A Tutorial Review of Microarray Data Analysis

Alex Sánchez and M. Carme Ruíz de Villa
Departament d'Estadística. Universitat de Barcelona.
Facultat de Biologia. Avda Diagonal 645. 08028 Barcelona. Spain.
asanchez@ub.edu;mruiz_de_villa@ub.edu

July 7, 2008

Contents

1 Foreword and objectives

This paper presents a review of microarray data analysis. It is reasonable to ask what is the use of yet another review when many good ones can be easily found. We intend to give to this work a slightly different orientation. We do not pretend to be neither so brief that we simply mention each topic, nor so exhaustive as to describe each method completely. Our goal is to give an overview which is deep enough as to understand the basic ideas and to demonstrate the use of the basic tools, that can be briefly summarized as:

- First, it is oriented towards an audience formed mainly by statisticians, that is, we can assume that the potential readers have a good knowledge

of standard statistical methods and a lesser knowledge of related topics such as molecular biology or bioinformatics.

- One problem for many statisticians considering to start working on microarray data analysis is how to implement all the methods and concepts in practice. Thus,a second goal of this paper is to simplify this approach by providing some completely worked through examples with the corresponding R code which can be used as templates for potential studies. So we expect that, after reading the paper, one should be able to start analyzing microarray data by oneself.

- To reach our goals many emerging technologies and the methods for their analysis cannot be seen in detail. Nevertheless they will be mentioned in the last sections, simply to get acquaintance about their existence.

This review is organized as follows: Section 2 presents basic concepts in molecular biology and the technology of microarrays. Section 4 describes preliminary aspects such as experimental design jointly with topics, such as normalization, which are specific for this field. Section 5 is the core of the paper describing some of the available statistical methods to perform the different types of analyses. Section 3 contains worked, reproducible examples, which can be used as "templates" for new analyses. Section 6 deals with less statistical yet important aspects such as the data management or software available. Functional genomics is a very quickly evolving field, and since the advent of microarrays, less than a dozen years ago, many new technologies –and the corresponding statistical issues there have appeared. These are briefly considered in section 7.

2 Introduction

In recent years a new type of experiments are changing the way that biologists and other specialists analyze many problems. These are called "high throughput experiments" and the main difference with those that were performed some years ago is mainly in the quantity of the data obtained from them. Thanks to the technology known generically as microarrays, it is possible to study nowadays in a single experiment the behavior of all the genes of an organism under different conditions.

The data generated by these experiments may consist from thousands to millions of variables and they pose many challenges to the scientists who have to analyze them. Many of these are of statistical nature and will be the center of this review.

There are many types of microarrays which have been developed to answer different biological questions and some of them will be explained later. For the sake of simplicity we start with the most well known ones: expression microarrays. This section is organized as follows: first we present some examples of biological problems whose research requires performing experiments with microarrays. Next we set up some biological background about gene expression

and related topics. The section ends with a comprehensive presentation of the two most popular types of expression microarrays.

2.1 High troughput experiments

Microarrays are useful in a wide variety of studies with a wide variety of objectives. Many of these objectives fall into the following categories [59].

1. A typical microarray experiment is one who looks for genes *differentially expressed* between two or more conditions. That is, genes which behave differently in one condition (for instance healthy [or untreated or wild-type] cells) than in another (for instance tumor [or treated or mutant] cells). These are known as *class comparison* experiments.

2. When the emphasis is on developing a statistical model that can predict to which class a new individual belongs we have a *class prediction* problem. Examples of this are predicting the response to a treatment (e.g. classes are "responder" and "non-responder") or the evolution of a disease (e.g. recidivated or cured).

3. Sometimes the objective is the identification of novel sub-types of individuals within a population. For example it has been shown that certain types of leukemia present some subclasses that are very hard to distinguish morphologically but which can be classified using gene expression. This is an example of *class discovery*.

4. *Pathway Analysis* studies are those that try to find genes whose co–regulation reflects their participation in common or related biochemical processes.

The statistician will easily associate to this list of problems an statistical method such as testing in class comparison, discrimination in class prediction or clustering in class discovery. While it is the case that many classical statistical methods have been found to be adequate in many cases it is also true that other situations have required the adequation or even the development of new methods and tools to fit well the nature of these data.

2.2 Biological background

Gene expression has to do with the behavior of the cells and thus an understanding of basic biological concepts is highly recommended if not essential. A quick introduction to the minimum necessary concepts can be found in [4]. In this section we present briefly those concepts that will be used later.

2.2.1 DNA, proteins and the central dogma

Many important functions performed in cells involve *proteins*. A protein can be represented as a linear sequence of simpler molecules called *amino acids*. By the

way it is the simplicity of this representation that has favored the analysis of protein sequences by computer scientists and mathematicians setting the basis of traditional bioinformatics.

Proteins do not self-assemble. The information needed to specify their sequence, structure and function is contained in DNA, which in higher organisms (eukaryotes), is located in the cell nucleus, packaged in the chromosomes.

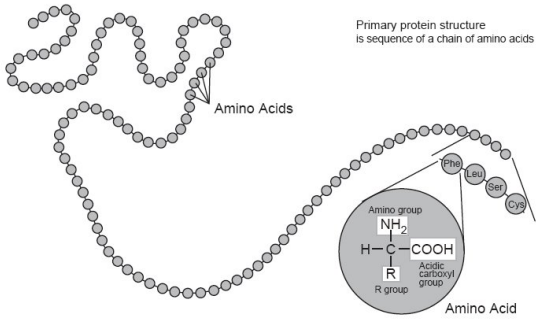

Figure 1: Primary protein structure is defined by the sequence of amino acids.

DNA is organized as a chain of small molecules, called *nucleotides*. There are four different nucleotides Adenosine (A), Guanine (G), Cytosine (C) and Thymidine (T), which are usually referred to as "bases". DNA may be single or double stranded (the well-known "double helix"). DNA forms a double strand by establishing chemical bonds between pairs of *complementary bases* on the two strands. Adenine binds (only) with Thymine and Guanine binds (only) with Cytosine. This complementarity is a central feature of DNA and it is behind such important processes as replication and gene expression.

Another important molecule is RNA which, like DNA, is constructed from nucleotides, but instead of the Thymine (T), it has a similar molecule, Uracil (U), which is not found in DNA. Because of this difference RNA does not form a double helix, instead they are usually single stranded, but may have complex spatial structure due to complementary links between the parts of the same strand. RNA has different functions in the cell. Mainly, we are interested in its role as an intermediate between DNA and proteins.

It is common to use the term *polynucleotide* to describe a chain of either DNA or RNA. Some polynucleotide chains are unstable, and, instead of working with them it is common to use their complementary sequence which has to be specifically synthesized. In this case, one talks of *cDNA* or *cRNA*.

2.2.2 Genes and protein synthesis

A gene can be defined [4] as *a continuous stretch of a genomic DNA molecule, from which a complex molecular machinery can read information (encoded as*

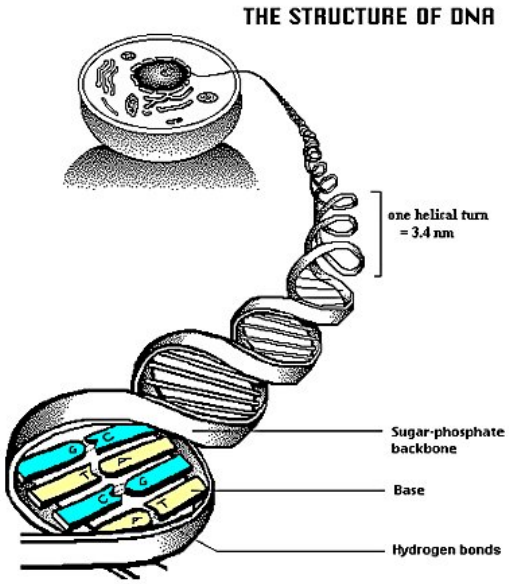

Figure 2: Illustration of the double helical structure of the DNA molecule.

a string of A, T, G, and C) and make a particular type of a protein or a few different proteins.

The correspondence between the DNA and the amino acid sequence of a protein is stated by the *Central Dogma of Molecular Biology* (see figure 3).

By virtue of the central dogma, genes are "decoded" to perform different functions, the best known of which is to synthesize proteins. This is done in a process that follows three stages: (1) transcription, (2) splicing, and (3) translation (see figure 4)

1. In the **transcription** phase one strand of DNA molecule is copied into a complementary pre-mRNA (or nuclear RNA). During this process the two-stranded DNA double helix is unwound and information is read only from one strand (sometimes called the W-strand).

2. **Splicing** (see figure 5) removes some stretches of the pre mRNA, called *introns*. The remaining sections, called *exons*, are then joined together. Exons are the part of the gene that code for proteins and they are interspersed with non coding introns which must be removed by splicing. The number and size of introns and exons differs considerably among genes and also between species. The result of splicing is mRNA.

   Many eukaryote genes are known to have different alternative splice variants, i.e. the same pre-mRNA producing different mRNAs, known as alternative splicing (see figure 6).

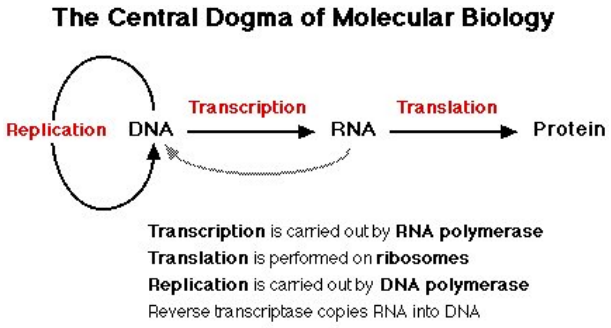

Figure 3: The Central Dogma of Molecular Biology.

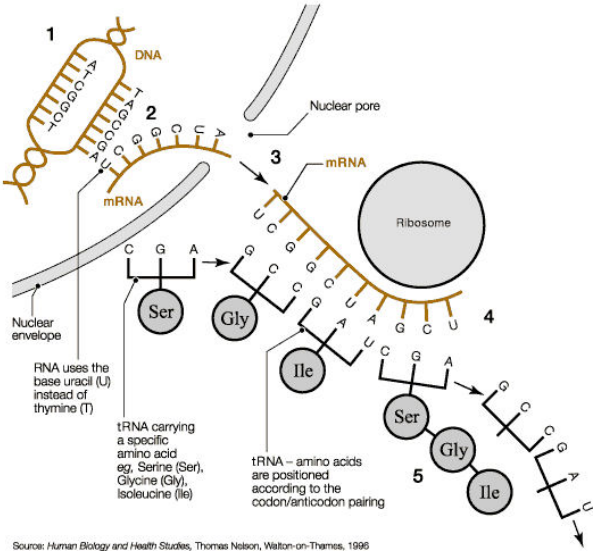

Figure 4: The synthesis of proteins reflects the central dogma.

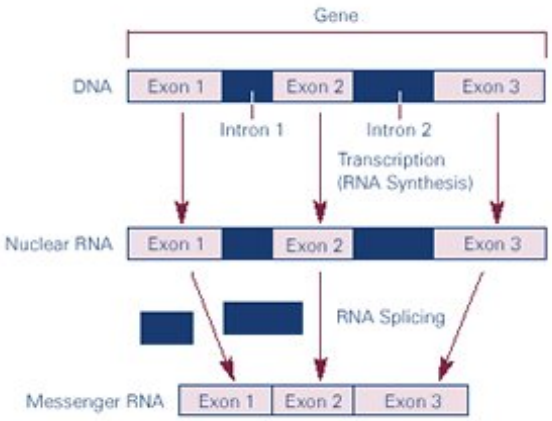

Figure 5: The Splicing Process.

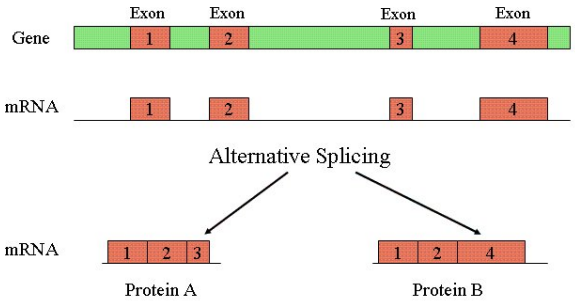

Figure 6: Alternative splicing can produce different forms of the same gene.

3. **Translation** is the process of making proteins by joining together amino acids in the order encoded in the mRNA. An amino acid is determined by 3 adjacent nucleotides (triplets) in the DNA. This is known as the *triplet* or *genetic code*. Each triplet is called a *codon* and codes for one amino acid. As there are 64 codons and only 20 amino acids the code is redundant, for example histidine is encoded by CAT and CAC.

Second base in codon

First base in codon	U	C	A	G	Third base in codon
U	Phe	Ser	Tyr	Cys	U
	Phe	Ser	Tyr	Cys	C
	Leu	Ser	STOP	STOP	A
	Leu	Ser	STOP	Trp	G
C	Leu	Pro	His	Arg	U
	Leu	Pro	His	Arg	C
	Leu	Pro	Gln	Arg	A
	Leu	Pro	Gln	Arg	G
A	Ile	Thr	Asn	Ser	U
	Ile	Thr	Asn	Ser	C
	Ile	Thr	Lys	Arg	A
	Met	Thr	Lys	Arg	G
G	Val	Ala	Asp	Gly	U
	Val	Ala	Asp	Gly	C
	Val	Ala	Glu	Gly	A
	Val	Ala	Glu	Gly	G

Figure 7: The Genetic Code.

The end of translation is the final part of gene expression and the final product is a protein, whose sequence corresponds to the sequence encoded by the mRNA. Proteins can be post-translationally modified e.g., by addition of sugars or cleavage (chopping), and this affects their location and function.

