biology and Applied Algorithmics, Munich, 2005.

[16] U. Braga-Neto and E. Dougherty, "Bolstered error estimation", *Pattern Recognition*, vol. 37, no. 6, pp. 1267–1281, june 2004.

[17] U. Braga-Neto, "Fads and fallacies in the name of small-sample microarray classification - a highlight of misunderstanding and erroneous usage in the applications of genomic signal processing", *Signal Processing Magazine, IEEE*, vol. 24, no. 1, pp. 91 –99, jan. 2007.

[18] T. Hastie et al., *Supervised Harvesting of Expression Trees*, Springer, 2000.

[19] J. Novovivcova P. Pudil and J. Kittler, "Floating search methods in feature selection", *Pattern Recogn. Lett.*, 1994.

[20] Songyot Nakariyakul and David Casasent, "An improvement on floating search algorithms for feature subset selection", *Pattern Recogn.*, 2009.

[21] U. Alon et al., "Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays", vol. 96, pp. 6745–6750, 1999.

[22] A. Alizadeh et al., "Distinct types of diffuse large b-cell lymphoma identified by gene expression", *Nature*, 2000.

[23] N. Pochet et al., "Systematic benchmarking of microarray data classification: assessing the role of non-linearity and dimensionality reduction", *Bioinformatics*, vol. 20, no. 17, pp. 3185, 2004.

[24] D.S. Huang and C.H. Zheng, "Independent component analysis-based penalized discriminant method for tumor classification using gene expression data", *Bioinformatics*, vol. 22, no. 15, pp. 1855, 2006.

[25] C. Lai et al., "A comparison of univariate and multivariate gene selection techniques for classification of cancer datasets", *BMC bioinformatics*, vol. 7, no. 1, pp. 235, 2006.

[26] K. Deb and A. Reddy, "Reliable classification of two-class cancer data using evolutionary algorithms", *BioSystems*, 2003.

[27] C. Zhang T. Li and M. Ogihara, "A comparative study of feature selection and multiclass classification methods for tissue classification based on gene expression", *Bioinformatics*, vol. 20, no. 15, pp. 2429, 2004.

[28] Wikipedia, "Tree traversal — Wikipedia, the free encyclopedia", 2011, [Online; accessed 10-July-2011].

[29] Wikipedia, "Dichotomic search — Wikipedia, the free encyclopedia", 2011, [Online; accessed 10-July-2011].

[30] L. Sheng et al., "Treelets as feature transformation tool for block diagonal linear discrimination", in *Genomic Signal Processing and Statistics, 2009. GENSIPS 2009. IEEE International Workshop on*. IEEE, pp. 1–4.

[31] P. Sudarsanam Y. Pilpel and G. M. Church, "Identifying regulatory networks by combinatorial analysis of promoter elements", *Nature genetics*, vol. 29, no. 2, pp. 153–159, 2001.