Doctoral thesis

Statistical methods for Time Course Microarray data

Author: María José Nueda Roldán

Supervisors: Ana V. Conesa Cegarra
Alberto J. Ferrer Riquelme

UNIVERSIDAD POLITECNICA DE VALENCIA

Departamento de Estadística e Investigación
Operativa Aplicadas y Calidad
Valencia, 2009

Abstract

The present thesis addresses the statistical analysis of single and multiple series Time Course Microarray (TCM) data. This type of data comes from studies in which gene expression evolution is analysed throughout time for one or several experimental conditions of interest. The work describes the development, application and evaluation of novel specific methodologies which take into consideration particular aspects and problematic that this type of data causes, both from a gene selection and from a functional point of views. The developed algorithms are compared to other state-of-the-art methodologies, evaluating the different approaches in terms of performance and biological meaning of the results.

The thesis has been structured in two main blocks. First, the relevant literature is revised and summarized in an introductory part. A general overview of microarray technology and a discussed review of statistical methods applied to microarray data are presented in Chapters 1 and 2. By using data from a multifactor microarray experiment we show how the application of general methods to time series microarray data suffers from a number of limitations. This indicates the need for the development of specific methods for the analysis of TCM. Chapter 3 ends this first block with a dedicated review on the up-to-date statistical methods for the analysis of time series data. Most of the methodologies presented in this chapter have been published during the time span of this thesis.

In the second block, the novel methodologies for TCM developed within the present research are introduced and discussed. Chapter 4 introduces the first novel approach to TCM analysis: **maSigPro** (microarray Significant Profile) methodology. The maSigPro strategy uses linear regression analysis to model gene expression and follows a two-step strategy for the selection of differentially expressed genes (d.e.g): a first step identifies responsive genes while the second discloses the patterns of significant differential time evolution, in a gene-by-gene fashion. In Chapter 5 a multivariate technique ASCA (ANOVA Simultaneous Component Analysis) is adapted to TCM, resulting in the **ASCA-genes** method. This new methodology combines multivariate exploration of time course data with a selection procedure for the identification of relevant changing genes. In Chapter 6 the ability of ASCA to dissect expression signals and exploit the coordinative behaviour of gene expression is combined with the strong ability of maSigPro to model time series data and identify significant d.e.g. Our results show that, especially when high structural noise is present in the data, the use of **ASCA** as **pre-processing** strategy greatly improves maSigPro results. We also demonstrated that this data filtering strategy can be applied as well to other methods of TCM analysis improving false and negative discovery rates. These approaches, as others in microarray time series data analysis, provide results as lists of differentially expressed genes. However, in the study of gene expression, a much more interpretable and useful result appears when gene regulation is indicated as cellular functions or processes. In most cases, this translation is done subsequent to the generation of a list of differentially expressed genes. This implies in many cases limitations in discovery power for the need of an arbitrary calling of d.e.g. and the ignorance of the coordination between biological functions. The last methodological chapter of this thesis (Chapter 7) deals with the development of statistical approaches for an integrated or direct assessment of the alterations in gene functions embedded in time series expression data. To this end, maSigPro, ASCA and PCA have been adapted to incorporate functional data resulting in the novel methodologies **maSigFun**, **PCA-maSigFun** and **ASCA-functional**.

This dissertation is ended with Chapter 8 which includes conclusions and some proposals deserving future research.

Resumen

La presente tesis doctoral aborda el análisis estadístico de series simples y múltiples de experimentos de "Time Course Microarray" (TCM). El trabajo se centra en el desarrollo, aplicación y evaluación de métodos estadísticos específicos que consideran la problemática de este tipo de datos, tanto desde el punto de vista de selección de genes como del análisis funcional. Las técnicas desarrolladas se comparan con otros métodos del estado del arte actual evaluando las diferentes metodologías en términos de eficiencia y significado biológico de los resultados.

La tesis está estructurada en dos bloques principales. En el primero, se revisa la literatura relevante y se resume en una parte introductoria. Los Capítulos 1 y 2 incluyen la descripción del funcionamiento de la tecnología de "microarrays" así como una revisión crítica de los métodos estadísticos aplicados a este tipo de datos. En esta parte se muestran los inconvenientes que surgen al aplicar métodos generales a series temporales de "microarrays" y se justifica la necesidad de desarrollar nuevas técnicas para el análisis de TCM. El Capítulo 3 finaliza este primer bloque con una revisión de los métodos estadísticos específicos para TCM. Muchas de las técnicas que se presentan en este capítulo han sido publicadas en el mismo período de elaboración de esta tesis.

En el Segundo bloque, se presentan las técnicas estadísticas para TCM desarrolladas en el proceso de investigación llevado a cabo en esta tesis. El Capítulo 4 describe la primera técnica de análisis de TCM propuesta: la metodología **maSigPro** ("microarray Significant Profile"). La técnica maSigPro usa análisis de regresión lineal para modelar la expresión génica y lleva a cabo una estrategia en dos pasos para seleccionar los genes diferencialmente expresados (d.e.g.): en el primer paso se identifican los genes de interés y en el segundo paso se detectan, gen a gen, los perfiles con evolución diferencialmente significativa en el tiempo. En el Capítulo 5 se adapta la técnica multivariante ASCA (ANOVA Simultaneous Component Analysis) a datos de TCM, obteniendo como resultado el método **ASCA-genes** que combina la exploración multivariante de datos de series temporales con un procedimiento de selección para la identificación de genes con cambios relevantes. El capítulo 6 incluye un tercer estudio en el que se combina la habilidad de ASCA para detectar las señales de expresión génica, teniendo en cuenta el comportamiento coordinado de los genes, con la habilidad de maSigPro para modelar los datos de series temporales e identificar los d.e.g. Los resultados muestran que, especialmente cuando hay alto nivel de ruido estructural en los datos, el uso de **ASCA** como una estrategia de **preprocesamiento** de datos mejora los resultados de maSigPro. También se muestra que la estrategia de filtrado de datos desarrollada puede ser usada con otros métodos para análisis de TCM mejorando las tasas de falsos negativos y positivos. Estas técnicas, al igual que muchas otras de análisis de datos de TCM, ofrecen como resultados listas de genes diferencialmente expresados. Sin embargo, en el estudio de la expresión génica, se considera un resultado mucho más útil e interpretable el indicar la regulación génica como función celular o de procesos biológicos. Normalmente, esta traducción se lleva a cabo a partir de la lista de genes diferencialmente expresados (d.e.g.). Esto implica en muchos casos limitaciones en el poder de detección debido a la necesidad de un nivel arbitrario de d.e.g. sin tener en cuenta la coordinación entre funciones biológicas. El último capítulo aborda el desarrollo de métodos estadísticos para una evaluación directa e integrada de las alteraciones que pueden sufrir las funciones génicas en TCM. Para este propósito, se han adaptado las técnicas maSigPro, ASCA y PCA incorporándoles información funcional obteniendo las metodologías **maSigFun**, **PCA-maSigFun** y **ASCA-functional**.

El documento acaba con el Capítulo 8 donde se incluyen las conclusiones y algunas propuestas de investigación futura.

Resum

La present tesi doctoral aborda l'anàlisi estadística de sèries simples i múltiples d'experiments de " Time Course Microarray " (TCM). El treball es centra en el desenvolupament, aplicació i avaluació de mètodes estadístics específics que consideren la problemàtica d'aquest tipus de dades, tant des del punt de vista de selecció de gens com de l'anàlisi funcional. Les tècniques desenvolupades es comparen amb altres mètodes de l'estat de l'art actual avaluant les diferents metodologies en termes d'eficiència i significat biològic dels resultats.

La tesi està estructurada en dos blocs principals. En el primer, es revisa la literatura rellevant i es resumeix en una part introductòria. Els Capítols 1 i 2 inclouen la descripció del funcionament de la tecnologia de "microarrays" així com una revisió crítica dels mètodes estadístics aplicats a aquest tipus de dades. En aquesta part es mostren els inconvenients que sorgeixen a l'aplicar mètodes generals a sèries temporals de "microarrays" i es justifica la necessitat de desenvolupar noves tècniques per a l'anàlisi de TCM. El Capítol 3 finalitza aquest primer bloc amb una revisió dels mètodes estadístics específics per a TCM. Moltes de les tècniques que es presenten en aquest capítol han estat publicades en el mateix període d'elaboració d'aquesta tesi.