Biologists used to believe in the paradigm - 'one gene - one protein'. Now this is known not to be true: due to alternative splicing and post-translational modifications one gene can produce a variety of proteins. There are also genes that do not encode proteins but encode RNA (for instance tRNA and ribosomal RNA).

After following this introduction one should be able to understand one of the assumptions underlying microarray data analysis: *Given that genes are expressed by transcribing and translating their information into m-RNA –which*

will be later used to synthetize proteins– if we are able to find out ***which*** *and* ***how much*** *mRNA is around we should be able to find out which genes and with which intensity they are being expressed.*

Actually, it is not so simple because complications may arise, but it will be helpful as a guide to understand the basic rationale of microarray analysis.

2.2.3 Nucleic acids hybridization

Hybridization is the process by which two complementary, single-stranded nucleic acids combine into a single molecule. Nucleotides bind to their complement (A with T and C with G) under normal conditions, so two perfectly complementary strands will bind to each other readily. This is called *annealing*. However, due to the different molecular geometries of the nucleotides, a single inconsistency between the two strands will make binding between them more energetically unfavorable.

Hybridization has been used to identify genes in cellular DNA for more than 30 years now ([5]). Microarrays, discussed in next section are based on the same principle, but differ in the quantity. Whilst traditional hybridization techniques, such as "Southern blot" can detect one gene at a time, microarrays are intended to do the same with thousands of genes in a single experiment.

2.3 Microarrays

2.3.1 The technology

DNA microarrays, also known as DNA chips, are tools that allow the identification and quantification of the mRNA transcripts present in the cells.

As we have suggested above the number of molecules of mRNA, coming from the transcription of a given gene, can be considered as an approximation to the level of expression of that gene. There is a great variability related with that assertion: some genes act on other genes without transcription; in other cases a high activity is the consequence of small mRNA concentration. In spite of this variability the broad idea can be considered valid and here we consider it so.

A microarray consists of a solid surface on which strands of polynucleotide –called *probes*– have been attached or synthesized in fixed positions. Two types of expression microarrays are the most popular between users. One of the main differences among them relies on how the way these probes are put on the slide.

- *Spotted or cDNA microarrays* take their name because probes are synthesized apart and printed mechanically on the slide. The term "cDNA" is used because the probe is a complimentary copy of the original sequence and each probe represents one gene.

- In *oligonucleotide chips*, where main representatives are Genechip or Affymetrix (c), the name of the commercial brand that manufactures them, the probes are directly synthesized on the surface. The term "oligonucleotide" refers to the fact that the synthesis process allows to create only small fragments

so that a gene is not represented by one probe but by as a set of them (a "probe set").

To start a microarray experiment [46] RNA is extracted from the subject cells. After this, some of its molecules are substituted by others containing a fluorescent dye. The resulting labelled transcripts are called *targets*.

Once the samples are prepared they are deposited over the array and left inside a hybridization chamber for some hours. The labelled targets bind by hybridization to the probes on the array with which they share sufficient sequence complementarity. After this time the array is washed which eliminates those targets which have not hybridized.

The way in which the previous step is performed is the second important difference between the two types of chips.

- In spotted microarrays cDNAs from **two** tissues of interest, labelled with fluorescent dyes of different color (usually red and green), are hybridized to a single chip (see figure 8). The two targets are said to compete to hybridize with the probes. For obvious reasons spotted chips are also called "two-color arrays".

- The Affymetrix system hybridizes only one sample per chip (see figure 9). This requires more slides per experiment and does not enjoy the advantage of using competitive hybridization, however it simplifies experimental design and is based on a much more sensitive technology.

At this point each probe on the microarray may be bound to a certain quantity of labelled target that, following our basic assumptions, should be proportional to the level of expression of the gene represented by that probe. To determine the amount of sample hybridized the microarray is illuminated by a laser light that causes the labelled molecules to emit fluorescence (proportionally to their quantity). This fluorescence is captured by a scanner yielding an image that consists in a grid of shined spots, corresponding each one to a probe. Finally, this image will be transformed into numbers and will be the basis of the analysis.

2.3.2 Expression measures

DNA microarrays quantify gene expression by means of fluorescence intensity which is captured by the scanners into an image. The images are turned into numbers by a process which will not be discussed here, but which can be considered to be relatively reliable and stable.

Each technology generates different types of images and thus generates different quantities which have to be adequately operated to provide some kind of estimate of a unique variable: the gene expression.

When the image obtained from a *cDNA microarray* is analyzed ("quantitized") several quantities are produced for each spot (see figure 10). Although this depends on the software used, basically they consist on (i) signal measures,

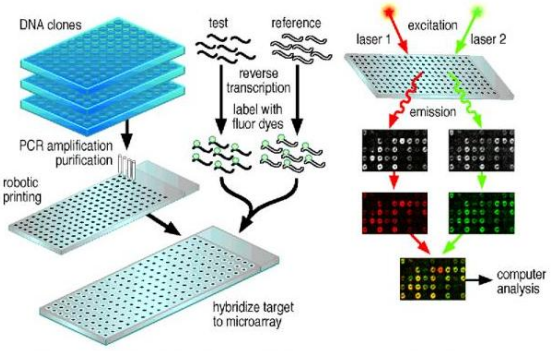

Figure 8: Two color cDNA chips

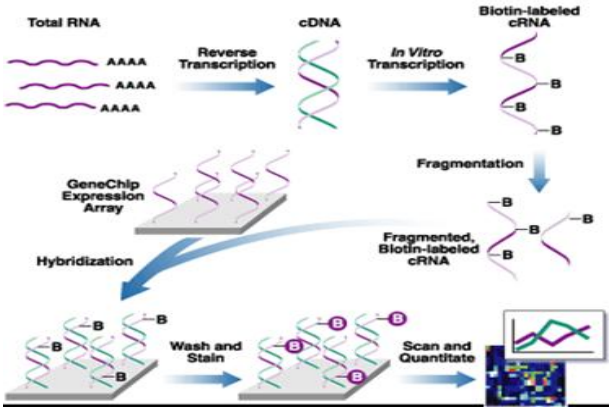

Figure 9: One color affymetrix chips

Red (R)or Green (G), for each channel , (ii) background measures,R_b , G_b, intended to measure fluorescence not due to hybridization, and (iii) some quality measures for that spot. These quantities can be used to provide a naive measure of such as the expression ratio:

$$M = \frac{R}{G}, \tag{1}$$

or the background-corrected expression ratio:

$$M = \frac{R - R_b}{G - G_b}. \tag{2}$$

It is very common to use the base 2 logarithm of this quantity as the final outcome of *relative expression*. This is mainly due to two reasons: On one side the expression data are better approximated by a log-normal distribution, and on the other side taking logarithms symmetrizes the differences, making the interpretation easier.

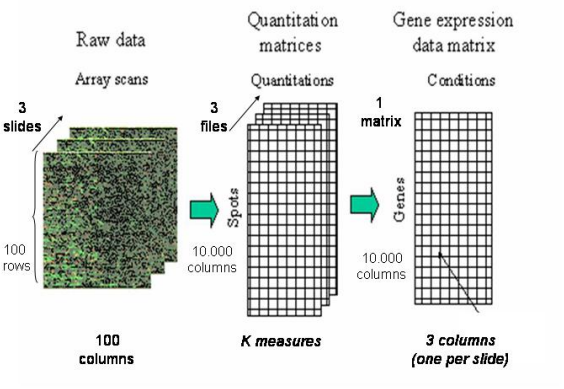

Figure 10: Image quantitation for two color cDNA chips

Figure 11 shows the values of red and green channels for the first 10 genes in the first two arrays of the Callow dataset, described below, jointly with the corresponding log–ratios.

Genechip (Affymetrix) arrays represent each gene as a set of probes corresponding each one to one short (oligonucleotide) chain. Indeed each probe is a "probe pair" made of a "perfect match" (PM) probe that corresponds to the original DNA chain and a "mismatch" (MM) probe whose central nucleotide has been changed. The idea underlying this approach is that anything that hybridizes with the mismatch probe should not represent "real expression" but anything else, that is background. Affymetrix suggested to combine both measures in a background corrected expression measure. The formula used has

ID	gene(=spot)	X1(=block row)	X2 (=block column)	C1.G	C1.R	C2.G	C2.R	C1.R/G	C2.R/G
1	BLANK	1	1	5592.58	2765.58	4749.89	1768.22	0.4945088	0.37226546
2	BLANK	1	1	4746.38	2868.43	3088.12	2277.18	0.60434057	0.7374001
3	mSRB1	1	1	2108.48	1236.32	3669.53	1546.84	0.58635605	0.42153627
4	BLANK	1	1	548.46	383.62	708.16	532.5	0.69944937	0.75194871
5	BLANK	1	1	856.48	377.36	715.64	525.44	0.44059406	0.73422391
6	BLANK	1	1	629.39	402.09	552.49	493.09	0.63885667	0.89248674
7	unknown	1	1	18176.65	13782.69	10004.73	8562.25	0.75826349	0.8558202
8	W13502	1	1	9605.38	3561	11334.83	7097.08	0.37072974	0.62613026
9	W13547	1	1	10362.71	5838.29	9650.38	6115.41	0.56339413	0.63369629
10	W13549	1	1	3191.72	2014.22	3792.29	2596.83	0.63107666	0.68476567

Figure 11: Red and Green Channel values jointly with the corresponding log ratios

evolved but a naive estimate given in the first versions is:

$$Avg.diff = \frac{1}{|A|} \sum_{j \in A} (PM_j - MM_j), \tag{3}$$

where A is the set of probe pairs whose intensities do not deviate more than three times the standard deviation of the mean intensity over all probes.

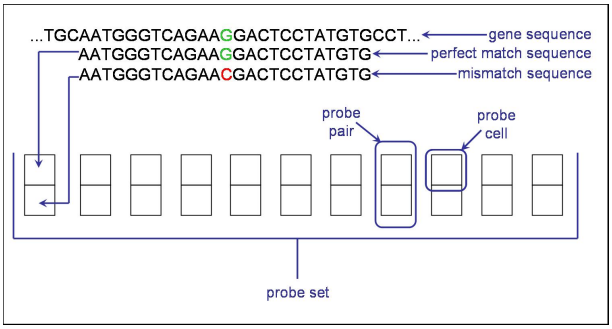

Figure 12: The Perfect Match and the Mismatch

The most important difference between these two ways to measure expression does not rely on the specific formula which has evolved in both cases but in the fact that, whereas in affymetrix chips one has a single expression value for each condition, in cDNA arrays one works with a relative expression measure between two conditions. Although Affymetrix yields more precise estimates, relative expressions have a much more intuitive interpretation.

3 Examples

One of the handicaps for statisticians who may consider entering this field is how to start applying their knowledge to these problems. We present below some examples, which will be used along the paper to illustrate different concepts.

Tn order to make the examples more realistic they use real data sets from real published studies, that is the data can be obtained online from public databases and they correspond to published research papers. However, to simplify the examples only the broad goal of the papers is considered in the examples, that is, no attempt is made to reproduce the results, but only to do a similar approach to that taken in the paper.

A brief description of each dataset is the following:

- The dataset ("celltypes") has been obtained from the public database `caarray` maintained by the National Institute of Health, NIH (https://caarraydb.nci.nih.gov/caarray/performExperimentSearchAction.do). It corresponds to a paper from [15] which studies the molecular basis of processes regulated by a molecule (cytokine) in aged mouse.

- The second dataset, ("arabidopsis") has been obtained from another public database the ("Gene Expression Omnibus,http://www.ncbi.nlm.nih.gov/geo) where it is stored with the identification number `GSE1110`. It corresponds to an experiment performed to investigate changes in gene expresion in *Arabidopsis thaliana* as response to IndoleAcetic Acid (IAA). The use of this dataset as well as many details of this example have been inspired in the excellent Bioconductor manual by Thomas Girke available at http://faculty.ucr.edu/~tgirke/Documents/R_BioCond/R_BioCondManual.html.

- A third dataset ("melanoma")is represented by a group of cutaneous malignant melanomas and unrelated controls which were analyzed by Bittner*et al.* [9] who performed an analysis to detect tumor subtypes based on gene expression profiles. The data set is also available at GEO, and is one of the first datasets deposited in that database.

- The dataset ("bladdercancer") also available at GEO with the number GDS183, corresponds to a study performed by Dyrksjot *et al.* [26] to identify clinically relevant subclasses of bladder carcinoma using expression microarray analysis. This dataset is also available as a BRB array tools project downloadable from the BRB web site http://linus.nci.nih.gov/BRB-ArrayTools.html.

- The last dataset, "Callow", has become a classical example of cDNA data analysis. It is based on an experiment performed by Callow *et al.* [13, 24] to study lipid metabolism and atherosclerosis susceptibility in mice. The goal of the study was to identify genes with altered expression in the livers of transgenic mice with SR-BI gene over–expressed (T) compared to normal control mice (C). The data are available as text files at http://www.stat.berkeley.edu/users/terry/zarray/Html/apodata.html.

A complete analysis of each dataset using `R`or other publicly available tools is available as supplementary material at http://estbioinfo.stat.ub.es/pubs/MDAreview.

4 The microarray data analysis process(MDA)

The goal of this section is to present an integrated view of the whole process of analyzing microarray data (see figure 13). Many review papers discuss at this level the statistical techniques available for the analysis. However given that this paper is aimed at statistically-trained readers we will omit elementary concepts and we will try to focus on how statistics may/must be used in this specific context.

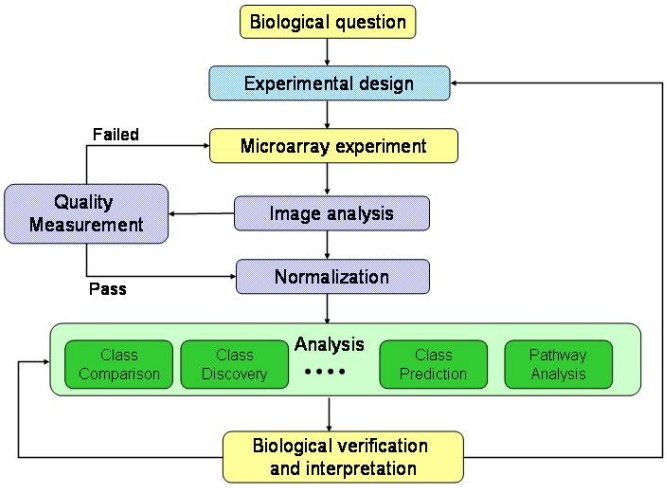

Figure 13: The Microarray Analysis Process

Microarrays and other genomic data are different in nature from the classical data around which most statistical techniques have been developed. In consequence, in many cases it has been necessary to adapt existing techniques or to develop new ones in order to fit the situations encountered.

We will examine some key components of microarray analysis, experimental design, quality control, preprocessing and statistical analysis. In the last section we will consider some topics where open questions still remain and which can be considered attractive for statisticians who wish to focus some of their research in this field.

One of the handicaps for statisticians who may consider entering this field is how to start applying their knowledge to these problems. We will present some real examples, which we will use along the paper to illustrate some concepts. We will also show how to make a complete analysis of these data using **R**, which has become a *de facto* standard in the field.

4.1 Experimental design

4.1.1 Sources of variability

Genomic data are very variable. Figure 14 adapted from Geschwind ([31] illustrates some of these sources.

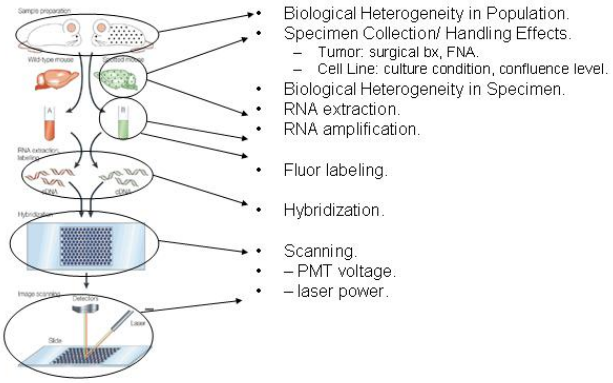

Figure 14: Sources of Variability in Microarray Data

As usual in most experimental situations we can distinguish between systematic and random variation.

Systematic variation is mostly due to technical procedures whereas random variation is attributable to both technical and biological reasons. Examples of systematic variation can be found in RNA extraction, labelling or photodetection. Random variation can be related to many factors such as DNA quality or to the biological characteristics of the samples.

The natural way to deal with random variation is, of course, to use an appropriate experimental design followed by adequate statistical inference tools. Issues related with experimental design will be discussed in this section and those related with application of statistical methods will be discussed in section 5.

Traditionally corrections for systematic variation are estimated from the data in what is generically called "calibration". In this context we will talk of "normalization" which will be discussed in section 4.2.

4.1.2 Replication

Usually one distinguishes two types of replication in microarray analysis:

- *technical replication* is used when several replicates of the same biological material are used. This can be either replicate spots on the same chip or different aliquots of the same sample hybridized to different microarrays.
- *biological replication* is done when measurements are taken from multiple cases.

Technical replication provides measurement–level error estimates an biological replication provides estimates of population–level variability.

4.1.3 Power and sample size

Surprisingly, early microarray experiments used few or no biological replicates. The main explanation for this fact -apart of statistical illiteracy- was in the high costs of each microarray. In few years the necessity of replication has become undisputed, and at the same time the cost of chips has decreased considerably. It is common now to use at least from three to five replicates per experimental condition but this consensus has appeared more by empirical reasoning than from the availability of adequate models for power analysis and sample size.

In recent years there has been an important affluence of papers describing methods for power and sample size analysis. In spite of their variety, no method appears as a clear candidate for use in practical situations. This is probably due to the complexity of microarray data mainly because genes are not independent, so that correlation structures exist in the data, but most dependencies are unknown making these structures very difficult to estimate.

As indicated by Allison ([2]) although there is no consensus about which sample-size determination procedures are best, there is a consensus that power analyses should be done, that newer methods specifically for microarray research should be used, and, of course, that more replicates generally provide greater power.

4.1.4 Pooling

In the microarray context, pooling means combining mRNA from different cases in a unique sample. Two reasons have been argued in favor of this. Sometimes there is not enough RNA available, and this is the only way to obtain enough material to do the arrays. Another, more controversial reason, is the belief that variability among arrays can be reduced by pooling. The rationale is that combining samples is equivalent to "averaging" expressions, and, "as it is known, averages are less variable than individual values". In spite of the weakness of this argument it is true that in certain situations pooling can be appropriate and many statisticians have devoted their efforts in recent times to help answering the "to pool or not to pool question" ([39]). For example if biological variability is high relative to measurement error, and biological samples are inexpensive relative to array cost an appropriate pooling strategy can be clearly cost-efficient.

In any case pooling should not be used for any type of studies. If the goal is comparing mean expressions ("class comparison" below) it can work adequately, but when what the goal of the experiment is to build predictors that rely on individual characteristics it should clearly be avoided.

4.1.5 Single vs dual Channel Microarray Design

In two color arrays two experimental conditions are applied to each array. This allows the estimation of the effect of the array, as a block effect. In Affymetrix or other single channel arrays each condition must be applied to a separate chip, not making possible to estimate the effect of the arrays, which on the by other hand, is usually considered to be very small relative to treatment effects, due to the industrial process used to produce these chips.

As a consequence of the preceding, experiments using single channel arrays can be considered "standard" experiments, so that traditional concepts and techniques of experimental design can be readily applied to them.

Dual channel present a more complicated situation. On one hand the "two colors" are not symmetrical, that is, with the same amount of material an array hybridized with one or another color, say Cy5 or Cy3, will emit signals with different intensity. The usual way to deal with this problem is *dye–swapping* which consists of using two arrays for the same comparison with the dyes changed, that is, if in the first array sample 1 is labelled with Cy3 and sample 2 with Cy5, in the second array this is reversed (see figure 15).

On the other hand, the fact that only two conditions can be applied to each array complicates the design, either because usually there are more than two conditions, or because it is not recommendable to directly hybridize two samples in one array, creating artificial pairings.

The problem of how to assign samples efficiently to microarrays, given a number of conditions to be compared and a fixed number of available arrays has been studied intensively ([40])

The most commonly used design within the biological community is the *reference design* where each condition of interest is compared with samples taken from some standard reference common to all the arrays (see 16 (a)).

Reference designs allow to do indirect comparisons between the conditions of interest. The main criticism raised against this approach is that 50% of the hybridization resources are used to produce a control or common reference signal of non–intrinsic interest to the biologists. In contrast, a loop design compares two conditions via a chain of other conditions, thereby removing the need for a reference sample (see 16 (b)).

The selection of the best design from a set of possibilities can be done in several ways. A common approach is to rely on the A-optimality procedure that selects the design which minimizes some function of the variances of parameter estimates. Using this criteria it can be shown (Kerr et al, 2000) that the

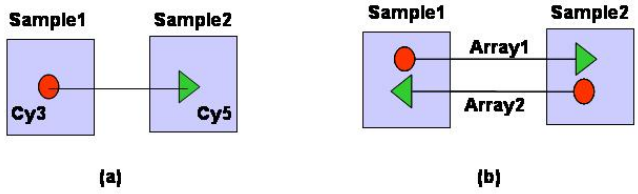

Figure 15: (a) Simplified representation of a design. Each arrow stands for a single two-channel array where the origin indicates the Cy3 dye. (b) Dye swapping.

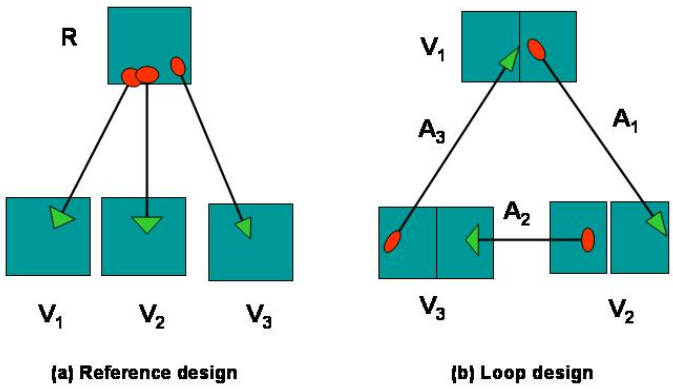

Figure 16: (a) Reference design. (b) Loop design.

theoretical relative efficiency of reference vs loop design is:

$$\sqrt{\frac{\mathrm{tr}\,(C_L(X_L^tX_L)^{-1}C_L^t)}{\mathrm{tr}\,(C_R(X_R^tX_R)^{-1}C_R^t)}} \tag{4}$$

where X_R and X_L are the design matrices for the reference and the loop design, respectively, and C_R and C_L are the matrices that transform the two designs to the same parametrization.

Finding optimal designs, however, is a non-trivial task, particularly for designs with many microarrays and many conditions. Kerr and Churchill ([39]) showed that it is possible to search for A–optimal designs exhaustively only when the number of slides and conditions is less than 10, which is not particularly realistic for most microarray designs.

Other authors have tried different approaches. For example Witt et al. ([65]) used simulated annealing to search the design space and find local optima which offer relatively good solutions.

4.2 Preprocessing

A microarray experiment produces one set of images which are transformed into numerical values representing absolute (single–channel) or relative (two–channel) intensities.

As in any statistical analysis, and particularly in image analysis, the quality of the data must be checked first. High throughput data have an additional difficulty: the huge data matrices obtained make it virtually impossible to detect most problems by visual inspection, what has led to the development of specific quality control procedures.

4.2.1 Quality control

The goal of the quality control step is to determine if the whole process has worked well enough so that the data can be considered reliable.

There are no standard methods for microarray QC although there are groups such as MAGE (see http://scgap.systemsbiology.net/standards/mage_miame.php) trying to develop standards and one recent important study based on hundreds of arrays (The MicroArray Quality Control or "MAQC", [8]) was devoted to review this problem.

Most quality controls are based on images and plots although in the case of Affymetrix numerical summaries whose use is very extended have also been developed .

QC for two channel arrays Quality control for two–color arrays is mainly based on inspection of images or plots such as:

- Image inspection to detect irregularities, such as scratches, bubbles, or high background.

- Signal and signal-to-noise histograms are inspected to detect possible abnormalities or excessively high background.
- Most scanner programs can generate *spot flags* indicating how good the spot can be considered. These values can be used later to filter out some of these spots.

Figure 17 shows some diagnostic plots for two channel microarrays.

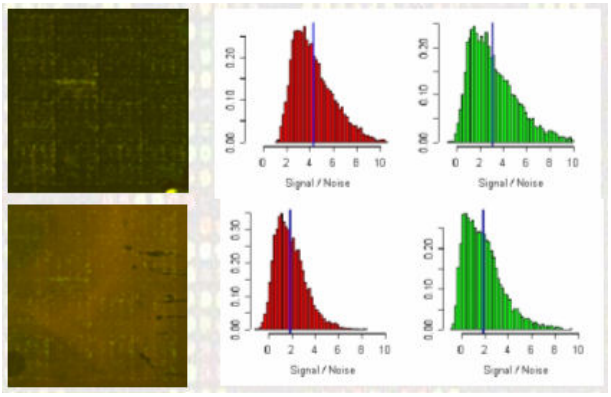

Figure 17: Good quality images (up) should have low background and a high signal to noise ratio. Bad quality images (down) have high background and low signal to noise.

QC for one channel arrays In single channel (mostly Affymetrix) arrays it is slightly different:

- Histograms or other plots such as degradation plots are useful for a first visual inspection and can help to detect arrays with serious problems.
- Affymetrix provides numerical summaries (background, presence calls, scale factor) whose values can be compared to determine array quality.
- State of the art quality control consists of fitting a linear model to the probe signals along arrays and analyzing the residuals. It can be seen that arrays experimenting some problems which might not show in other plots will appear here as clearly deviated from the rest.

Figure 18 shows some diagnostic plots for one channel microarrays.

4.2.2 Background Correction and Normalization

Once the quality of the data has been assessed it is still necessary to make some preprocessing before the analysis. Essentially, this means to go through two or three steps depending on the type of array:

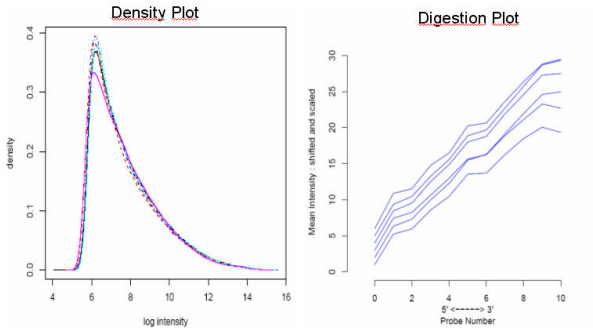

Figure 18: Diagnostic plots for the celltypes example. Degradation plots indicate the quality of RNA hybridization along the probesets

1. A *background adjustment* must be performed to remove signal due to non-specific hybridization, that is signal emitted by other things than sample hybridized to probe.

2. A *normalization* of the data must be done to correct for systematic biases due to causes such as different dye absorption, spatial heterogeneity in the chip or others.

3. In Affymetrix arrays, it is necessary to summarize the different signals obtained from all the probes representing one gene in a unique value.

Background Correction The goal of the microarray production process is to obtain an intensity value which can be considered proportional to the level of expression. This is based on determining how much hybridization has been produced between the sample and the targets.