En el Segon bloc, es presenten les tècniques estadístiques per a TCM desenvolupades en el procés d'investigació portat a terme en aquesta tesi. El Capítol 4 descriu la primera tècnica d'anàlisi de TCM proposta: la metodologia **maSigPro** ("microarray Significant Profile"). La tècnica maSigPro usa anàlisi de regressió lineal per a modelar l'expressió génica i porta a terme una estratègia en dos passos per a seleccionar els gens diferencialment expressats (d.e.g.): en el primer pas s'identifiquen els gens d'interès i en el segon pas es detecten, gen a gen, els perfils amb evolució diferencialment significativa en el temps. En el Capítol 5 s'adapta la tècnica multivariante ASCA (ANOVA Simultaneous Component Analysis) a dades de TCM, obtenint com resultat el mètode **ASCA-genes** que combina l'exploració multivariante de dades de sèries temporals amb un procediment de selecció per a la identificació de gens amb canvis rellevants. El capítol 6 inclou un tercer estudi en el qual es combina l'habilitat de ASCA per a detectar els senyals d'expressió génica, tenint en compte el comportament coordinat dels gens, amb l'habilitat de maSigPro per a modelar les dades de sèries temporals i identificar els d.e.g. Els resultats mostren que, especialment quan hi ha alt nivell de soroll estructural en les dades, l'ús de ASCA com una estratègia de preprocesamiento de dades millora els resultats de maSigPro. També es mostra que l'estratègia de filtrat de dades desenvolupada pot ser usada amb altres mètodes per a anàlisis de TCM millorant les taxes de falsos negatius i positius. Aquestes tècniques, igual que moltes altres d'anàlisis de dades de TCM, ofereixen com resultats llestes de gens diferencialment expressats. No obstant això, en l'estudi de l'expressió génica, es considera un resultat molt més útil i interpretable l'indicar la regulació génica com funció cel·lular o de processos biològics. Normalment, aquesta traducció es porta a terme a partir de la llista de gens diferencialment expressats (d.e.g.). Açò implica en molts casos limitacions en el poder de detecció a causa de la necessitat d'un nivell arbitrari de d.e.g. sense tenir en compte la coordinació entre funcions biològiques. L'últim capítol aborda el desenvolupament de mètodes estadístics per a una avaluació directa i integrada de les alteracions que poden patir les funcions géniques en TCM. Per a aquest propòsit, s'han adaptat les tècniques maSigPro, ASCA i PCA incorporant-los informació funcional obtenint les metodologies **maSigFun**, **PCA-maSigFun** i **ASCA-functional**.

El document acaba amb el Capítol 8 on s'inclouen les conclusions i algunes propostes d'investigació futura.

Acknowledgements

I would like to thank some people the direct or indirect help that I was received during the time I have been working on this thesis.

First, to my supervisors: Ana and Alberto for having guided me during this work in a very nice working environment. Alberto, thank you for accepting to supervise my thesis when there were not a lot of possibilities and for putting me in contact with Ana Conesa who has been the promoter of this work. Ana, during these last years you have been my reference in the work and in other aspects of the life. I am very happy for finding a teacher and a friend in the same person. I hope our relation lasts many years.

Secondly, I have to thank Age, Johan and Huub the opportunity they gave me to join them in Amsterdam University during two summers. They introduced me in the ASCA methodology that has been the origin of a great part of this thesis. Furthermore, they were very patient with my English accent.

I would also like to thank my colleagues: Mariló, Mariola, Marga and Aurora for their daily support. And also Jeroen, Maikel and Ana Levin for the hospitality they offered me in Holland.

Me gustaría también agradecer a mis padres la ayuda recibida en innumerables ocasiones, que ha hecho posible que haya conseguido muchos de mis objetivos. Sin ellos, ni siquiera habría iniciado este trabajo. A mi hermana Cari y a mi hermano Miguel por demostrar lo orgullosos que están de mí y de mi trabajo, y también a Cari por las clases de inglés. A Ana y Vicente por la incondicional ayuda que nos dan siempre que lo necesitamos. Y a Adela y Moisés por el interés que siempre han mostrado por mi trabajo.

Por ultimo, a Dani, por creer siempre en mí y haber aguantado pacientemente momentos de trabajo y agobios. Y a Daniel y Guille, por todos los momentos de alegría y ternura que día a día compartimos.

A Daniel, Guillermo y Dani

Table of Contents

Part I: Introduction

Part II: Developed methodologies

Part III: Conclusions, Annexes and References

Justification, Objectives and Contributions

DNA Microarrays or DNA chips have imposed in the last two decades as technology of choice for the high-throughput analysis of gene expression. Microarray technology involves multiple design and data mining issues that need to be addressed throughout the overall process of data analysis. Most of the questions that arise during this process have been solved with classical mathematical, statistical and visualization tools such as linear regression, analysis of variance, multivariate projection techniques, clustering methods, etc. However, specific adaptations were frequently required to particularly respond to the characteristics of this type of data, such as the high levels of noise, the large number of variables, the limited possibilities of replication, etc.

Gene expression studies involve infinity of applications of biological interest: differentially expressed gene selection between several conditions, the study of gene trends through time, gene regulatory networks, biological classification, survival analysis, etc. In this work we focus on "Time Course Microarray" (TCM) experiments which are the studies in which gene expression evolution is analysed through time for one or several experimental conditions of interest. At the time of start of this thesis very few specific methodologies were available for the analysis of this particular type of experiments and there was a strong need of dedicated methods.

••••••••

The main objectives pursued in this work are the following:

a) To study the state-of-the-art transcriptomic analysis methodologies applied to TCM.

As the focus of this thesis is time course microarray data analysis, we started our study by providing an overview on the generation process of this type of data. Moreover, as generic tools for microarray data were being applied to TCM, we reviewed some of these in detail and showed their performance on TCM using a public dataset. In particular, we focused on the following issues:

- Overview of microarray technology and bioinformatics field.

- State-of-the-art of general tools for microarray data throughout the overall process of data analysis: experimental design, normalization, visualization and inferential analysis.
- State-of-the-art of specific tools for time course microarray data.

b) To develop new statistical methods to deal with TCM focusing on short, independent and multiple series time course (MSTC)

After identifying the main limitations of the application of general tools available for microarray data to time series we have set out to develop techniques that could properly address the specific needs of this type of data. We have taken into consideration the following aspects:

- Modelling gene expression evolution through time for different experimental conditions.
- Taking into account the correlation structure of the data.
- Removing systematic noise inherent to microarray technology.
- Considering functional annotation in order to obtain biological knowledge.
- The use of combinations of multivariate techniques as PCA, linear regression and ANOVA models.
- The implementation of the developed methodologies in the statistical language and free software R to make it easily accessible to the scientific community.

c) To study the effectiveness of the developed methods by comparing their performance with other available techniques.

The developed methods have been applied to real and simulated datasets to study their performance. Furthermore our results have been compared with the results that other microarray tools offer.

●●●●●●●●

The main contributions of this thesis are outlined in the following.

This work began in July 2004 when very few specific methodologies for time course microarrays were available. Therefore, we first reviewed the existing tools for microarray data in the process of design, normalization, visualization and statistical analysis to get a general overview of the problematic. This review is summarized in Chapter 2 where classical statistical methods and some popular methodologies have been applied to a real time course dataset. The results show the limitations of such methods in MSTC.

Our efforts in providing the scientific community with new specific tools for TCM analysis resulted in the development of the **maSigPro** (microarray Significant Profile) methodology and package. This method was published in 2006 in *Bioinformatics* journal and it is explained in Chapter 4. The maSigPro methodology was implemented in the statistical language R and it is freely available from the Bioconductor contributed packages repository http://www.bioconductor.org/. It can also be run at the GEPAS (http://www.gepas.org) suite for microarray data analysis. The maSigPro method applies a linear regression model per gene to address the problem of evaluating statistically significant profile differences and selects genes for which the model is statistically significant. The analysis of the regression coefficients of the fitted models permits the identification of the conditions for which the gene shows significant profile changes. Although maSigPro has proven to be a very useful approach to TCM analysis, the methodology does not take into account the correlation structure of the data. However, biological processes do operate in a coordinated fashion and thereby gene expression obeys to mechanisms of co-regulation and co-expression. This translates in transcriptomics data matrices into the existence of correlation structures, and this property can be used to get better estimates of the parameters of the models computed by maSigPro.

Following this consideration we explored the application of multivariate techniques to the analysis of MSTC. Chapter 5 describes an adaptation of ASCA (ANOVA-SCA) to MSTC. ASCA is a versatile approach that can deal with a temporal and/or design structure of complex multivariate datasets which are increasingly abundant in genomic technologies. Based on the ASCA model we developed a new strategy for gene selection called **ASCA-genes**. This resulted in a powerful tool to understand the shared behaviours of gene expression studied under different conditions, to identify genes that follow the discovered patterns and to avoid noisy components that pollute the data. This work was published in 2007 in *Bioinformatics* journal.

Next, we considered the integration of maSigPro and ASCA, using **ASCA** as a **filter** (Chapter 6) to benefit from the best of both technologies. The exploitation of the correlation structure of the data provided by ASCA helps to identify most significant trends by filtering out the noise in the signal and the signal in the noise, while the regression model fit present in maSigPro allows for the identification of genes with statistically significant different behaviours. In practise, we use first ASCA as a data pre-processing technique and then apply maSigPro to the filtered data matrix. We have applied this new approach in several simulation studies with different levels of random and structural noise. The results show that, especially when high structural noise is present in the data, the strategy greatly improves maSigPro results. We have also checked that this data filtering method can also be applied to other recent methods developed for the analysis of TCM.

All these techniques, as other microarray data analysis methods, generate lists of differentially expressed genes. In all cases, statistical analysis has focussed in the modelling of gene expression patterns and in the identification of differentially expressed genes. This orientation, though valid and useful, solves only one (frequently the first) requirement in the interpretation of gene expression changes. In most cases, the analysis proceeds with the identification of cellular process and functions which are represented by the gene selection, i.e. genes are identified for their functional role (functional annotation) and the question is then which functional alterations can be derived from the gene changes. In Chapter 7 we have set out to develop data analysis methods that consider biological knowledge when analysing TCM data. For this we have adapted maSigPro, ASCA and PCA to integrate functional annotation data resulting in the novel methodologies **maSigFun**, **PCA-maSigFun** and **ASCA-functional**.

Finally, a general conclusion of the thesis has been included in Chapter 8 where a summary of the most noticeable results obtained throughout this work are shown.

Journal papers

Conesa A., Nueda, M.J., Ferrer, A. and Talón, M. (2006) maSigPro: a Method to Identify Significantly Differential Expression Profiles in Time-Course Microarray Experiments. Bioinformatics, 22 (9), 1096-1102.

Nueda, M.J.; Conesa, A.; Westerhuis, J.A.; Hoefsloot, H.C.J.; Smilde, A.K.; Talón, M. and Ferrer, A. (2007) Discovering gene expression patterns in Time Course Microarray Experiments by ANOVA-SCA. Bioinformatics, 23 (14), 1792-1800.

Nueda, M.J.; Sebastián, P.; Tarazona, S.; García-García, F.; Dopazo, J.; Ferrer, A. and Conesa, A. (2009) Functional Assessment of Time Course Microarray data. BMC Bioinformatics. In Press.

Nueda, M.J.; Westerhuis, J.A.; Hoefsloot, H.C.J.; Smilde, A.K.; Ferrer, A. and Conesa, A. (2009) Using ASCA as data pre-processing technique in TCM. In preparation.

Nueda, M.J.; Ferrer, A. and and Conesa, A. (2009) Critical review of TCM methods. In preparation.

Nueda, M.J.; Ferrer, A. and and Conesa, A. (2009) Tools for microarray data. In preparation.

Conference presentations and posters

V Jornadas Nacionales de Bioinformática. Barcelona, December, 2004. Conesa, A.; Nueda, M.J.; Ferrer, A. and Talón, M. "maSigPro: a Method to Identify Significantly Differential Expression Profiles in Time-Course Microarray Experiments".

X Conferencia española de biometría. Oviedo, Spain. May, 2005. Conesa, A.; Nueda, M.J.; Ferrer, A. and Talón, M. "Analysing Profile Differences in Time-Course Microarray Experiments".

7th Spanish Symposium on Bioinformatics and Computacional Biology, Zaragoza, November, 2006. Nueda, M.J.; Conesa, A.; Westerhuis, J.A.; Hoefsloot, H.C.J.;

Smilde, A.K. and Ferrer, A. "Combining ANOVA and Principal Components to analyse Time-Course Microarray data".

7th Spanish Symposium on Bioinformatics and Computacional Biology, Zaragoza, November, 2006. Nueda, M.J.; Conesa, A.; Westerhuis, J.A.; Hoefsloot, H.C.J.; Smilde, A.K. and Ferrer, A. "Model-based multivariate methods as pre-processing in time series microarray data analysis".

XI Conferencia española y 1er encuentro iberoamericano de biometría. Salamanca, Spain. June, 2007. Nueda, M.J.; Conesa, A.; Westerhuis, J.A.; Hoefsloot, H.C.J.; Smilde, A.K. and Ferrer, A. "ASCA-genes: Analysis of Time-Course Microarray Data by ASCA modelling".

XXX Congreso Nacional de Estadística e Investigación Operativa. Valladolid, September, 2007. Nueda, M.J.; Conesa, A.; Westerhuis, J.A.; Hoefsloot, H.C.J.; Smilde, A.K. and Ferrer, A. "Combining maSigPro and ASCA procedures to enhance analysis in Time-Course Microarray experiments".

VIII Jornadas de Bioinformática. Valencia, February, 2008. Nueda, M.J.; Conesa, A.; Dopazo. J. and Ferrer, A. "maSigFun: finding functional changes in time course microarray".

Developed software

Conesa, A. and Nueda, M.J. (2005). **maSigPro**: MicroArray SIGnificant PROfiles. R Software in Bioconductor contributed packages. Also available at http://www.ua.es/personal/mj.nueda and http://bioinfo.cipf.es/aconesa.

Conesa, A. and Nueda, M.J. (2007). **ASCA-genes**. R Software free available at http://www.ua.es/personal/mj.nueda and also at http://bioinfo.cipf.es/aconesa.

Conesa, A. and Nueda, M.J. (2008). **Functional Series**. R Software free available at http://www.ua.es/personal/mj.nueda and also available at http://bioinfo.cipf.es/downloads.

Abbreviations

ANOVA: Analysis of Variance

ASCA: ANOVA-SCA

BIC: Bayesian Inference Criteria

BNs: Bayesian Networks

BP: Biological Processes

CART: Classification and Regression Tree

CC: Cellular Locations

cDNA: complementary DeoxyriboNucleic Acid

CGH: Comparative Genome Hybridization

DBNs: Dynamic Bayesian Networks

d.e.g.: differentially expressed genes

DNA: DeoxyriboNucleic Acid

EBI: European Bioinformatics Institute

EM: Expectation-Maximization

FDR: False Discovery Rate

FN: False Negatives

FP: False Positives

FWER: Family-Wise Error Rate

GEO: Gene Expression Omnibus

GO: Gene Ontology

GSA: Gene Set Analysis

KEGG: Kyoto Encyclopedia of Genes and Genomes

LIMMA: Linear Models for Microarray Data

LSD: Least Significance Difference

MAANOVA: MicroArray ANOVA

maSigPro: microarray Significant Profiles

MF: Molecular Functions

MM: Mismatch

mRNA: messenger Ribonucleic acid

MSTC: Multiple Series Time-course

NCBI: National Center for Biotechnology Information

ODP: Optimal Discovery Procedure

PAM: Prediction Analysis of Microarrays

PBNs: Probabilistic Boolean Networks

PCA: Principal Component Analysis

PLS: Partial Least Squares

PM: Perfect Match

RNA: Ribonucleic acid

RT-PCR: Real Time-Polimerase Chain Reation

SAGE: Serial Analysis of Gene Expression

SAM: Significance Analysis of Miroarrays

SCA: Simultaneous Component Analysis

SNPS: Single Nucleotide Polimorfisms

SOM: Self Organizing Maps

SOTA: Self Organizing Tree Algorithm

SPE: Squared Prediction Error

TCM: Time-Course Microarray

UPGMA: Unweighed Pair-Group Method using Arithmetic averages

Chapter 1

Introduction

1.1 The Omics Era and Bioinformatics

In the last two decades life sciences research has undergone a revolution due to the development of the omics technologies. These new technologies allow the study of all biomolecules of one organism simultaneously. The term omics refers to the comprehensive analysis of biological systems. A variety of omics disciplines have begun to emerge (Figure 1.1). **Genomics** deals with the systematic use of genome information and it includes investigations about the structure and function of the genes. **Transcriptomics** examines the expression level of mRNAs of the genes in a given cell population. **Proteomics** addresses the large-scale study of proteins, particularly their structures and functions. Similarly, **Metabolomics** studies the metabolites, which are chemical substances that cellular processes produce or synthesize. The integration of different information from these fields of study to understand and model biological processes is named **Systems Biology**.

All these technologies have generated large quantity of information and required the development of a new discipline to store and analyse the data in a way different to that employed in traditional genetic studies where only some biomolecules are analysed. As a response **Bioinformatics** and **Computational Biology** emerge as the use of techniques including applied mathematics, statistics, computer science, chemistry and biochemistry to address biological questions, usually at the molecular level. The terms bioinformatics and computational biology are often used interchangeably. However bioinformatics more properly refers to the creation and advancement of algorithms, computational and statistical techniques, and theory to solve formal and practical problems posed by or inspired from the management and analysis of biological data. Computational biology, on the other hand, refers to hypothesis-driven investigation of a specific biological problem using computers, carried out with experimental and simulated data, with the primary goal of discovery and the advancement of biological knowledge.

Initially, the term bioinformatics was used to denote specific tasks related to the storage of biological data and sequence alignment. As the field evolved, the term was also employed to include all the algorithms and techniques developed to interpret and understand all biological data produced and stored by omics technologies. Moreover, databases were created to standardize, collect and integrate molecular data, biological knowledge and experiments, and tools were developed to access and interrogate this information (some popular examples are Ensembl http://www.ebi.ac.uk/ensembl/ and GenBank http://www.ncbi.nlm.nih.gov/Genbank/, for genome and sequence data, the Gene Ontology (GO) http://www.geneontology.org/, for gene product annotation, or ArrayExpress http://www.ebi.ac.uk/microarray-as/ae/ and GEO, Gene Expression Omnibus http://www.ncbi.nlm.nih.gov/geo/, for transcriptomics data). Currently, understanding the biological phenomena at the genome level implies the use of multiple data intensive resources and algorithm development tends to orientate towards data integration and systems biology.