It is known that a part of the observed signal is due to non-specific binding, that is, a small quantity of the sample may combine to non-complementary chains. Besides, some of the signal may be due to non–biological sources. Altogether there is a need to estimate and remove that signal due to specific ("real") hybridization from that due to any other reasons, generically called *background*.

In the first microarray studies a naive approach was used. It consisted of estimating the intensity by subtracting a background from a signal measure, both provided by the scanner. The main problem with this approach is that it could give negative intensity estimates.

Different methods have been developed as alternatives and several comparisons have been published recently (Ritchie *et al.* [55], Freuendberg *et al.* [28]). A general conclusion of these studies is that model–based methods are those performing best at removing background.

Three commonly used methods are: `normexp` ([60]) for two channel arrays, VSN ([36]) for both types of arrays and RMA ([37]) for oligonucleotide chips.

Interestingly the last two methods combine background correction and normalization, to be discussed below, in the same process.

Normalization Normalization is a key point in the microarray analysis process and much effort has been devoted to develop and test different methods ([53, 69]). One reason for such abundance is that there are different technical artifacts that must be corrected for, and not every method can deal with all of them.

In general, normalization methods are based on the following general principle: most genes in the array are either not expressed or equally expressed in any condition. Only a small amount of genes show changes of expression between conditions.

This gives an idea of how should a plot of the intensities look like. For instance, if there were no technical artifacts, in a two channel array, a scatterplot of Red vs Green intensities should leave most points around a diagonal line. Any deviation of this situation should be attributable to technical, non–biological reasons, and consequently it should be removed. This has lead to a very popular normalization method consisting of estimating the transformation to be applied as a function of the intensities using the lowess method on a transformed representation of the scatterplot known as MA–Plot.

Figure 19 (a) displays a scatterplot of Red vs Green channel in array # 1 of the "Callow" example. The fact that the data are not centered around the diagonal suggests the need for normalization. A very popular representation, which helps to better visualize this asymmetry are MA plot (19(b)). Geometrically they represents a rotation of the scatterplot, where the meaning of the new axes is:

- $A = \frac{1}{2}(\log_2(R * G))$: the average log–intensity of the two channels,
- $M = \log_2 \frac{R}{G}$: The (logarithm) of the relative expression between both channels (usually known as "log–ratio").

Figure 20 shows the effect of normalizing the data using the lowess method. After fitting a lowess to the data a different quantity is subtracted from each point, depending on its intensity ("A" value). As a consequence the transformed data is not only centered but also symetrical around zero.

The lowess method normalizes expression values to make intensities consistent within each array. This is called a *within slides normalization* approach. In many situations it is also necessary to achieve consistency between arrays and methods such as *scale* or *quantile* normalization can be applied. The idea of the *scale normalization* is simply to scale the log-ratios to have the same median-absolute-deviation (MAD) across arrays. *Quantile* normalization, which can be used in both one and two-color arrays, ensures that the intensities have the same empirical distribution across arrays and across channels.

One channel arrays, present different technical artifacts requiring different normalization methods. The most used method for this type of chips is RMA

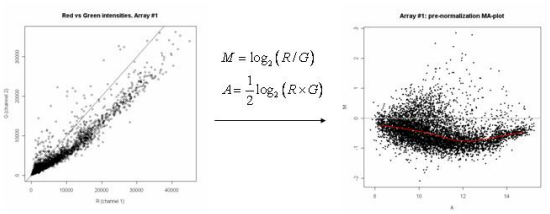

Figure 19: (a) Scatterplot of R vs G (b) MA plot (intensity vs log-ratio)

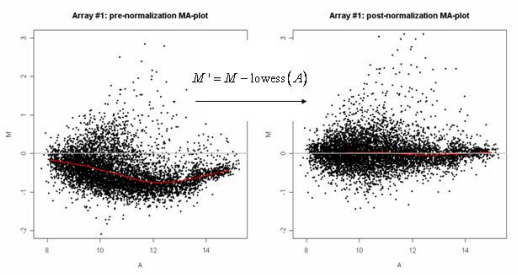

Figure 20: (a) MA plot on original (raw) values (b) MA plot on normalized values

("Robust Multichip Average"). It consists of three steps: a background adjustment based on a probe –level model, a quantile normalization and, finally, a summarization integrating the values of all probes corresponding to one gene. RMA is very popular between statistically–oriented researchers because it is based on elaborated mathematical models which allow to understand the rationale beneath the method. A conceptually simpler approach is the one proposed by the manufacturer of the chips: the MAS5 algorithm. Some studies comparing both (and other) methods ([35, 11, 28]) conclude the superiority of the RMA method, although, this is not a closed discussion yet.

5 Statistical Analysis

The steps described in the previous section are preparatory for data analysis. The output of this initial process is the *gene expression matrix*, whose rows (1000-50000) represent the genes and whose columns represent the samples (from 2 to several hundreds). It is interesting to note that the structure of this data matrix is different to the commonly used in statistics: rows represent variables and columns represent individuals, so that the *curse of dimensionality* appears in all its strength.

In the next paragraphs we will briefly describe the different types of problems with which an investigator is faced. Obviously, it is in that part of the study where the statistician will play the most important role, or, equivalently, where its absence can be most prejudicial.

As in any statistical analysis a main point is to clearly determine the outcome or the response variable. In this case this must be a measure of expression but depending on the technology used it may have different forms:

- For two-channel arrays the most common approach is to rely on *relative expression*, that is the response variable is a log–ratio of intensities,

$$Y_g = \log \frac{R_g}{G_g}. \tag{5}$$

- Another possibility, usually applied in one–channel arrays is to rely on *absolute expression*, that is the response variable, Y_g is the intensity value of each single array measured in logarithmic scale.

5.1 Class Comparison

The class comparison problem can be defined as the selection of genes whose expression is significantly different between conditions. These are called "differentially expressed genes".

Differential expression analysis is one of the fields where statisticians have been involved since the introduction of microarray technologies. In consequence there have been developed many models and methods for the analysis. Some are based on parametric models whereas other rely on non–parametric approaches

in order to overcome the difficulties associated with distributional assumptions. A comparative review of all methods exceeds the purpose of this work and has already been done elsewhere (see e.g. Pan *et al.* [51] or Cui *et al.* [17]). However, in order to give a "feeling" of what and how it can be done, several common approaches will be presented in this section.

- Model–based methods use analysis of the variance models ANOVA to capture the main sources of variability in the experiment. In this case a single model is used for all the genes simultaneously. An example of such approach, the `MAANOVA` method ([67]) is presented below.
- Global tests, in spite of their name, analyze each gene separately, using a common model which can be parametrical or not. We will briefly discuss two methods, which can be considered representative, the SAM method ([63]), a popular non-parametric approach, and the `limma` method ([60]) a parametric approach using linear models and empirical bayes.

5.1.1 Model–based methods

Wu et al ([67]) proposed an analysis of variance model specified in two stages for two–color microarrays where the expressions are treated separately (that is, it relies on absolute expression values). The first-stage model is as follows:

$$Y_{ijgr} = \mu + A_i + D_j + AD_{ij} + r_{ijgr}, \tag{6}$$

where the indices track the (A)rray (i), the (D)ye, (j), the gene (g) and the (r)eplicated measurement (r). The first stage generates the term r_{ijgr} which, in a second stage, is modelled in terms of gene–specific effects as:

$$r_{ijgr} = G + TG_{ij} + DG_j + AG_i + \epsilon_{ijr}, \tag{7}$$

where G is the average intensity associated with a particular gene, AG_i is the effect of the array on that gene, DG_j is the effect of the dye on that gene and ϵ_{ijr} is the residual. TG_{ij} is called the "treatment–by–gene" term and is the main interest in the analysis which captures variations in the expression levels of a gene across samples. It must be noted that this approach does not need a previous normalization to account for dye or array effect, because this is already done by the corresponding dye or array terms.

The gene–specific model can be modified for Affymetrix data by removing the DG and AG terms because there is no dye factor ("one-color") and the array effects become part of the residual error term.

In practice what a user will do is to fit model 6 to the data and call differentially expressed those genes where the interaction term TG is significative.

There may be found different variations of this approach for instance incorporating random effects or changing the hierarchical structure of the models.

5.1.2 Global tests

One of the main practical differences between model–based and global methods lies in the way that normalization is done. Model–base methods do it implicitly when the model is fitted whereas global tests require a previous normalization step as described in 4.2.2.

If one considers one gene at a time a microarray experiment can be seen as "simply an experiment" so that a reasonable way to analyze it is to use a standard linear model approach.

This is however considered inefficient due mainly to two common problems in this type of experiments: first, sample sizes very small, which complicate variance estimation; second, the variances themselves may be very variable between the genes. These facts altogether may yield non–stable variance estimates, which at their time induce high variability in F–like test statistics. To deal with this problem a commonly accepted strategy is *variance shrinkage* which consist of relying on improved variance estimates, $\tilde{S}$, where this improvement comes from borrowing information from all the genes in the array. The test statistics used by the SAM ([63]) or the limma methods ([60])use different versions of variance shrinkage.

$$t = \frac{\overline{X}}{\hat{\sigma}_n} \approx \frac{\overline{X}}{\tilde{S}}, \tag{8}$$

where

$$\tilde{S}_{SAM} = c_0 + \hat{\sigma}_n \tag{9}$$

$$\tilde{S}_{limma} = \sqrt{\frac{d_0\hat{\sigma}_0^2 + d\hat{\sigma}_n^2}{d + d_0}} \tag{10}$$

where $\hat{\sigma}_n$ is the usual standard error estimate (with d degrees of freedom) for each gene (subindex omitted). In SAM c_0, is estimated from the data using a permutation method. In limma d_0 and s_0 are unknown and are estimated from the data using an empirical bayes approach.

5.1.3 Sample size calculations

There are different models to do power analysis of microarray data but many of them (see e.g. Simon *et al.* [59]) are mere generalizations of traditional procedures or make so many simplifications that are hard to believe. Besides this the number of arrays usually recommended is far from the affordable number for most experiments ([45, 62]). What many users do is to look for a tradeoff between cost and reproducibility and, in practice they tend to use a fixed number of arrays such as 3 or 5 without many additional considerations.

5.2 Multiple testing

The analysis of microarrays on a gene–by–gene basis involves multiple testing. Testing thousands of genes is likely to produce hundreds of false positives if no correction is applied.

One approach is to control the family-wise error rate (FWER), which is the probability of accumulating one or more false positive errors over a number of statistical tests. The simplest FWER procedure is the Bonferroni correction but more sophisticated approaches such as the permutation–based one–step method or the Westfall and Young step-down adjustment have been developed. Dudoit *et al.* ([23]) contains an excellent review of multiple testing applied to microarray data analysis.

FWER criteria may be too restrictive because control of false positives implies a considerable increase of false negatives. In practice, however, many biologists seem willing to accept that some errors will occur, as long as this allows findings to be made. For example a researcher might consider acceptable a small proportion of errors (say 10%–20%) between her findings. In this case, the researcher is expressing interest in controlling the false discovery rate (FDR), which is the proportion of false positives among all the genes initially identified as being differentially expressed. Unlike a significance–level which is determined before looking at the data, FDR is a post–data measure of confidence. It uses information available in the data to estimate the proportion of false positive results that have occurred. If one obtains a list of differentially expressed genes where the FDR is controlled at, say, the 20%, one will expect that a 20% of these genes will represent false positive results. This represents a less restrictive approach than controlling the FWER.

The decision of controlling FDR or FWER depends on the goals of the experiment. If the objective is "gene fishing" allowing a certain number of false positives is reasonable and FDR is preferred. If instead one is working with a shorter list which one wishes to verify if some specific genes are expressed, then FWER is the appropriate criteria.

5.2.1 Volcano Plots

However one chooses to compute the significance values (p-values) of the genes, it is interesting to compare the size of the fold change to the statistical significance level. The "volcano plot" arrange genes along dimensions of biological and statistical significance. The first (horizontal) dimension is the fold change between the two groups (on a log scale, so that up and down regulation appear symmetric), and the second (vertical) axis represents the p–value from the moderated–test on a negative log scale, so smaller p–values appear higher up. The first axis indicates biological impact of the change; the second indicates the statistical evidence, or reliability of the change.

This allows the researcher to make judgements about the most promising candidates for follow-up studies, by trading off both these criteria by eye. With a good interactive program, it is possible to attach names to genes that appear promising.

Figure 21 shows a Volcano Plot for the "Celltypes" example.

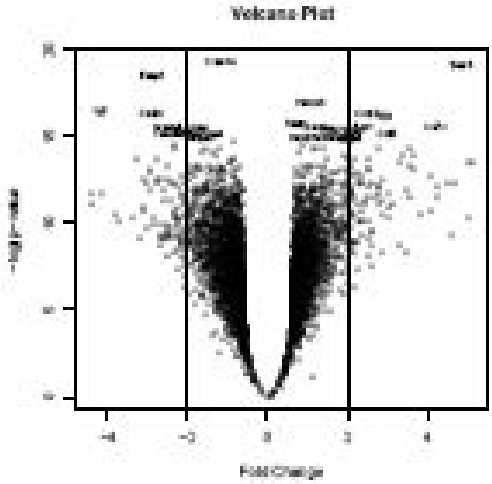

Figure 21: A volcano plot showing the candidates to most differentially expressed genes in the comparison LPS vs Medium in the Celltypes example

5.3 Class Discovery

Clustering, also known as class discovery, is the most popular method currently used in the first step of gene expression matrix analysis to try to identify and group together similarly expressed genes and then try to correlate the results to biology.

The idea is that co–regulated and functionally related genes are probably going to express (go up or down) simultaneously, so they can be grouped into clusters. Also, clustering, much like Principal Components Analysis, reduces the dimensionality of the system and by this, allows easier management of the data set.