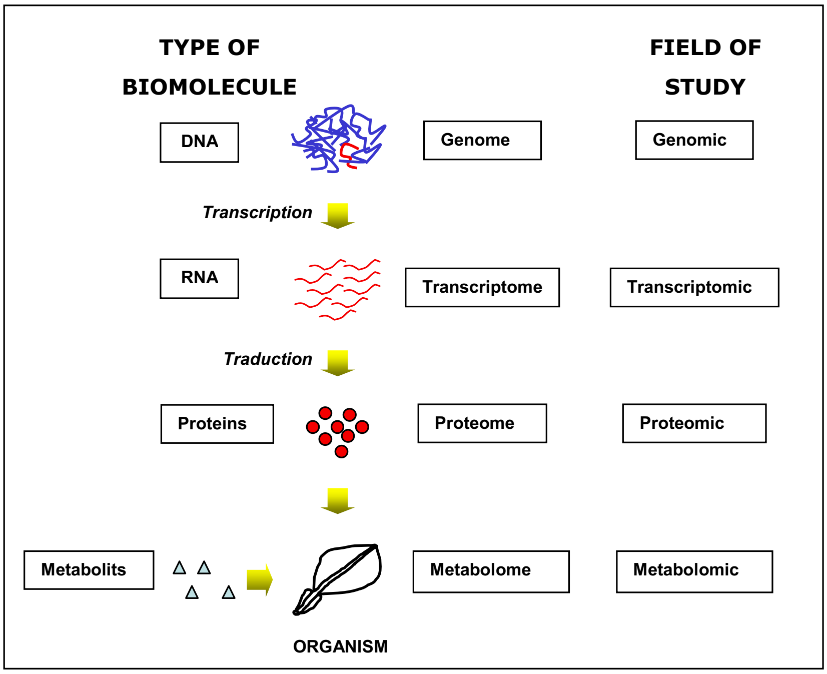


Figure 1.1: Omics technologies. These methodologies deal with all the biomolecules of a specific organism.

This work focuses on the analysis of transcriptomics data that has a time component. Throughout this thesis we present and apply several statistical tools to deal with gene expression data, although most techniques described here can be applied to data from any other omics technology. The transcriptomics technology is briefly described in the next section. This technological advance permits the simultaneous measuring of the gene expression levels of a large proportion of the genes on a genome, thereby allowing study the gene interactions and function gene regulation on the large scale. The use of microarrays and their biological applications was recently reviewed by Gresham *et al*., 2008.

1.2 Gene expression microarray data

The classical paradigm in genetics establishes that genes express or copy themselves to transcript RNA, and this RNA is translated into proteins which are the ultimate molecules that control and establish the cellular biochemical status. It is well known that gene expression is not constant, but variable with time and across tissues, and these changes control the cellular physiology and, ultimately, the phenotype.

There are several techniques available for measuring gene expression: serial analysis of gene expression (SAGE), cDNA library sequencing, differential display, cDNA substraction, multiplex quantitative RT-PCR (Real Time-Polimerase Chain Reation), and gene expression microarrays (Ahmed, 2002). Microarrays quantify gene expression by measuring the hybridization of DNA immobilized on a small glass matrix to mRNA representation from the sample under study. The advantage of the microarray technology is that it can measure, with only one experiment or array, the expression of all of the genes of an organism. At present, there are two main microarray transcriptomics technologies: cDNA arrays and oligonucleotide arrays which can be used in combination with one or two dye labelling strategies. Although traditionally cDNA arrays were coupled to the two colour arrays, the use of commercial oligonucleotide arrays of either one or two colour has imposed in the last years. The most popular oligonucleotide chip is commercialized by the company Affymetrix with the name GeneChip, which use probe sets. Other platforms are from Agilent, Codelink and Nimblegene that use short oligonucleotides, i.e., DNA sequences of up 80 nucleotides which are normally synthesised in vitro and placed onto the array. In the following we briefly comment the elaboration process of the two traditional microarray platforms: the two colour (cDNA) chips and Affymetrix chips.

cDNA microrrays consist of glass slides where a collection of DNA fragments, normally cDNA libraries, are spotted at defined positions, each cDNA fragment or probe ideally representing one gene of the genome. These chips are interrogated with two biological samples, habitually the sample of interest and a control or reference sample. mRNA from these samples is isolated, and reverse-transcribed with incorporation of a fluorescent label or dye. Two dyes (Cy3, green, and Cy5, red) are used to mark differentially each sample. After labelling, samples are mixed and hybridized onto the microarray, previous denaturalization of both array probes and sample double-stranded cDNAs. After hybridization, slides are scanned at two wavelengths to identify fluorescent signals corresponding to each dye on every probe position or spot. Spot signal intensity is quantified as the mean pixel level at red and green channels. The relative gene expression of the sample related to the control will be the ratio between these intensities. High fluorescence indicates large amounts of hybridized cDNA of the respective sample and it is related to the level of gene expression (Figure 1.2). A ratio greater than one indicates that the gene is more expressed in the sample than the control (i.e. is over-expressed), while for a ratio smaller than one follows that this gene is less expressed in the sample than in the control (i.e. is repressed). If the expression is similar in both cases the ratio will be close to 1 and the gene is not differentially expressed. cDNA arrays and oligonucleotides of Agilent, Codelink and Nimblegene use this strategy. More details of this technology can be found in DeRisi *et al.* (1997).

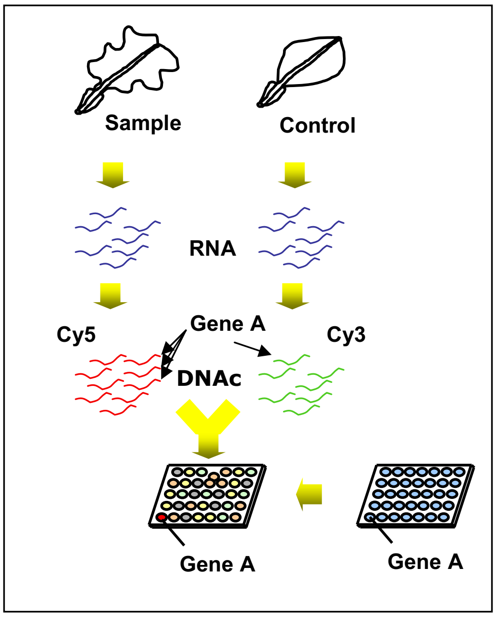
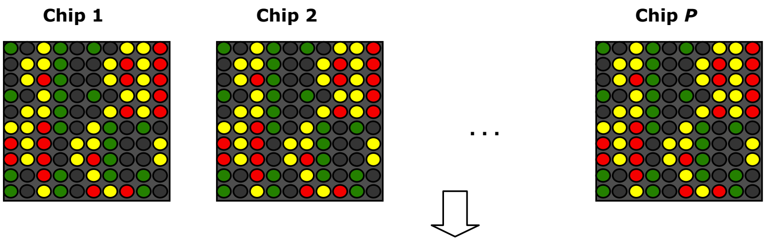


Figure 1.2: DNA chip preparation process.

The Affymetrix chip uses probe sets along with one colour technology. In these arrays, expression of each gene is measured by comparing hybridization of the sample mRNA to a set of probes, composed of 11-20 pairs of oligonucleotides, each of length 25 base pairs. The first type of probe in each pair is known as perfect match (PM) and it is taken from the gene sequence. The second type is known as mismatch (MM) and it is created by changing the middles base of the first sequence to reduce the rate of specific binding of mRNA for that gene. An RNA sample is taken, labelled with a fluorescent dye and hybridized onto the array. When scanning the arrays, two vectors of intensity readings, one for PMs and one for MMs are obtained for each gene. The expression level of the gene is the average difference between PM and MM. This technology is described in Affymetrix (1999).

Apart from DNA chips to measure gene expression, there exist other types of DNA chips with other purposes. For instance, Exons chips are useful to study alternative splicing, CGH (Comparative Genome Hybridization) to the detection of large genome alterations, and Illumina chips to detect SNPS (Single Nucleotide Polimorfisms) variations across individuals.



$$\boldsymbol{X} = \begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1P} \\ r_{21} & r_{22} & \cdots & \\ \vdots & & \cdots & \\ r_{N1} & r_{N2} & \cdots & r_{NP} \end{pmatrix}$$

Figure 1.3: Gene expression profiles for *N* genes in *P* microarrays. Each cell is measured with a gene expression ratio in a scale from green (ratio<1) to red colours (ratio>1) going through yellow colour (ratio=1). Gene expression values are represented in matrix ***X***, where rows are the genes and columns the chips.

Normally, in a transcriptomic experiment more than one sample is analysed. Each sample will be associated with an experimental condition. The information obtained from the microarrays is organised in a data matrix ***X*** where the rows represent the genes and the columns the different situations or arrays (Figure 1.3). We name **gene expression profile** at the different expression values that one gene has for the different conditions, treatments or tissues (rows of matrix ***X***). If we consider the columns of this matrix ***X***, we are looking to the transcriptional status or **fingerprint** of a given sample. Before applying data analysis to find responses to our biological questions, addressed by the microarray study, preliminary processes have to be made to the data to extract the signal of interest. These preliminary processes depend on the employed technology but, in general, the normalization and removal of the noise are critical steps required before proceeding with the statistical analysis of transcriptome differences. These steps are discussed in next chapter.

1.3 Time Course Microarray Data

This thesis focuses on the analysis of a specific type of transcriptomics experiments, where gene expression is measured over time and under one or more biological conditions. Biological questions addressed by these experiments could be divided into three major types. One type of experiments aims to the understanding of gene-expression basis of physiological phenomena or developmental processes, for instance the study of the cell cycle. Another kind of experiments tries to determine the gene expression response to stimuli or treatments. Finally, time course experiments are also designed to study gene regulatory networks or interactions between genes.