Clustering techniques can be applied to construct classifications of arrays (experimental conditions), genes or both together. When they are applied to cluster the genes they can help:

- to identify groups of co–regulated genes,
- to identify spatial or temporal expression patterns,
- to reduce redundancy in prediction models.

If they are used to cluster samples they will be useful:

- to identify new biological classes (i.e. new tumor classes),
- to detect experimental artifacts,
- or for display purposes.

It is usual to cluster simultaneously the rows and columns of the expression matrix (see figure 22).

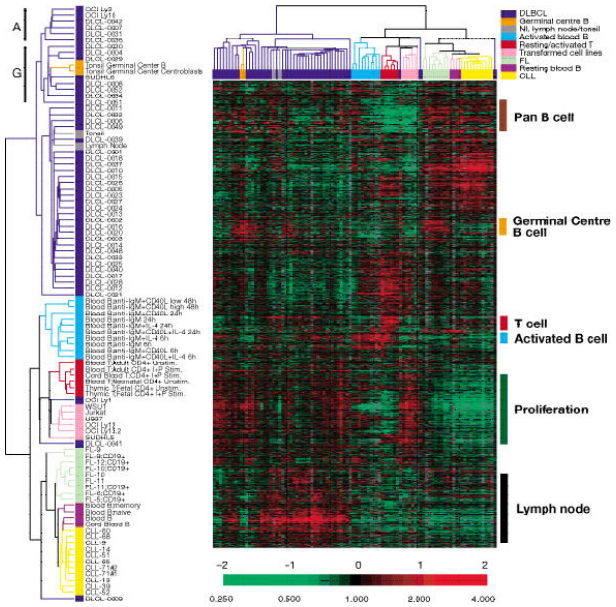

Figure 22: An example of simultaneous clustering of arrays (discovery of related types of tumours) and genes (discovery of co–regulated groups of genes). Source:Alizadeh *et al.* [1]

5.3.1 Algorithms

We will give in this section some characteristics of standard clustering methods in relation to microarray data analysis.

Hierarchical clustering has been mainly used to find a partition of the samples more than of the genes because there are much less samples than genes so that, with genes, the resulting dendrogram is often difficult to interpret. Eisen [27] is the now classical reference on using hierarchichal clustering with microarray data.

A popular display, related to this method, is a color image plot called *heatmap* (see Gentleman *et al.* [30]) which consists of a rectangular array of colored blocks, with the color of each block representing the expression level of one gene on one array (see figure **??**). Typically, in a heatmap, shades of red are used to represent degrees of increasing expression, and shades of green are use to represent degrees of decreasing expression. This is however an arbitrary choice and many other combinations of colors are possible. Each column of boxes represents an array and each row of boxes corresponds to a gene. Heatmaps display

intensities, and can be used independently of clustering. However it is very common to perform a hierarchical clustering of samples and/or genes and to sort the columns and/or rows according to the resulting dendrogram to emphasize the presence of groups.

The *k-means* method (see Kaufman & Rosseeuw [38] is also very popular although it has the disadvantage that it does require specification of a number of clusters and an initial partitioning, what makes the final results to be very sensitive to these choices. In this case the researcher may try different cluster numbers (k) and then pick up the k number that fits best the data. In addition, the resulting groups may change between successive runs because of different initial clusters. K-means and hierarchical clustering share another problem, which is more difficult to overcome, that the produced clustering may be hard to interpret: the order of the genes within a given cluster and the order in which the clusters are plotted do not convey useful biological information. This implies that clusters that are plotted near each other may be less similar than clusters that are plotted far apart.

Other methods such as *Partition Around Medoids* (PAM) and (*Self-Organizing Maps*(SOM) [20] have been applied successfully to microarray data. However each of them has its own drawbacks, and for most users hierarchical clustering keeps being the option of choice.

To end with this section we just mention one algorithm that has been specifically designed for microarray data: the *Hierarchical Ordered Partitioning and Collapsing Hybrid* (HOPACH) (Pollard and van der Laan [52] builds a hierarchical tree of clusters by recursively partitioning a data set, while ordering and possibly collapsing clusters at each level. The algorithm uses the Mean/Median Split Silhouette (MSS) criteria to identify the level of the tree with maximally homogeneous clusters. Then it goes from up to down the hierarchical tree to produce an ordered list of the elements. Finally a non-parametric bootstrap allows one to estimate the probability that each element belongs to each cluster (fuzzy clustering).

5.3.2 Number of clusters

There is an extensive literature on determining the number of clusters in multivariate data. A good review can be found in Milligan and Cooper [49]. Other classical approaches are based on the *Silhouette plot* introduced by Rousseeuw ([56] or the *Average Silhouete Width* where Kaufman and Rousseeuw [38] extended the previous.

Some of these methods have been successfully applied to microarray data. In other cases specific extensions have been developed to better suit their particularities:

- Yeung et al. [70] and Mc Lachlan *et al.* [48] proposed different types of model–based methods.
- Hastie *et al.* [34] introduced the GAP statistic as a measure of tightness to guide cluster number selection.

- Dudoit & Fridlyand, [21] proposed an algorithm called *Clest* wich uses resampling to estimate the number of clusters based on prediction accuracy. The method can be used with *any* partitioning algorithm and seems to be better suited for clustering samples than for clustering genes.

5.3.3 Validation

When one performs a clustering of samples a dendrogram can give insights about the similarity and relatedness among samples, but it does not indicate robustness to variability associated with the sampling process. In order to draw valid conclusions about the clustering structure present in the data, it is necessary to investigate how variability affects the results of the cluster analysis.

Assessing cluster validity is specially important when clustering microarray data. The fact that proteins are organized into pathways and the genes are co–regulated suggests that the expression profiles of a large set of genes are expected to have structure. Thus there is a claim that there are real clusters and they should be discovered.

The difficulty in cluster validation is that there is no initial classification against which the clustering results can be compared. One way to deal with this problem is to examine the relationship between the clustering results and external variables that have not been used previously although this approach is not always possible. However, the partition can produce clusters that are not explainable by this variables.

A common approach to assessing cluster validity is to use some form of resampling such as the bootstrap method developed by Kerr and Churchill ([41]). The *Figure of Merit* (FOM) (Yeung *et al.* ([70]), is another purely experimental approach widely used in other contexts, which has been validated specifically for microarray data. Other authors (Bolshakova *et al.* [10]) give alternative methods to deal with the validation process but additional approaches are still necessary.

5.3.4 The goals of clustering revisited

In this section we have discussed the use of class discovery methods in microarray data analysis. The discussion has been centered about the use of this methodology to find groups of co–regulated genes or related samples.

When one thinks of grouping samples, one usually considers discovering groups related with the process that's being analyzed, e.g. finding that there are distinct types of tumors in what seemed originally one single class.

It has to be noted that this type of discovery is mainly done on the set of genes that have been proved to change in some sense (e.g. differentially expressed genes).

However there is another important application of class discovery, which is performed on all the genes, not only the ones that have been selected. One can cluster the initial (normalized) dataset to discover patterns, probably due to some systematic (block) effect. There can be multiple sources of systematic

variation: production batch, technician, biological source (cell lines) etc. Clustering samples with all the genes followed by an appropriate visualization can help discovering the existence of these effects.

Figure 23 shows a heatmap performed, after a hierarchical clustering, where the main grouping factor is the technician who prepared the arrays, and the second one the cell line used to do the experiment.

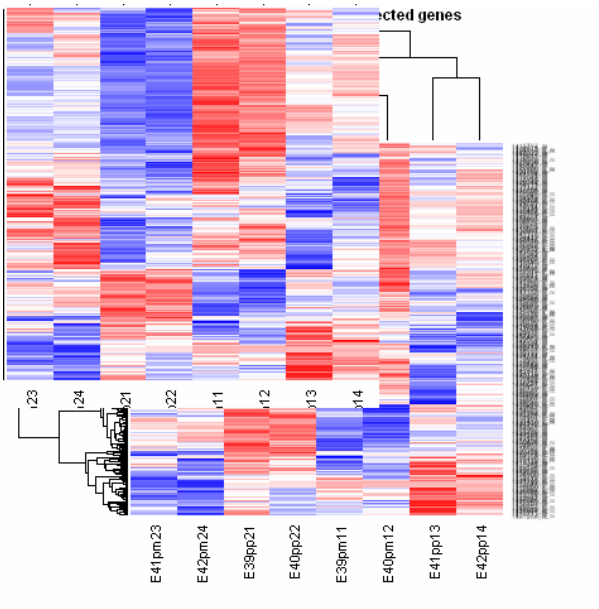

Figure 23: Clustering can show the existence of a batch effect. In this case this is due to differences between technicians (numbers 1,2 first position) whose effect dominate over the differences between cell lines (numbers 1 to 4 second position).

After detecting such unexpected effects it is possible to include them into the model used for detecting differentially expressed genes so that they can be estimated and eventually removed.

5.4 Class Prediction

5.4.1 Overview and goals

The goal of class prediction (in MDA as in most classification problems) is to develop a multivariate function for accurately predicting class membership (phenotype) of a new individual. That is, if each object i is associated with a class label (or response) $Y \in \{1, 2, \ldots, K\}$ and a feature vector of predictor variables of G measurements, $X = (X_1, \ldots, X_G)$, the goal is predicting Y from $\mathbf{X}$ for a new unclassified individual.

From the biomedical point of view it is important to distinguish between *class prediction* –assignment of a new sample to existing categories– and *prognostic prediction* –predicting the progress of a patient's disease. An example of the former can be assigning tumors to one of several predefined types as in was done by Golub *et al.* [32] (see figure 24, a) whereas an example of the later can be building a predictor to determine which tumors may evolve in metastasis after a certain period of time as studied by Van't Veer *et al.* [64] (see figure 24,b).

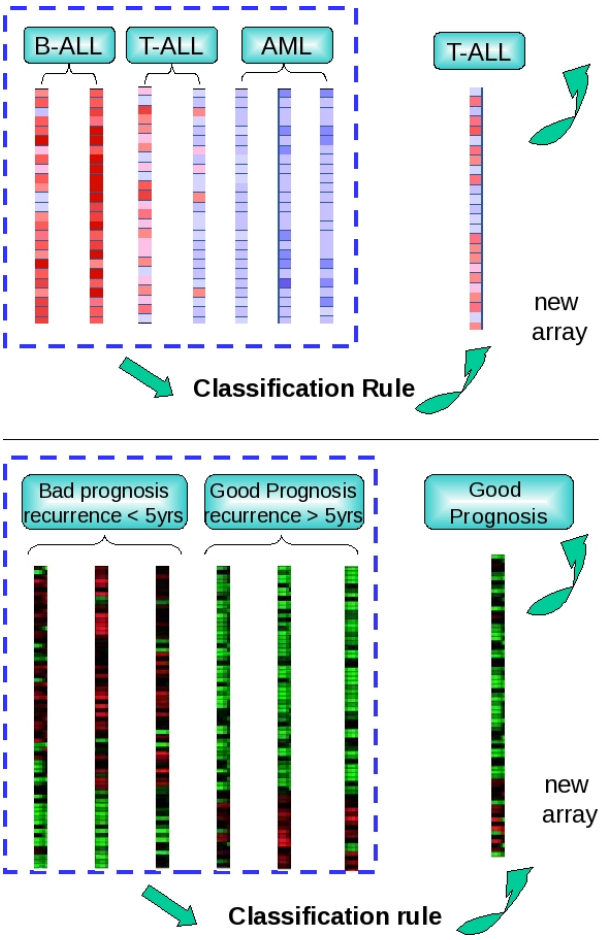

Figure 24: Two examples to illustrate classification problems in microarray data analysis. (a) Class Prediction example: Assignment of tumor type to a new tumor. From [32]. (b) Sources of Variability in Microarray Data

This section is organized as follows: First an enumeration of the main classification methods with emphasis in their application to MDA is presented. After

this the problem of feature selection is discussed and some ideas about measuring the performance of a classifier are given. The section ends considering some specific issues of class prediction with expression data and reproducing some 'practical admonitions" for users developers and practitioners.

This section is heavily based on [18].

5.4.2 Class prediction Methods

The number of available classification methods is very high, probably due to the fact that it is a very general term that may embrace from a simple logistic regression to a complex multi–categorical support vector machine.

One of the most popular methods between statisticians may be *discriminant analysis* [16] which allows to classify binary or multiple outputs using a discriminant function of continuous variables which under normality assumptions may be obtained by maximum likelihood maximization of certain within to between groups sums of squares. Two variants of discriminant analysis have proven to be useful in MDA. One is *Diagonal Linear Discriminant Analysis* which provides optimal discrimination when class densities have the same diagonal variance-covariance matrix. Another is the the weighted voting algorithm introduced by Golub *et al.* ([32]) which has become relatively popular and comes out to be a variant of DLDA ([23])

K Nearest Neighbour methods have also been used, probably due to their simplicity –the group of a test case is predicted as the majority vote among the k nearest neighbors of this test case– and lack of assumptions. Also the fact that the number of neighbors used (k) can be taken as fix (and low) or optimized by cross-validation (as in Barrier *et al.* [6]).

Borrowed from the machine leranining field, rather than classical statistics *Support Vector Machines* have also become very popular as class prediction methods in microarrays. Support vector machines obtain the best separating hyperplane between classes locating this hyperplane so that it has maximal margin (i.e., so that there is maximal distance between the hyperplane and the nearest point of any of the classes). Even when there is no separating hyperplane SVMs can yield decent classifiers by trying to maximize the margin and allow some classification errors subject to the constraint that the total error (distance from the hyperplane in the "wrong side") is less than a constant. The flexibility and versatility of SVM has made them a very popular option between practitioners, but its black–box side, as well as the fact that they are relatively more difficult to understand than simpler approaches has probably restrained its extension.

There are many more methods available, from simple traditional ones, such as logistic regression to more sophisticated modern methods such as Forest Trees, not to talk of all the methods developed *ad hoc* for gene expression data analysis, such as Prediction Analysis for Microarrays (PAM) or Gene Shaving. Good reviews can be found in most microarray data analysis textbooks such as Speed *et al.* ([61]) or Allison *et al.* ([3]).

Class prediction for microarrays, as is the case in other fields has also made

extensive use of aggregation methods, that is the combination of several predictors to obtain improved classifiers. Aggregation was first suggested by Breiman ([12]) who found that gains in accuracy could be obtained by aggregating predictors built from perturbed versions of the learning set. Bagging ([12]) and Boosting ([29]) are two resampling–based aggregator methods that have been applied with relative success to microarrays.

5.4.3 Comparison between methods

Given the number and diversity of available methods one of the first concerns of a potential user of class prediction methods is which one should be used.

To help answer this question Dudoit *et al.* [22] made a comparison of several popular classification methods. Their main conclusion was that simple classifiers such as Diagonal Linear Discriminant Analysis (DLDA) and Nearest–Neighbor (NN) performed remarkably well compared with more sophisticated ones, such as aggregated classification trees.

Won Lee *et al.* [66] extended the previous analysis including more methods (up to 21) and more datasets (7). They reached similar conclusions than Dudoit el al., although they found better performance for more complex methods.

In any case, and whatever the chosen method is, there is a good agreement about the fact that the performance of most methods depends on the set of genes used to build the classifier. This is discussed in next section.

5.4.4 Feature selection

In order to build a predictor one must decide which variables to use. This is not a trivial problem because this selection will guide all the process and the results. If the number of variables is small it is not difficult to choose among them or simply use them all. But having thousands of variables to select from makes it a challenging task.

Some methods, such as SVM, DLDA or KNN can use as many features as desired. Other methods, such as logistic, Cox or multiple regression cannot, mainly due to the curse of dimensionality, $p >> n$, that is the fact that the number of variables (=features=genes) (p) is much greater than the number of samples (n). In any case the use of some procedure for pre–selecting genes is considered to benefit the performance of the predictor.

One first, naif, approach is to rely on those genes that have been called differentially expressed in a previous analysis. This is an intuitive way to proceed but poses a serious drawback: any type of correlation between genes is ignored, which may lead to missing important aspects relevant for the prediction.

The situation described above has been known by the machine learning community for a long time and a great number of methods have been developed to accomplish this goal. These *Feature Selection Algorithms* can be grouped depending on the selection strategy applied (*filter* or *wrapper*) or on the way the features are evaluated (*individual ranking* or *subset* evaluation).

- Filter models rely on general characteristics of the data to evaluate and select gene subsets. For example selecting the top most differentially expressed genes using an ANOVA model is a common filtering strategy.
- Wrapper models require one predetermined mining algorithm and use its performance as the evaluation criterion. They search for features better suited to the mining algorithm, aiming to improve mining performance and they are more computationally expensive than filter models.
- Feature ranking (FR), assesses individual features and assigns them weights according to their degrees of relevance.
- Feature subset selection (FSS) evaluates the goodness of each found feature subset. (Unusually, some search strategies in combination with subset evaluation can provide a ranked list).

Detailed description of these methods is out of the scope of this work but a good review can be found in [44].

Diaz–Uriarte [19] suggests a method for variable selection based on random forests. which not only highlights the possibilities of this approach but also emphasizes the possibly different goals of features selection: either obtaining a –perhaps big– set of genes related to the outcome of interest or a –probably small– set of discriminative genes useful for diagnostic purposes in medical resarch.

5.4.5 Assessment of the classifier's performance

A predictor can always be built from a data set. The important thing in practice is to obtain a good one, (if the "best" predictor is unreachable). In order to establish how good a predictor is one must account for its *discriminability*, that is, how well it predicts unseen data, as well as the *reliability* or robustness of the predictions.

In practice many users rely on some form of error rate to assess the predictor's discriminability, that is on the percentage of bad/good classifications obtained by the predictor.

Any classification rule has to be evaluated for its performance on the future samples. However it is almost never the case in microarray studies that an independent set of samples is available at the time of initial classifier-building phase. This means that one needs to estimate future performance based on what is available: often the same set that is used to build the classifier.

This is in strict contradiction with one well known principle of supervised methodstheÂąat is the data used for evaluating the classifier must be distinct from the data used for selecting the genes and building the predictor. Ignoring this principle may lead to various forms of bias which cause overoptimistic if not simply wrong predictors.

The recomendation of most experts such as Simon *et al.* ([59]) is to integrate in the process of predictor building as many cross-validation steps as needed so

that any potential bias is avoided. In practice it may mean not only cross–validating the error estimating process but also the initial steps of selecting genes for the predictor.

Figures 25 and 26 inspired in those in Dupuy *et al.* ([25]) illustrate two standard approaches to avoid biases when building a classifier.

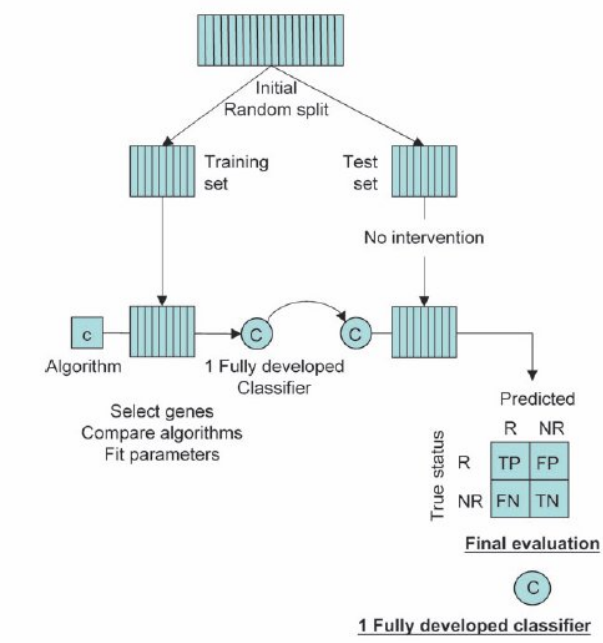

Figure 25:)

Even if the classifier is built in order to avoid possible biases it is generally considered (Dupuy and Simon [25]) that prediction accuracy with its statistical significance alone is insufficient if one is to obtain a complete picture of the classifier's predictive ability and its potential clinical utility. These authors recommend always to present the number of true and false positives and true and false negatives, allowing the calculation of sensitivity and specificity or positive and negative apparent predictive values, or if possibly providing ROC curves as the appropriate guide for performance of a classifier.

5.5 Pathway Analysis

A typical microarray experiment is one who looks for genes *differentially expressed* between two or more conditions. That is, genes which behave differently in one condition (for instance healthy [or untreated or wild-type] cells) than in another (for instance tumor [or treated or mutant] cells). Such an experiment will result very often in long lists of genes which have been selected

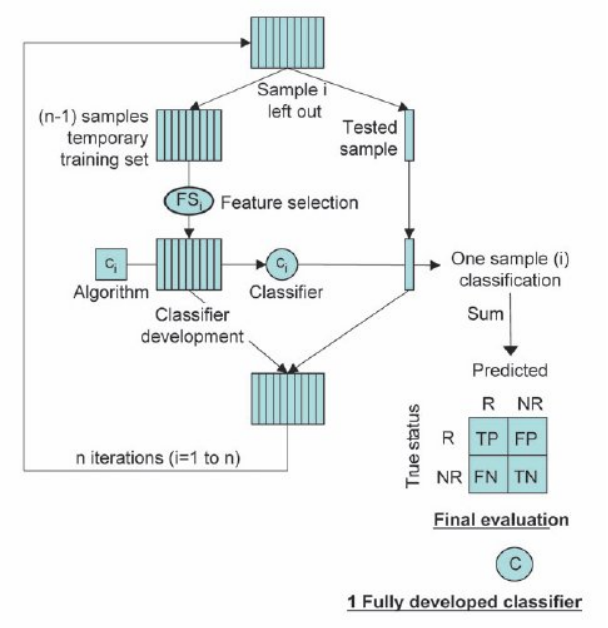

Figure 26:)

using some criteria (think for instance of a moderated t–test followed by p–value adjustment) to assign them *statistical significance*.

With such a list in hand the researcher can move into several, not necessarily excluding, directions. We briefly discuss two of them which are related with the work presented here: (i) Biological interpretation and (ii) Comparison of experiments.

5.5.1 Biological interpretation

A common approach to biological interpretation is to re–process the list trying to relate the genes it contains with one or more functional annotation databases such as the Gene Ontology (GO), Kyoto Encyclopedia of Genes and Genomes (KEGG) or others. There are many methods and models to do this (see Draghici *et al.*, [42] or Mosquera and Sánchez–Pla, [50, 57]) and we briefly discuss the basic structure of two of the most commonly used: *Gene Enrichment Analysis* and *Gene Set Enrichment Analysis*. Gene Enrichment Analysis (GE) aims at establishing if a given category, representing for example a biological process (GO) or a pathway (KEGG), appears more ("enriched") or less ("impoverished") often in the list of selected genes than in the (gene) population from where they have been obtained, i.e., the array, the genome, or simply the genes which were selected for testing. The significance of this potential enrichment/impoverishment is established using a hypergeometric test. The Gene Set Enrichment Analysis

(GSEA) method differs from the previous in that it requires, besides the list of genes, a numerical variable to rank them, usually the p–value of a test for differential expression. Starting from the ranked list a cumulative (enrichment) score based on the presence or absence of each gene in a selected category or 'gene set" is computed. A Kolmogorov–Smirnov test is used to compare the distribution of the scores in the category with the empirical distribution of the numerical variable in the gene list in order to decide if the gene set is over–represented at the top or bottom of the gene list. In spite of the differences between GE and GSEA they also share some traits. One of them is the fact that the tests are performed one category –or one gene set– at a time, followed by a multiple testing adjustment.

5.5.2 Comparison and metaanalysis of microarray experiments

Comparison between microarray experiments is another topic which is receiving increasing attention along with the availability of similar or complementary studies which one may be interested in comparing or combining. In spite of a higher heterogeneity in methods for comparison than in those for biological interpretation one can distinguish different approaches, sketched below. Kupin [43], contains some reflections on different possibilities for comparing microarray experiments.

Some methods for comparing microarrays use the raw data or the lists of selected genes as the basis for quantitative comparison. They rely on some form of statistical reasoning such as similarity scores based on the number of overlapping genes in the top ranks of the lists [68, 47] or the average squared correlation between gene pairs in the data set [58].

Other methods focus on the combination of the experiments more than in their comparison. These can be grouped under the generical term of microarray meta-analysis, [54] although the term meta–analysis is used in this context more liberally than in its standard definition [14].

Last, there are methods that perform functional comparison, that is they base the comparison on functional annotations (e.g. GO categories) associated with the genes in the lists. This is the case of the `eGon` tool, [7] which implements the tests developed by Günther *et al.* [33].

6 Microarray Bioinformatics

The growth in the use of microarrays experienced in the last decade has been parallelled by the necessary developments in methodology –new methods to model and analyze the data were often required– and bioinformatics –new tools were necessary to implement the methods as well as to store, to access or to organize the increasing bulk of available data. This takes us to consider two important aspects very related with microarray data analysis:

1. Which software is there available to analyze microarray data?

2. Which database systems are there available to store and manage microarray data either at the local or at the global level?

This topic can be considered complementary but necessary to implement the points discussed in the paper so that a brief presentation of existing software and database systems will be presented below.

6.1 Software for microarray data analysis

Assume that a statistician wants to get involved in analyzing microarray data and after some reading she understands what is to be done. An obvious question is "which tool should I use"? As most professionals in the field she is familiar with several packages and probably has some preferences

After some google searching it becomes obvious that there are several possibilities

- To use standard statistical packages –SPSS or SAS– and analyze data whch must have been preprocessed and exported to text–delimited files
- To use one of the many freely available tools, either web or locally based.
- To rely on extensions specific for for microarray data analysis such as the Bioconductor Project.
- To buy one of the existing commercial programs.

As usual every option has positive and negative aspects. Using standard statistical packages –SPSS or SAS– has the shortest learning curve, but does not allow to make most of the pre–processing steps such as normalization or summarization, so it must be combined with other software. Besides, if one wishes to do an ANOVA or a K–means they are fine, but if what one wants to do is to apply specific methods such as SAM or local–FDR adjustments they will quickly prove unsufficient. Some Statistical packages such as `S+` or `SAS` have developed powerful extensions for microarray data analysis

6.1.1 Open source software

"Free tools cost no money", but it is less clear that they cost no time. There are dozens of freely available tools, either web or locally based (see `http://www.nslij-genetics.org/microarray/soft.html`) for a classification. The problem however is that they are completely unstandardized so that learning one does not usually help in learning next and, as freee tools, they can present a higher rate of errors than desired. It is often the case that these tools are useful for "toy analysis" or for teaching but if one wishes to use them for repeatedly performing studies of mid to high complexity most of them prove to be unsufficient, either because they lack methods, they are unefficient or simply because they do not have programming capabilities to automate repetitive tasks.

In spite of these criticisms free programs may be a soft way to introduce oneself to microarray data analysis. To guide an unexperienced user we make a short, biased, comment of some of our favourite free tools.

- *BRB array tools* is an Excel add-ins which combines `R`, `C` and Java to do the calculations and uses Excel to interface with the user –which means it is only available for windows users. It is provided by The Biometrics Research Branch of the National Cancer Institute (USA). it is complemented with complete tutorials and a a database of real studies prepared to be used with it. It happens to be very attractive at first sight specially when used with its own examples. However creating a new analysis from the beginning is not an easy task and what is worst it tends to crash in a hard–to–recover manner with criptic Visual Basic messages, specially if used in computers with non-english versions of windows.