From the experimental design point of view, time series are classified based on different criteria: the number of time points, the number of biological conditions and the independency between each individual time point. When the number of time points ranges between 3-6 we talk of short time series while more than 6 time points are considered as long series. The second categorization divides time course data into single or multiples series data, if one experimental group or more are evaluated. Finally, time course experiments can be classified into longitudinal and independent, also named cross-sectional data. In longitudinal series individuals are sampled at different time-points, whereas in independent experiments samples at different times are from different and independent individuals. Normally, in these experiments several replicate measurements are available and the evolution of the averages of each time point across time will be analysed taking into account their independence, as they belong to different individuals.

Experiments where the purpose is the study of biological processes are normally related to long, longitudinal and single time series. In these cases gene expression of natural biological processes are studied, as the case of the study of cell cycles and circadian rhythms where periodic expression patterns are expected. In contrast, studies addressed to investigate the responses to stimuli use frequently short, independent and multiple time series. The goal of multiple time course experiments is to analyse the differences in gene expression between the various experimental groups, i.e. different treatments or tissues. These responses to stimuli are usually expected in a predefined period of time included in the design therefore these series are also short. Finally, gene network data usually are associated with single series of time-course data. Most recent research in this area integrates multiple datasets to derive co-expression modules defined through a large variety of biological conditions.

Chapter 2

Tools for Microarray Data

2.1. Introduction

Microarray technology generates gene expression data in unprecedented amounts. Specific algorithms and computational tools are required to cope with the analysis of these huge data volumes. In this chapter we provide an overview of the main steps in microarray data analysis, illustrating their current use with experimental data and discussing the achievements and limitations. In particular we address aspects such as experimental design, data pre-processing, descriptive and inferential analysis.

Due to the relatively high cost of microarrays and the normally limited amount of biological material available, a very important aspect is to determine the proper experimental design and to establish the number of replicates necessary to obtain a given statistical/discovery power. Furthermore, microarray data must be pre-treated and normalized to remove technical noise that disturbs the biological signal and to set all arrays in the same experiment to the same base-line. Moreover, in two colour technology, the effect of the labelling dyes and the fact that the data consists of ratio values between the two dye intensities are relevant aspects to consider during data pre-processing. Finally, the researcher must choose the adequate data analysis technique that correctly addresses the biological question under investigation. A first step in the analysis of transcriptomics is normally the visualization of the data globally. Different clustering algorithms are used to this end and in some cases this is the only analysis that is performed, which could be sufficient when few arrays and few conditions are studied. Clustering techniques help to group genes with similar behaviour and the visualization of the profiles of these groups can reveal meaningful biological patterns. However, these techniques are only descriptive and the conclusions can not be evaluated in terms of statistical significance.

One of the main goals in microarray studies, which is also within the scope of this thesis, is the selection of genes with different expression between two or more conditions. Classical inferential techniques and also microarray-specific algorithms have been applied to this end. These aspects will be treated in detail in the following sections. Other uses of transcriptomics such as the classification of biological samples or gene network analysis are not in the scope of this thesis and we will only mention them briefly.

2.2 Experimental design

As previously stated, one of the main goals in transcriptomics is to identify genes with different expression under the experimental conditions of study or, in other words, to analyse if the different experimental conditions have an influence in gene expression. However, transcriptomics signals are not noiseless and the proper identification of the existing sources of variation is an important aspect in a successful analysis of the data. Factors that contribute to the measured intensity values are the array, the gene, the treatment and, in two colour chips, also the labelling dye. Array effect refers to non-biological sources of variation collected in a single hybridization experiment, such as the differences in slide manufacturing, variations in the biochemical reactions, scanner intensity, etc. Gene effect relates to the different levels of expression between genes but also variations in the quantity of DNA in the array spots. Treatment effects measures the variation in gene expression through the different experimental conditions such as different tissues, treatments doses, time-points or a combination of these. Finally, in two colour arrays, a dye effect means that the two dyes (green Cy3 and red Cy5) can undergo differential incorporation efficiencies into the molecules of their respective cDNAs. Other sources of variation exist that are more difficult to estimate as the instability of the RNA molecule, the synthesis reaction of cDNA from the RNA, the initial quantity of RNA in the samples, etc. These sources will contribute to increase the residual variation.

Although the cost of microarrays has reduced significantly in recent years, a multiple treatment microarray experiment is still quite costly. For this reason the election of an adequate design is required to optimize power in statistical analysis with the minimum number of replicates. In two colour arrays, this implies making a decision on how to label and distribute samples through hybridizations, which is referred as "array design". Due to the high influence of the dye in gene expression, several specific designs for two colour microarray experiments have been developed. The "Reference" design is the most used in practise although there are other alternatives such as the "Loop" design and the "dye-swap" design (Figure 2.1).

The "Reference" design. In this design the samples associated with the treatments are labelled with the same dye and are hybridized against a common control that is labelled with the other dye. This common sample can be a truly experimental control, but more frequently it is a pool of all the samples included in the experiment, which is then denoted as "reference". This design is very extensive due to the fact that it only needs *v* arrays to the study of *v* treatments, it allows for adding *a posteriori* treatments and the comparison between the different treatments is simple as all of them are referenced to the same control. Furthermore, in this design the treatment effects are completely confounded with the dye effects and the reference is the sample from which more information is being obtained and it is possibly the sample with the least interest for the study, which has caused some criticisms towards this approach.

The "Loop" design. Kerr and Churchill (2001) proposed this design with the intention of extracting more information from the samples of interest with the same number of arrays as the "reference" design. From each treatment two samples are labelled with red and green dyes and samples are hybridized following a loop structure (see Figure 2.1). In this way, the dye effects are not confounded with the treatment effects but a drawback appears as the number of labelling reactions is doubled (2*v* instead of *v*+1 in the reference design) which can be considered as an extra effort. On the other hand, this design complicates the comparison between non-adjacent treatments and suffers from great instability of missing values.

Sometimes it is useful to introduce a reference as another variety in the loop design ("loop with reference", Draghici, S., 2003), for example if the interest is to study the evolution of a drug through time the control could be a sample before the treatment.

The "Dye-Swap" design. This experimental design provides two measurements for each sample through labelling with each dye. It uses two arrays to compare two samples. In the first array the control is labelled with one of the dyes (i.e. the green dye) and the treatment is labelled with the other (i.e. the red dye) while in the second array the dyes are exchanged. This design allows estimating dye effects, avoiding and removing this from the treatment effect.

These are the three basic designs but combinations of them are also widely used such as, for example, a "reference" design with "dye-swap". Some of these combinations are shown in Yang and Speed (2002) where designs for factorial and time course experiments are compared.

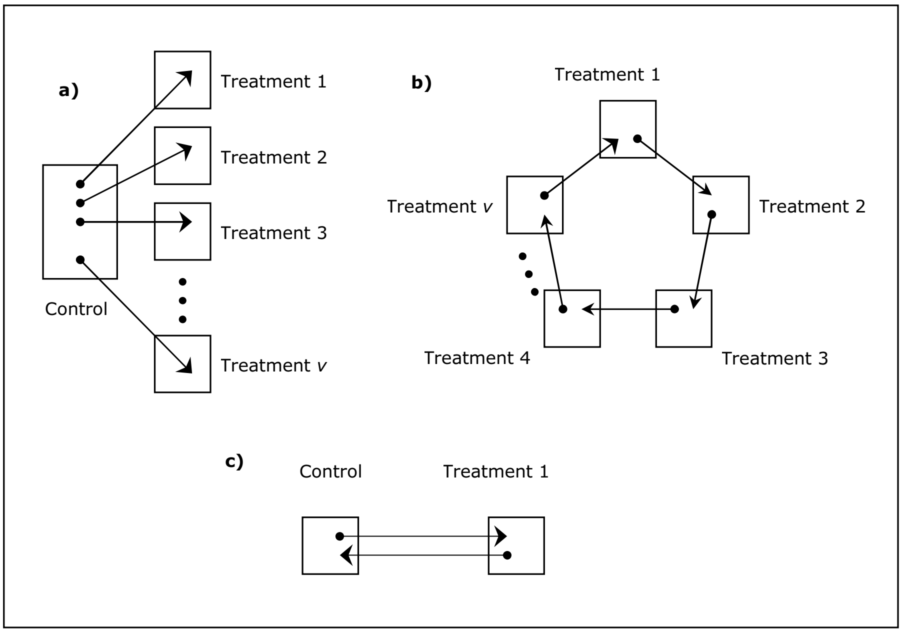


Figure 2.1 The most common designs. a) Reference design, b) Loop design and c) Dye-swap design. Nodes represent the control samples and arrows the treatment with which the control is hybridised in the same array. See text for descriptions.

2.3 Data pre-processing

Data pre-processing are the transformations and manipulations needed to prepare the data for posterior statistical analysis. Although there is no standard protocol, the usual steps are logarithm transformation, treatment of missing values and outliers, replicate handling and normalization. The large dimensions of gene expression datasets sometimes hamper the performance of these computations due to the amount of memory required and the fact that effective tools for other types of data can fail in transcriptomics analysis. Visual inspection to estimate the amount of missing data or to look for outliers, for example, is not enough. For this reason specific programs and web services have been developed to carry out the pre-processing microarray data (Tárraga *et al*., 2008 and Kapushesky *et al*., 2004 to cite some popular web sites). In the following sections we discuss the logarithm transformation and normalization as being the most relevant and specific steps in two colour microarray data pre-treatment process.

2.3.1 The logarithm transformation

When gene expression is computed as the ratio between two hybridization signals, the range of over-expressed genes, that is greater than 1, is not the same as the range of repressed genes, that is]0,1[. In this case the logarithm transformation is applied to microarray ratios to generate data symmetrical distributions and provide more interpretable comparisons between genes. The most usual logarithm base is 2. In this manner, for a gene whose expression in the treatment is the double of the control, the ratio=2 and *log2(ratio)*=1*.* Conversely, a gene with half of the expression in the treatment will have a *ratio*=0.5 and $log_2(ratio)$=−1. With this transformation the values now reflect that the two genes change in different directions by the same magnitude (Quackenbush, 2001). For one colour data, log transformations are also normally taken to reduce the scale of the data. In some cases, logarithms are taken for the ratio of each sample to the control array, to make expression data look similar to the two colour situation.

2.3.2 Normalization

As previously mentioned, technical aspects inherent to microarray technology introduce sources of variation in the data that alter the identification of the true gene expression signals. A data normalization step needs to be applied to remove this noise and calibrate all observations. Array, dye and background variations are the most usual sources of noise which are taken care of during normalization. However, not every statistical analysis equally needs the normalization steps, for instance ANOVA can isolate this variation in the model and only focuses on the variation of interest.

The use of fluorescent dyes, inherent in 2 colour microarray technology, introduces variation and error sources. The different labelling efficiency between the dyes means that the red and green signals are not exactly equivalent. Moreover, the auto-fluorescence of spotted DNA on the red spectrum, adds a positive signal value to red-labelled samples which results in a bias of the logarithm of the ratio (computed as *log2(*Red/Green)) towards positive values at low intensity ranges. Moreover, the noisy nature of data at low intensity values are amplified when ratios are used. Working with ratios implies that the magnitude and variation of the data is greater in low intensities than in high intensities. For example, a gene with an absolute difference of 2 at low intensities: Red=4, Green=2 corresponds to a log-ratio=1, meanwhile the same difference at high levels is an absolute difference, Red=257, Green=255 corresponds to a log-ratio=0.01. This effect is clearly shown in a self-self experiment where two samples of the same RNA are labelled with different dyes (Dudoit *et al.*, 2002b). If the dye did not have an influence on gene expression the log_2R vs. log_2G plot would be on the diagonal. However we can observe in Figure 2.2a) a deviation to high values at low intensities. This effect, known as the name of "Banana shape", is more noticeable in the MA-plot (Figure 2.2b). This plot represents $M=log_2R/G$ *vs.* $A=log_2\sqrt{R\times G}$ ($A=\frac{1}{2}(\log_2 R+\log_2 V)$, mean of the intensities logarithm) and it is a 45º counterclockwise rotation of the log_2R vs. log_2G plot along with a scale change.

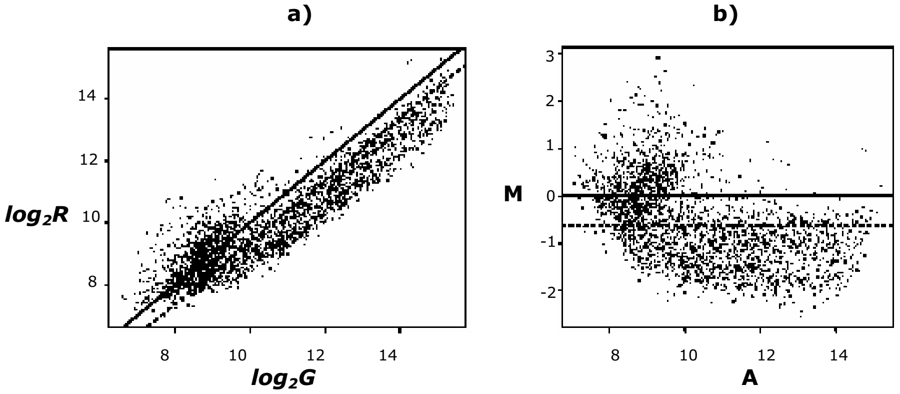


Figure 2.2: Hybridization of two samples of the same RNA. **a)** log_2R vs. log_2G *R: red and G: green intensities*. **b)** MA-plot. $M=log_2R/G$ $A=log_2\sqrt{R\times G}$. M=0 is the solid line and the median the dashed line (Dudoit *et al.,* 2002b).

Normalization can be carried out using genes for which a constitutive and invariant expression level is assumed (i.e. house-keeping genes), or using spiked controls, that are synthetic or unrelated DNA sequences, included in the array design and mixed at known concentrations with the labelled samples. However, these strategies are not always available, because the first ones require study and definition of the house keeping genes –which is very controversial– and the second require additional purchases. Therefore, the most usual normalization techniques are based on several properties of global gene expression. The most used assumption is the invariability: thousands of genes are analysed in a chip but the condition under study only affects a small percentage of them. Therefore we can assume without introducing big errors that gene expression of the majority of the genes does not change or that the number of over-expressed and repressed genes is nearly equivalent. This hypothesis implies that the average of the ratios should be close to 1. Under this assumption several normalization methods have been proposed, from simple standardization to non-linear regression techniques.

Yang *et al*. (2001) describe normalization methods that account for the intensity and spatial dependence for cDNA microarray experiments. We distinguish two types of normalization: within-slide normalization and between-slide normalization. Figure 2.3 shows a graphical example with gene expression representation of a slide through the different transformations until normalized data is obtained.

1) Within-slide normalization: the normalization is carried out taking into account the available information separately for each array.

- Global normalization: To normalize the different arrays, often global normalization is the simplest methodology: a *c* constant adjustment to achieve the distribution of the log-ratios intensity has a median or average of zero for each slide:

  $$\log_2(R/G) \rightarrow \log_2(R/G) - c\,. \tag{2.1}$$

  However, such global normalization approaches are inadequate in situations where systematic noise depends on overall intensity or spatial location within the array.

- Intensity dependent normalization (smoother lowess): To account for the intensity and spatial dependence normalization methods based on robust local regression have been proposed. The most used normalization method is "Lowess" (locally weighted regression technique) that adjusts the ratios through locally linear fits that depend on the intensity: *f(A)*, representing A the intensity as $A = log_2\sqrt{R \times G}$:

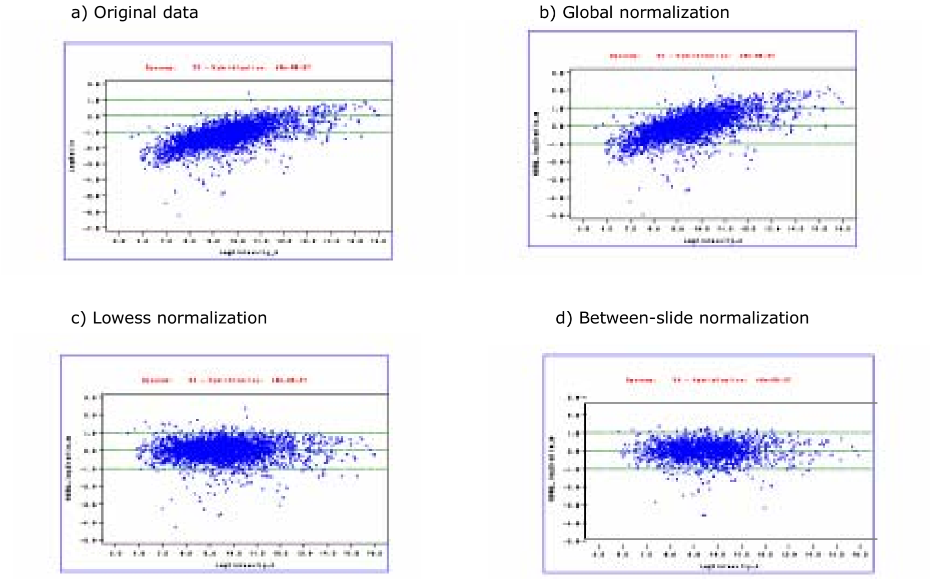
  $$\log_2(R/G) \rightarrow \log_2(R/G) - c(A)\,. \tag{2.2}$$

  The estimation of the function *f(A)* depends on the choice of the bandwidth, that establishes the area where the function is estimated, and the parametric function to fit. These criteria are related because changes in the bandwidth can determine that another type of function was more adequate (Cleveland and Loader, 1996).

- Within-print-tip group normalization: Apart from the intensity it is possible that the print-tip or subdivision of the chip affects the gene expression measurements. In spotted arrays there are several grids that are printed with the same print-tip or position of the arrayer, and systematic differences may exist between the print-tips. In these cases it is recommendable to apply different lowess fits for each grid, *i:*

$$\log_2(R/G)_i \to \log_2(R/G)_i - c_i(A), \quad i = 1,\ldots,I. \tag{2.3}$$

The problem here is that the assumption of invariability must hold for each print-tip group, which is not always possible.



a) Original data

b) Global normalization

c) Lowess normalization

d) Between-slide normalization

Figure 2.3: MA-plots with the change of the data distribution through the different transformations until normalized data is obtained. a) Original data. We can see the dependency between gene expression and intensity. b) With global normalization the points are shifted up. c) Lowess normalization eliminates the intensity dependency. d) Finally the between-slide normalization homogenizes the variation between the different slides.