- *TM4* is a suite of four free programs written in Java and running in Linux and Windows systems developed by the TIGR (now J.Craig Venter) institute. Albeit a little old and relatively biased towards two–colour arrays, for which it was originally developed, it is very robust (crashes much less than BRB) and offers not only analysis capabilities (`MeV`) but also image analysis (`Spotfinder`), separate normalization (`MIDAS`) and a database system (`MADAM`) to store experiments.

- One serious drawback of the previous tools is their historical bias towards two-colour microarrays which implies that they miss (as of beginning 2008) important preprocessing methods such as `RMA`. A good –easy to use– alternative for the first steps of quality check and preprocessing of affymetrix chips is offered by the Company. Its is called *Expression Console* and can be downloaded from Affymetrix Web site after free registration.

- *The http://gepas.bioinfo.cipf.es/* is an integrated packages of tools for microarray data analysis available over the web. GEPAS has been designed to provide an intuitive web-based interface that offers diverse analysis options from the early step of preprocessing (normalization of Affymetrix and two-colour microarray experiments and other preprocessing options), to the final step of the functional profiling of the experiment (using Gene Ontology, pathways, PubMed abstracts etc.), which include different possibilities for clustering, gene selection, class prediction and array-comparative genomic hybridization management. Figure 27 shows in a graphical manner a map of GEPAS functionalities as a subway line.

6.1.2 The Bioconductor Project

One of the options for data analysis mentioned above is to combine some standard software such as `Matlab`, `Mathematica` or `R` with specific libraries designed for microarray analysis. Although some extensions exist for Matlab (see eg `http://ihome.cuhk.edu.hk/~b400559/arraysoft_matlab_mfiles.html`) it's

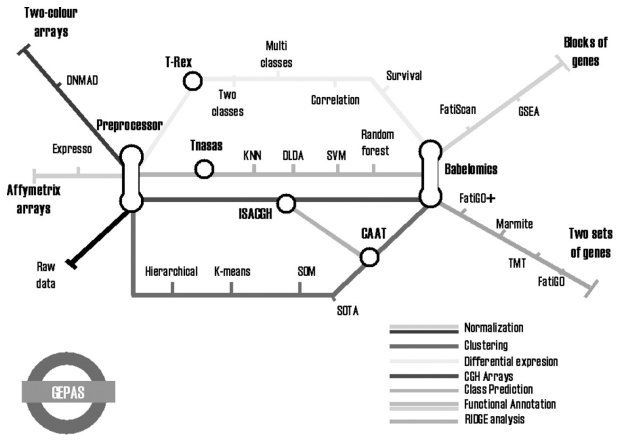

Figure 27: A map of GEPAS functionalities organized as in a subway line. A user should usually start somewhere in the left of the map and end somewhere in the right

with R that this complementarity has reached unexpected dimension. The Bioconductor Project (http://www.Bioconductor.org) started in 2001 as an open source and open development software project for the analysis and comprehension of genomic data. Its great success has made it grow from hardly more than a dozen packages to hundreds of them. Almost every technique available in microarray analysis has its own package, and there are often several of them.

The great power of this project also entails some of its drawbacks: First, being an open source project means that developers contribute their programs "as is". Although there are checking systems to avoid non–running code, it is harder to guarantee (apart of the honestity of the developers) that it runs as indicated. The power of Bioconductor is also based on the flexibility of the R language. It is very hard for users who are non–proficient in Rto make efficient use of these libraries.

In spite of these apparent difficulties Bioconductor is the chosen tool for many statisticians and the main reason is that, when one has been able to feel confortable using it, its power is hard to equate. The programming facilities of R, malke it possible to automate analysis as well as report generation, making it the option of choice when repetitive tyasks have to be performed.

6.1.3 Proprietary software

There are many commercial tools available for microarray data analysis. These range from small programs specific of one data type to big software suites, such as Partek Genomics Suite which is a complete solution optimized for efficient and fast computations as well as for most existing genomic data. Commercial microarray software has the traditional pros and cons of any commercial soft-

ware: It may be good, but it is expensive and it may not be flexible enoughfor the expert user who wishes to introduce its own methods in the analysis

6.2 Microarray databases

The diversity of microarray formats and types of experiments has made it difficult that a any database format has imposed and no database system has emerged as the "gold–standard".

Indeed there as been some agreement on the minimum information about a microarray experiment that needs to be stored (the MIAME standard (http://www.mged.org/Workgroups/MIAME/miame.html) is an acronym for this), but as if it were a political topic the agreement has been so short that it is more symbolic than useful.

One can distinguish two levels at which databases systems have been developed.

1. *Local database systems* The analysis of microarray data goes through a series of steps where different types of data, images, binaries, text files have to be processsed. It requires to have them stored in an easily–accessible way. Some systems such as BASE (http://base.thep.lu.se/) or caArray (http://caarray.nci.nih.gov/) are powerful solutions for storing data and experiments but their use is far from being so extended as that of analysis software tools.

2. *Public array repositories* The biological community has agreed, from the beginning of microarrays, that data from published experiments should be made publicly available. This has created the need for public microarray repositories where any user could store their data in a suitable form. At the same time it has made an impressive quantity of data available for re-analysis by anyone who wishes to do it, offering an unparalleled wealth of opportunities whose power is just starting to show. A list of public data collections is available at http://www.nslij-genetics.org/microarray/data.html

7 Extensions And Perspectives

This article has been centered, around the most popular type of microarrays: DNA expression microarrays, that is, tools designed to study gene expression based on information about the quantity of DNA being transcribed as RNA.

The availability of genome technologies has allowed to develop other types of microarrays. By "other" one may mean microarrays that rely on DNA to study other problems than expression or microarrays which rely on other substances such as protein or carbohydrates. A full description of each type, its use, goals and data analysis is absolutely out of the scope of this work. However to give an example of similarities and differences between expression microarrays and related technologies we make give a brief review of the problems that require of

these alternative technologies and give a brief description of one of them: SNP arrays.

7.1 Different microarrays to answer different questions

One of the main focus of functional genomics is towards the understanding and cure of disease. It is known that many genetic alterations underlie abnormalities and/or diseases. For example:

- "Point" mutations –change of one or a few bases– may lead to altered protein or change in expression level.
- Loss of gene copies may reduce expression level. These changes are related to tumor suppression.
- Gain of gene copies may increase expression level and they are with related oncogene activation.
- Methylation or de–methylation of gene promoters may respectively decrease or increase expression level. These are also related to oncogene tumor suppressors.
- Breaking and abnormal rejoining of DNA makes novel genes.

Different types of microarrays are tailored to study the manifestations and effects of these alterations. The points raised above may be studied with (i) *genotyping* or SNP (spell "sneep") and (ii) *comparative genome hybridization* or CGH DNA microarrays and others such as *Methylation*, *Promoter* or *Tiling* arrays.

7.1.1 Genotyping or SNP arrays

Single Nucleotide Polymorphism are a form of point mutation consisting in variations in single base pairs that are randomly dispersed throughout the genome. Thousands of Single Nucleotide Polymorphisms have been -and continue being– identified as part of Genome Sequencing projects. SNPs have been highly conserved throughout evolution and within a population. Due to this conservation the map of SNPs serves as an excellent genotypic marker for research.

SNP arrays are a type of DNA chips used to detect polymorphism inside populations. They work under the same basic principles as expression arrays but each probe is designed to detect the different variations of single nucleotide polymorphisms for each known SNP.

Figure 28 depicts in a simplified manner how to use SNP arrays to detect polymorphism.

SNP arrays have many applications. Between them one may highlight:

- **Family-based linkage studies** DNA from family members affected with a particular condition may be compared with DNA from members of the same family who do not have the condition. These studies, allow to identify genetic differences which may be associated with the condition.

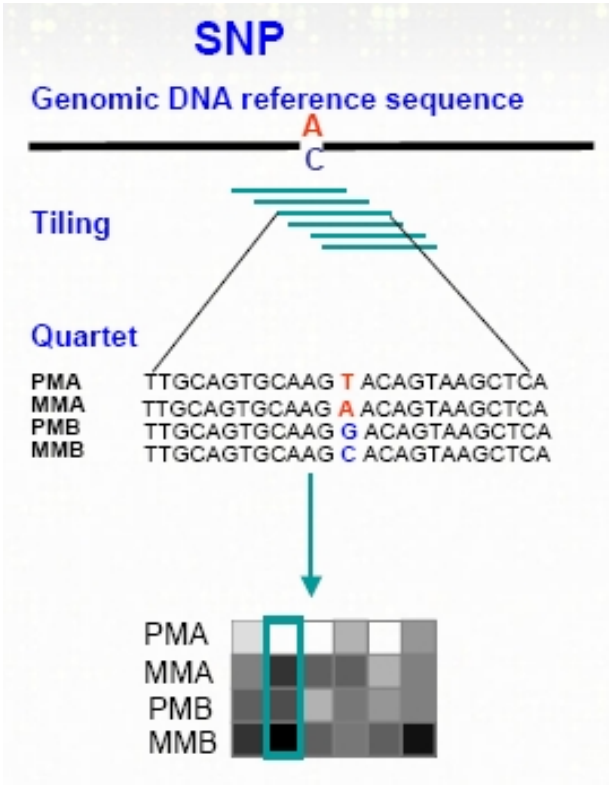

Figure 28: Simplified explanation of the use of SNP arrays to detect Single Nucleotide Polymorphisms

- **Population-based association studies** consist of determining differences in SNP frequencies in affected and unaffected individuals in a population. The aim is to identify particular SNPs or SNP combinations which differ between the two groups and are therefore associated with the disease. These studies require a large numbers of samples to adequately represent the population. This is one of the best–known application of SNPs arrays which illustrates how they can halp in the identification of genes related to complex disorders.

- **Copy number changes** SNPs can be used as tags for regions of copy number variability A copy number variant (CNV) is "a DNA segment that is 1kb or larger and is present at variable copy number in comparison with a reference genome". Identification of copy number changes is useful for detecting both chromosomal aberrations and copy number neutral loss of heterozygosity (LOH), events which are characteristic of many types of cancer.

7.2 Non-DNA microarrays

There is a wide consensus about the fact that information obtained from DNA microarrays is not enough to reach a complete understanding of cellular processes most of which are controlled by proteins which often interact with other molecules such as carbohydrates often involved in important biological mechanisms such as host–pathogen interaction, development or inflammation. Protein (i) and Carbohydrate (ii) microarrays are two examples of extension of using these tools for high throughput analysis of different types of molecules. Tissue microarrays (iii) are a different type of extension where the substract is not different variants of a single type of molecule but of a type of tissues.

8 Discussion and Conclusions

This article has presented the technology of DNA expression microarrays and has discussed how to analyze the data it generates. Microarray data analysis has a short history of hardly more than 10 years. But the fast technological development has allowed that, after a start–up period where microarrays were unreliable, expensive devices, they became more precise and affordable. In parallel to this process, studies have turned from using few or even one sample per condition, to using more reasonable designs, with a bigger number of replicates. This offered a golden opportunity for statistics and statisticians to enter massively in this field.

It is interesting to notice that the field of microarray is one of these few with the particularity that almost all statistical techniques may be used at some point of an analysis. A first obvious consequence is that people working in microarray data analysis need a high, or a wide, statistical background (for example a statistician).

Many aspects of "classical" statistics –experimental design, multivariate analysis– can be directly applied to microarray studies. In other cases –when the sample size is small or classical assumptions do not hold– techniques developed especially for these data types are preferable.

This highlights the feedback that has appeared between statistics and bioinformatics whose problems have raised opportunities to develop new statistical methodologies. It is not unusual, as of year 2008, to see that a statistics journal –such as Biometrics, JASA or Biostatistics– has a high percentage of articles devoted to these types of problems. Also, high impact journals such as Bioinformatics (ranked the number one between Statistics journals) have become a common place to –try to– publish for statisticians.

We can note that the relation between microarray data analysis and statistics has reached a maturity where the need for or the relevance of statistics is not discussed. One can even dare to say that this part has become "classic", and now statisticians integrated in interdisciplinary teams are already looking at new problems and new data types generated by modern molecular biology. Discussing them is out of the scope of this paper, but to say just a few things some of the challenges posed deal with the integration of different data types and different studies as a part of more general approach to understanding biological systems ("systems biology").

8.1 Concluding remarks

The previous discussion suggests the existence of a strong relation and cooperation between statisticians and life-scientists. This may be true in some countries, but in many others it is far from being the current situation.

There is a real need for statisticians who want to become involved in this field. There are many open problems and opportunities, not only to publish but also to find jobs.

To activate this process the implication of all actors is necessary: Research institutes must ask for statisticians in their job offers, without confusing them with bioinformaticians, who have a complementary but different role. Universities must offer a modern training integrating bioinformatics and biostatistics in mixed curricula. Last scientific societies also play a role. They should promote discussion within and between them so that what constitutes a real opportunity is not lost.

References

[1] A. Alizadeh, M.B. Eisen, E. Davis, C. Ma, I. Lossos, A. Rosenwald, J. Boldrick, H. Sabet, T. Tran, X. Yu, J.I. Powell, L. Yang, G.E. Marti, J. Hudson Jr, L. Lu, D.B. Lewis, R. Tibshirani, G. Sherlock, W.C. Chan, T.C. Greiner, D.D. Weisenburger, J.O. Armitage, R. Warnke, R. Levy, W. Wilson, M.R. Grever, J.C. Byrd, D. Botstein, P.O. Brown, and L.M.

Staudt. Distinct types of diffuse large B–cell lymphoma identified by gene expression profiling. *Nature*, 403:503–511, February 2000.

[2] David B Allison, Xiangqin Cui, Grier P Page, and Mahyar Sabripour. Microarray data analysis: from disarray to consolidation and consensus. *Nat Rev Genet*, 7(1):55–65, January 2006.

[3] D.B. Allison. *DNA Microarrays and Related Genomics Techniques: Design, Analysis, and Interpretation of Experiments*. CRC Press, 2006.

[4] Helen Parkinson Thomas Schlitt Mohammadreza Shojatalab Alvis Brazma. A quick introduction to elements of biology - cells, molecules, genes, functional genomics, microarrays.

[5] J C Alwine, D J Kemp, and G R Stark. Method for detection of specific rnas in agarose gels by transfer to diazobenzyloxymethyl-paper and hybridization with dna probes., December 1977.

[6] Alain Barrier, Pierre-Yves Boelle, Antoinette Lemoine, Antoine Flahault, Sandrine Dudoit, and Michel Huguier. [gene expression profiling in colon cancer]. *Bull Acad Natl Med*, 191(6):1091–101; discussion 1102–3, June 2007.

[7] Vidar Beisvag, Frode K R Jünge, Hallgeir Bergum, Lars Jølsum, Stian Lydersen, Clara-Cecilie Günther, Heri Ramampiaro, Mette Langaas, Arne K Sandvik, and Astrid Laegreid. Genetools–application for functional annotation and statistical hypothesis testing. *BMC Bioinformatics*, 7:470, 2006.

[8] N. Biotechnology. The MicroArray Quality Control (MAQC) project shows inter-and intraplatform reproducibility of gene expression measurements. *Nature Biotechnology*, 24:1151–1161, 2006.

[9] M Bittner, P Meltzer, Y Chen, Y Jiang, E Seftor, M Hendrix, M Radmacher, R Simon, Z Yakhini, A Ben-Dor, N Sampas, E Dougherty, E Wang, F Marincola, C Gooden, J Lueders, A Glatfelter, P Pollock, J Carpten, E Gillanders, D Leja, K Dietrich, C Beaudry, M Berens, D Alberts, and V Sondak. Molecular classification of cutaneous malignant melanoma by gene expression profiling., August 2000.

[10] N. Bolshakova and F. Azuaje. Cluster validation techniques for genome expression data. *Signal Processing*, 83(4):825–833, 2003.

[11] BM Bolstad, RA Irizarry, M. Astrand, and TP Speed. A comparison of normalization methods for high density oligonucleotide array data based on variance and bias, 2003.

[12] L. Breiman. Bagging Predictors. *Machine Learning*, 24(2):123–140, 1996.

[13] M.J. Callow, S. Dudoit, E.L. Gong, T.P. Speed, and E.M. Rubin. Microarray Expression Profiling Identifies Genes with Altered Expression in HDL-Deficient Mice. *Genome Research*, 2000.

[14] J.B. Carlin and T. Normand. Tutorial in biostatistics. meta-analysis: formulating, evaluating, combining, and reporting. *Stat Med*, 19(5):753–9, March 2000.

[15] R.L. Chelvarajan, Y. Liu, D. Popa, M.L. Getchell, T.V. Getchell, A.J. Stromberg, and S. Bondada. Molecular basis of age-associated cytokine dysregulation in LPS-stimulated macrophages. *Journal of Leukocyte Biology*, 79(6):1314, 2006.

[16] Carles M. Cuadras. *Análisis Multivariante*. EUNIBAR, 1989.

[17] X. Cui, J. T. Hwang, J. Qiu, N. J. Blades, and G. A. Churchill. Improved statistical tests for differential gene expression by shrinking variance components estimates. *Biostatistics*, 6:59–75, 2005.

[18] R. Dıá. Supervised Methods with Genomic Data: a Review and Cautionary View. *Data analysis and visualization in genomics and proteomics. New York: Wiley*, pages 193–214, 2005.

[19] Ramón Díaz-Uriarte and Sara Alvarez de Andrés. Gene selection and classification of microarray data using random forest. *BMC Bioinformatics*, 7:3, 2006.

[20] R. Duda, P. Hart, and DG. Stork. *Pattern recognition, 2nd. Ed.* John Wiley and Sons, 2001.

[21] S. Dudoit and J. Fridlyand. A prediction-based resampling method for estimating the number of clusters in a dataset. *Genome Biology*, 3(7):1–21, 2002.

[22] S. Dudoit, J. Fridlyand, and T. P. Speed. Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American Statistical Association*, 97(457), 2002.

[23] S. Dudoit, J. P. Shaffer, and J. C. Boldrick. Multiple hypothesis testing in microarray experiments. *Statistical Science*, 18:71–103, 2003.

[24] S. Dudoit, Y. H. Yang, M. J. Callow, and T. P. Speed. Statistical methods for identifying differentially expressed genes in replicated cDNA microarray experiments. *Statistica Sinica*, 12(1), 2002.

[25] A. Dupuy and R.M. Simon. Critical Review of Published Microarray Studies for Cancer Outcome and Guidelines on Statistical Analysis and Reporting. *JNCI Journal of the National Cancer Institute*, 99(2):147, 2007.

[26] L. Dyrskjøt, T. Thykjaer, M. Kruhøffer, J.L. Jensen, N. Marcussen, S. Hamilton-Dutoit, H. Wolf, and T.F. Ørntoft. Identifying distinct classes of bladder carcinoma using microarrays. *Nature Genetics*, 33:90–96, 2002.

[27] M. B. Eisen, P. T. Spellman, P. O. Brownand, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences USA*, 95(25):14863–14868, 1998.

[28] J.M. Freudenberg. Comparison of background correction and normalization procedures for high-density oligonucleotide microarrays. *Institut fur Informatik*, 2005.

[29] Y. Freund and R. Schapire. Experiments with a new boosting algorithm, in âĂ Machine Learning: Proceedings of the Thirteenth International ConferenceâĂŹ. *Morgan Kauffman, San Francisco*, pages 148–156, 1996.

[30] Robert Gentleman, Vince Carey, Wolfgang Huber, Rafael Irizarry, and Sandrine Dudoit. *Bioinformatics and Computational Biology Solutions using R and Bioconductor*. Springer, New York, 2005.

[31] D.H. Geschwind and J.P. Gregg. *Microarrays for the neurosciences: an essential guide*. MIT Press, 2002.

[32] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M.L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, and E. S. Lander. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 286:531–537, 1999.

[33] Clara-Cecilie Günther, Mette Langaas, and Stian Lydersen. Statistical hyhpothesis tesing of association between two lists of genes for a given gene class. Technical Report 1, Norwegian Institution of Science and Technology, 2006.

[34] T. Hastie, R. Tibshirani, and G. Walther. Estimating the number of clusters in a dataset via the gap statistic. *Journal of the Royal Statistical Society, B*, 63(41):1–423, 2001.

[35] A.A. Hill, E.L. Brown, M.Z. Whitley, G. Tucker-Kellogg, C.P. Hunter, and D.K. Slonim. Evaluation of normalization procedures for oligonucleotide array data based on spiked cRNA controls. *Genome Biol*, 2(12):1–0055, 2001.

[36] W. Huber, A. von Heydebreck, H. Sültmann, A. Poustka, and M. Vingron. Variance stabilization applied to microarray data calibration and to the quantification of differential expression. *Bioinformatics*, 18 Suppl. 1:S96–S104, 2002.

[37] R. A. Irizarry, B. Hobbs, F. Collin, Y. D. Beazer-Barclay, K. J. Antonellis, U. Scherf, and T. P. Speed. Exploration, normalization, and summaries of high density oligonucleotide array probe level data. *Biostatistics*, 4:249–264, 2003.

[38] L. Kaufman and P.J. Rousseeuw. *Finding Groups in Data. An Introduction to Cluster Analysis*. John Wiley and Sons, 1990.

[39] M K Kerr and G A Churchill. Experimental design for gene expression microarrays., June 2001.

[40] M Kathleen Kerr. Design considerations for efficient and effective microarray studies., December 2003.

[41] M.K. Kerr and G.A. Churchill. Bootstrapping cluster analysis: Assessing the reliability of conclusions from microarray experiments. *Proceedings of the National Academy of Sciences*, page 161273698, 2001.

[42] P. Khatri and S. Drăghici. Ontological analysis of gene expression data: current tools, limitations, and problems. *Bioinformatics*, 18:3587–3595, 2005.

[43] Isabelle Lesur Kupin. *Study of the Transcriptome of the prematurely aging* dna-2 *yeast mutant using a new system allowing comparative DNA microarray analysis*. PhD thesis, Universite Bordeaux I, April 2005.

[44] P. Larrañaga, B. Calvo, R. Santana, C. Bielza, J. Galdiano, I. Inza, J.A. Lozano, R. Armañanzas, G. Santafï£¡, A. Perez, and V. Robles. Machine learning in bioinformatics. *Briefings in Bioinformatics*, 7(1):86–112, 2006.

[45] M.L.T. Lee and GA Whitmore. Power and sample size for DNA microarray studies. *Statistics in Medicine*, 21(23):3543–3570, 2002.

[46] R J Lipshutz, S P Fodor, T R Gingeras, and D J Lockhart. High density synthetic oligonucleotide arrays., January 1999.

[47] Claudio Lottaz, Xinan Yang, Stefanie Scheid, and Rainer Spang. Orderedlist–a bioconductor package for detecting similarity in ordered gene lists. *Bioinformatics*, 22(18):2315–6, September 2006.

[48] GJ McLachlan, RW Bean, and D. Peel. A mixture model=-based approach to the clustering of microarray expression data, 2002.

[49] G.W. Milligan and M.C. Cooper. An examination of procedures for determining the number of clusters in a data set. *Psychometrika*, 50(2):159–179, 1985.

[50] J-L. Mosquera and A. Sánchez-Pla. A comparative study of go mining programs. In *X Conferencia Española de Biometría*. Sociedad Española de Biometría, 2005.

[51] W. Pan. A comparative review of statistical methods for discovering differentially expressed genes in replicated microarray experiments. *Bioinformatics*, 18(4):546–554, 2002.

[52] KS Pollard and MJ van der Laan. Cluster analysis of genomic data. *Bioinformatics and Computational Biology Solutions Using R and Bioconductor. New York: Springer*, pages 209–228, 2005.

[53] J. Quackenbush. Microarray data normalization and transformation. *Nature Genet.*, 32:496Ű–501, 2002.

[54] Daniel R Rhodes, Terrence R Barrette, Mark A Rubin, Debashis Ghosh, and Arul M Chinnaiyan. Meta-analysis of microarrays: interstudy validation of gene expression profiles reveals pathway dysregulation in prostate cancer. *Cancer Res*, 62(15):4427–33, August 2002.

[55] Matthew E Ritchie, Jeremy Silver, Alicia Oshlack, Melissa Holmes, Dileepa Diyagama, Andrew Holloway, and Gordon K Smyth. A comparison of background correction methods for two-colour microarrays. *Bioinformatics*, 23(20):2700–7, October 2007.

[56] P. Rousseeuw, E. Trauwaert, and L. Kaufman. Some silhouette-based graphics for clustering interpretation. *Belgian Journal of Operations Research, Statistics and Computer Science*, 29(3):35–55, 1989.

[57] A. Sánchez-Pla and J.L Mosquera. The quest for biological significance. In L.L. Bonilla, M. Moscoso, G. Platero, and J.M. Vega, editors, *Progress in Industrial Mathematics at ECMI 2006*. Springer, New York, 2007.

[58] Kerby Shedden. Confidence levels for the comparison of microarray experiments. *Stat Appl Genet Mol Biol*, 3:Article32, 2004.

[59] Richard M. Simon, Edward L. Korn, Lisa M. McShane, Michael D. Radmacher, George W. Wright, and Yingdong Zhao. *Design and Analysis of DNA Microarray Investigations*. Springer-Verlag, 2003.

[60] Gordon K Smyth, Joëlle Michaud, and Hamish S Scott. Use of within-array replicate spots for assessing differential expression in microarray experiments. *Bioinformatics*, 21(9):2067–75, May 2005.

[61] T. Speed. *Statistical Analysis of Gene Expression Data*. Boca Raton, Fla.: Chapman & Hall/CRC, 2003.

[62] R. Tibshirani. A simple method for assessing sample sizes in microarray experiments. *BMC Bioinformatics*, 7(1):106, 2006.

[63] V. Tusher, R. Tibshirani, and C Chu. Significance analysis of microarrays applied to ionizing radiation response. *Proceedings of the National Academy of Sciences USA*, 98:5116–5121, 2001.

[64] Laura J van 't Veer, Hongyue Dai, Marc J van de Vijver, Yudong D He, Augustinus A M Hart, Mao Mao, Hans L Peterse, Karin van der Kooy, Matthew J Marton, Anke T Witteveen, George J Schreiber, Ron M Kerkhoven, Chris Roberts, Peter S Linsley, René Bernards, and Stephen H Friend. Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, 415(6871):530–6, January 2002.

[65] E. Witt and John. McClure. *Statistics for Microarrays: Design, Analysis and Inference*. John Wiley & Sons, 2004.

[66] J. Won Lee, J. Bok Lee, M. Park, and S. Song. An extensive comparison of recent classification tools applied to microarray data. *Computational Statistics & Data Analysis*, 48(4):869–885, 2005.

[67] H. Wu, M.K. Kerr, X. Cui, and G.A. Churchill. MAANOVA: a software package for the analysis of spotted cDNA microarray experiments. *The Analysis of Gene Expression Data: Methods and Software*, pages 313–341, 2003.

[68] Xinan Yang, Stefan Bentink, Stefanie Scheid, and Rainer Spang. Similarities of ordered gene lists. *J Bioinform Comput Biol*, 4(3):693–708, June 2006.

[69] Y. H. Yang, M. J. Buckley, S. Dudoit, and T. P. Speed. Comparison of methods for image analysis on cDNA microarray data. *Journal of Computational and Graphical Statistic*, 11(1), 2002.

[70] KY Yeung, C. Fraley, A. Murua, AE Raftery, and WL Ruzzo. Model-based clustering and data transformations for gene expression data, 2001.