2) Between-slide normalization. Previous normalization methods are effective in obtaining centred data but it is still possible that the log-ratios have different variances across slides (see Figure 2.4). In such cases between-slide or scale normalization could be applied.

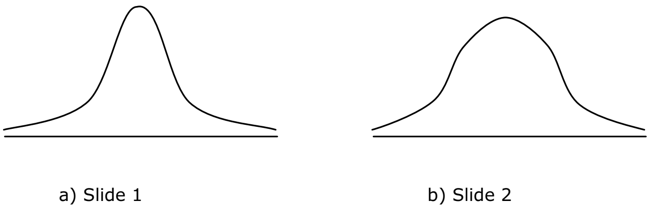


Figure 2.4: An example where gene expression variation is different in two different microarrays. After scale normalization the array variance will be the same and therefore the ratios comparable.

One useful approach is to assume that the log-ratios of the *i*-th array are normally distributed with mean zero and variance $a_i^2\sigma^2$, where σ^2 is the variance of the log-ratios and a_i^2 the scale factor for the *i-th* array. These factors can be estimated using the maximum likelihood estimators that are the sample variance of each array:

$$s_i^2 = \frac{\sum_{j=1}^{N_i}\left(M_{ij} - \bar{M}_i\right)^2}{N_i}, \tag{2.4}$$

where M_{ij} denotes the *j-th* log-ratio in the *i-th* array group, *j*=1,…,N_i and $\bar{M}_i$ is the ratio average of the *i-th* array. However it is more usual to use the statistic:

$$\hat{a}_i = \frac{MAD_i}{\sqrt[I]{\prod_{i=1}^{I} MAD_i}} \text{ being } MAD_i = median_i\left\{\left|M_{ij} - median(M_{ij})\right|\right\}, \tag{2.5}$$

that is an estimator for which numerous empirical works have shown that remove the influence of outliers. After estimating these parameters data will be rescaled by the factor $1/\hat{a}_i$ to achieve similar scales.

Huber *et al*. (2002) proposed another method to stabilize the variance. They considered that the variance of the data is related to the mean by

$s_x^2 = (c_1\bar{x} + c_2)^2 + c_3$, being $c_3 > 0$, a relation that must be checked in each experiment. To get independence between the mean and variance, they suggested the following transformation $h(x) = \gamma ar\sinh(a + bx)$, with $\gamma = c_1^{-1}$, $a = c_2 / \sqrt{c_3}$ and $b = c_1 / \sqrt{c_3}$, that coincides with the logarithmic transformation for large intensities.

Durbin and Rocke (2004) introduced a transformation within the generalized-log family which stabilizes the variance of the difference of transformed observations. They also introduced transformations from the started-log and log-linear-hybrid families which provide good approximate variance stabilization of differences. More recently, Papana and Ishwaran (2006) used Classification and Regression Tree (CART) to cluster genes by the different variances. This classification allows for the improval of the estimation of the population variance to better stabilize the variance of the data.

Quantile normalization is another important normalization technique (Bolstad *et al.* 2003). This method is the standard normalization procedure used in one-colour chips as Affymetrix but it can also be applied to two-colour technology. It is based on the idea that the distribution of two data vectors is the same if the quantile-quantile associated plot is a straight diagonal line that we can represent by the unit vector $\left(1/\sqrt{2}, 1/\sqrt{2}\right)$. Extending this to *n* data vectors the quantiles plot will be the line given by the *n*-dimensional unit vector $\boldsymbol{d} = \left(1/\sqrt{n}, \cdots, 1/\sqrt{n}\right)$. Therefore projecting the quantiles of *n* arrays onto the diagonal makes the distributions for each one of them the same. Due to the fact that the projection of each quantile is the mean quantile vector, by taking $\boldsymbol{q}_k = (q_{k1}, \ldots, q_{kn})$ for $k = 1, \ldots, p$ to be the vector of the k^{th} quantile the projections on $\boldsymbol{d}$ are $proy_{\boldsymbol{d}}\boldsymbol{q}_k = \left\{\frac{1}{n}\sum_{j=1}^{n} q_{kj}, \ldots, \frac{1}{n}\sum_{j=1}^{n} q_{kj}\right\}$, the method consists in replacing the original values for the corresponding mean quantile. We can see that the method forces the values of quantiles to be equal losing original information from the data.

2.4 Exploratory analysis

Graphical representations of transcriptomics data are useful mechanisms to obtain an overall understanding of the variation patterns contained in large datasets and to find associations between components. When exploring the data, two aspects can be considered, the gene fingerprint for each experimental condition (column-wise analysis) and the trend or profile of each gene in the different experimental conditions studied (row-wise analysis).

Initial transciptomics studies considered the comparison gene expression profiles, rows of matrix ***X***, for grouping genes in homogeneous classes and available clustering algorithms were used to this end (Spellman *et al*., 1998 and Cho *et al.,* 1998). Clustering analysis helps exploration of gene expression profiles as we can jointly visualize genes with similar profiles. Different experimental conditions can be also clustered by comparing columns of matrix ***X*** and in this way the relationships can be derived between clustered conditions and genes.

All clustering techniques are based on comparing distances between the elements to group: rows or columns of matrix ***X***. In data analysis the most frequently used distance measures are Euclidean, Manhattan, correlation and Mahalanobis distances. Given two vectors $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ of dimension q (p in row-wise analyses and N in column-wise analysis) these metrics are defined as:

1. Euclidean distance: $d(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sqrt{\sum_{k=1}^{q} \left(x_{ik} - x_{jk}\right)^2}$.

2. Manhattan distance: $d(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sum_{k=1}^{q} \left|x_{ik} - x_{ik}\right|$.

3. Correlation distance: $d(\boldsymbol{x}_i, \boldsymbol{x}_j) = 1 - r$, being r the Pearson correlation coefficient:

$$r = \frac{\sum_{k=1}^{q} (x_{ik} - \bar{x}_i)(x_{jk} - \bar{x}_j)}{\sqrt{\sum_{k=1}^{q} (x_{ik} - \bar{x}_i)^2 \sum_{k=1}^{q} (x_{jk} - \bar{x}_j)^2}}$$

4. Mahalanobis distance: $D^2 = (\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{V}^{-1} (\mathbf{x}_i - \mathbf{x}_j)$, being ***V*** variance-covariance matrix.

As we can observe the Euclidean and Manhattan distances are based on the differences from each pair of points, therefore they measure the geometric closeness of the two vectors. Correlation distance measures the co-variation of the two vectors and Mahalanobis distance takes into account the variance of the vectors and also their covariance. As most of the times there is an interest in understanding co-expression patterns, the most used distance in transcriptomics analysis is the correlation distance.

Traditional clustering techniques have been applied to microarray data and many new methodologies have also been developed or adapted from classical methods. In general clustering algorithms can be divided into two major types:

- **Hierarchical cluster**, where groups are joined in a hierarchical structure according to the distance between them. The most frequently used is UPGMA (Unweighed pair-group method using arithmetic averages).

- **Non hierarchical cluster or partitioning methods**, which seek to obtain an optimal division of the data being the number of clusters prefixed. Typical representative algorithms are k-means and Self Organizing Maps (SOM).

Another useful approach for microarray data exploration is Principal Components Analysis (PCA). Although it is not specifically designed for clustering purposes, PCA can help to visualize groups of genes with similar behaviour.

2.4.1 Hierarchical cluster

Hierarchical cluster analysis can be agglomerative or divisive. The agglomerative ones begin with all the elements as groups and join them until one unique group is formed. The divisive ones begin by considering the total group as unique and work by splitting it sequentially. Hierarchical procedures are usually performed visually in a tree diagram named dendogram, where the joints or divisions are represented.

As the agglomerative methods are the most widely used, to simplify the exposition, we only develop this procedure that has the following steps:

1. Initially the N elements to group are considered as clusters. We consider matrix $\boldsymbol{D}$ that collects the distances between the N objects to group.

2. An inter-cluster distance measure is chosen, represented by $\delta(C_i, C_j)$, being

C_i and C_j two different clusters. Distances between all pairs of clusters are computed.

3. The two nearest clusters are joined as one.

4. The distances between clusters are computed again considering the change done in step 3. Then, step 3 must be repeated till obtaining only one cluster.

Given the clusters C_i, C_j, the most used inter-cluster distances are:

1. Single linkage (closest neighbour): $\min\{d(\boldsymbol{x}_i, \boldsymbol{x}_j),\ \ \boldsymbol{x}_i \in C_i,\ \ \boldsymbol{x}_j \in C_j\}$.

2. Complete linkage (furthest neighbour): $\max\{d(\boldsymbol{x}_i, \boldsymbol{x}_j),\ \ \boldsymbol{x}_i \in C_i,\ \ \boldsymbol{x}_j \in C_j\}$.

3. Centroid linkage: $d(\bar{\boldsymbol{x}}_i, \bar{\boldsymbol{x}}_j)$, the distance between the centres of the clusters, being $\bar{\boldsymbol{x}}_i = \frac{1}{n_i}\sum_{\boldsymbol{x}_i \in C_i} \boldsymbol{x}_i$ and $\bar{\boldsymbol{x}}_j = \frac{1}{n_j}\sum_{\boldsymbol{x}_j \in C_j} \boldsymbol{x}_j$.

4. Median linkage: $d(\boldsymbol{Me}_i, \boldsymbol{Me}_j)$, the distance between the median of the clusters.

5. Average distance: $\frac{1}{n_i n_j}\sum_{\substack{\boldsymbol{x}_i \in Ci \\ \boldsymbol{x}_j \in Cj}} d(\boldsymbol{x}_i, \boldsymbol{x}_j)$, average distance of all elements in each cluster.

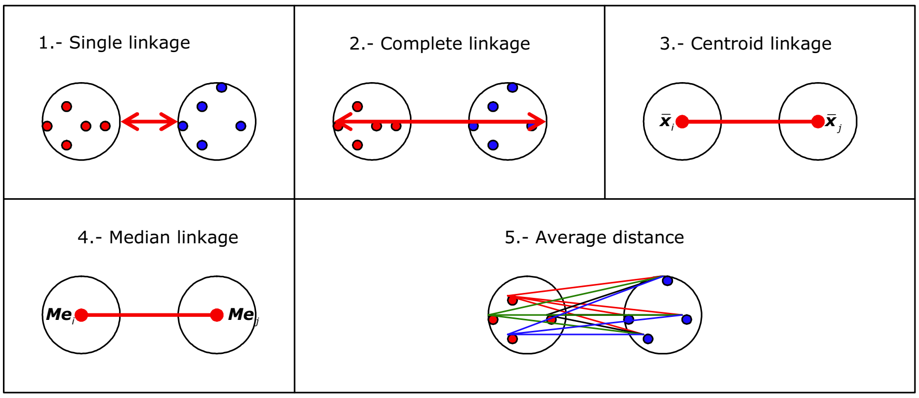


Figure 2.5: Inter-clusters distances graphical examples.

As we can see in Figure 2.5, single and complete linkage depend on the extreme data, therefore the existence of outliers can have a high influence on the results. The median linkage is the inter-cluster distance that must be used in these cases. However, centroid linkage and average distance are the most used inter-cluster distances, which are also influenced for outliers but with less intensity.

2.4.2 Non hierarchical cluster

Non hierarchical cluster or partitioning procedures are divisive and their goal is to make a unique partition of *K* groups, this number being a priori fixed. To determine *K* the researcher must be guided by their own experience or by the previous hierarchical cluster results. In such procedures the elements are assigned to the groups by optimizing a pre-selected criterion.

The most well known approach is the **k-means** algorithm. This procedure first chooses the centres of the clusters randomly or by using any information available. Secondly the elements are assigned to the closer cluster using an adequate distance measure and after each assignment the centroids are recomputed. Finally, when all the elements are assigned, distances between all the elements and centres are computed to evaluate possible changes of cluster. The process stops when there is no reassigning to do.

2.4.3 Self Organizing Maps (SOM)

Kohonen (1997) proposed another non hierarchical clustering algorithm named **SOM**. This procedure was designed to perform a grid in which similar cluster patterns are plotted next to each other, in other words, a plot where the neighbourhood of each cluster is similar to the other. The number of clusters and the desired geometry of nodes (for example, 6 clusters in a 3×2 grid) must be pre-specified.

Initially, the representation of the nodes in the *P*-dimensional space (if we are clustering genes) is random and they go changing sequentially to the optimum solution (see Figure 2.6 where P=2). The subsequent position of the nodes is represented by $f_t(k)$, $t=1,\ldots,T$, f_0 being the initial position and T the total number of iterations to do. To group N genes in K groups, firstly a gene is selected randomly (G) and the nearest node is computed (k_G). Then all the nodes are modified by applying:

$$f_{t+1}(k) = f_t(k) + \tau(d(k,k_G),t)(G - f_t(k)), \tag{2.6}$$

being $\tau(x,t) = \begin{cases} \frac{0,02T}{T+100t}, & x \le \rho(t) \\ 0, & \text{otherwise} \end{cases}$ and $\rho(t)$ the threshold to decide if a node is changed or not (Tamayo *et al.*, 1999). In this way only the closest nodes to k_G are changed and the rate $\tau(d(k,k_G),t)$ becomes smaller in each iteration, the changes to make also being smaller. This adjustment tries to approximate the nodes to the centroids of the existing clusters. The process is repeated with the rest of the genes until a cycle is complete. Several cycles can be developed.

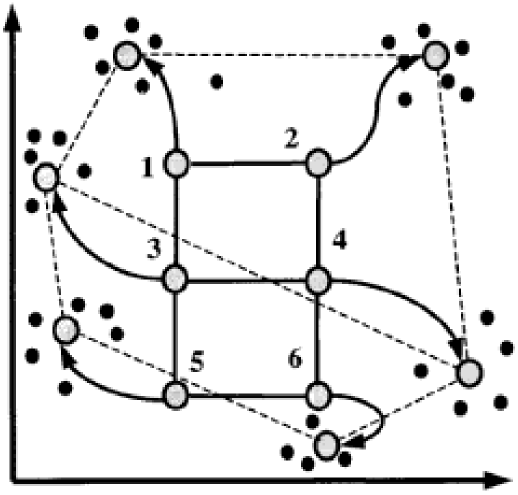


Figure 2.6: SOM example. Initially there are 6 nodes in a 3×2 rectangular grid. Black points are the data and grey circles the centroids. Arrows indicate changes in the position of the nodes until the final configuration is obtained.

SOM is a very effective algorithm to exploratory data analysis and it is capable of exposing the fundamental patterns of a great amount of elements. However, it does not perform a hierarchical structure to show the relationships between clusters. Furthermore, it has been detected that when a very common type of profile exists and with very little variation, as in flat profile cases, the majority of the clusters reflect this situation and the changing genes, normally a minority, are represented in a few clusters losing the important details in their representation.

2.4.4 Self Organizing Tree Algorithm (SOTA)

SOTA is another divisive method that combines the data management capability of SOM and the hierarchical cluster method advantages (Herrero *et al.*, 2001 and Herrero and Dopazo, 2002). SOTA begins doing a SOM where the number of clusters to develop is two. Then, the cluster with the most variation is chosen to have SOM applied to it to divide this cluster in two new groups and so on. Each time a cluster is divided the clusters are analysed to decide if the process stops or continues. The decision is made depending on two criteria: a pre-established level of variation in all the clusters and a prefixed number of clusters to develop.

By comparing the time consumed by hierarchical cluster, SOM and SOTA, we can see that this time for SOM and SOTA is proportional to N (genes to study) whereas in hierarchical cluster it is proportional to N^3.

2.4.5 Principal Components Analysis (PCA)

Projection techniques such as PCA can help to represent a summary of the available information in a single plot. PCA is a multivariate technique that reduces the dimension of a set of objects measured in a P-dimensional basis, taking advantage of the relationship between the variables. The method consists of projecting the individuals on a subspace of dimension $Q<P$ extracting the major information. The solution of this problem is the subspace defined by the Q eigenvectors associated with the Q higher eigenvalues of the variance-covariance matrix of the data. The representation of the objects (genes) and the variables (experimental conditions), in the new dimension allows for graphically visualizing the relationships between them.

$\boldsymbol{X}_{NxP}$ being the data matrix with N genes and P conditions, the problem can be formulated as minimizing the global deformation of the original cluster of dots when it is projected on a vectorial subspace W. This global deformation is called inertia over W and it is defined as $I_w = \sum_{i=1}^{N} p_i d^2(\mathbf{x}_i, \hat{\mathbf{x}}_i)$, being p_i the weight or importance of the object i, $\hat{\mathbf{x}}_i = \boldsymbol{proy}_{\mathbf{w}}(\mathbf{x}_i)$ is the projection vector of $\boldsymbol{x}_i$ on W and

$d^2(\mathbf{x}_i, \hat{\mathbf{x}}_i) = \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|^2$ the distance between the original representation of the object and its projection in the new space. The problem is translated to $Min \quad I_{\mathbf{w}}$.

Taking $W = span \ \{\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_Q\}$, with $\{\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_Q\}$ orthonormal system of vectors, I_W can be expressed as $I_W = \sum_{i=1}^{N} p_i \|\mathbf{x}_i\|^2 - \sum_{k=1}^{Q} \mathbf{w}_k^T \mathbf{V} \mathbf{w}_k$, where ***V*** is variance-covariance matrix of the variables $x_1, x_2, \ldots x_P$. As the first addend is constant, the problem is translated to maximize the second addend: $Max \quad \sum_{k=1}^{Q} \mathbf{w}_k^T \mathbf{V} \mathbf{w}_k$. The solution to this problem is $\{\mathbf{u}_1, \ldots, \mathbf{u}_Q\}$, an orthonormal system of eigenvectors associated with the Q highest eigenvalues of matrix ***V***.

The k-th principal component, $\mathbf{z}^k$, is the vector of $\mathbb{R}^N$ whose components are the projections of the original data on the new space: $z_i^k = \boldsymbol{x}_i^T \boldsymbol{u}_k, \ i = 1, \ldots, N$.

2.5 Inferential analysis

Visualization tools described in the previous section are very useful to find genes with similar profiles and to identify samples with similar gene expression fingerprints. However, these methods are only descriptive and they do not provide enough information to derive a value of significance, i.e. to indicate generalization power. As the main purpose of microarray analysis is the identification of differentially expressed genes, hypothesis testing methods are usually applied. In transcriptomics data, this implies as many statistical tests as variables and the appearance of a multiple testing scenario that demands adjustments of the single test *p*-values. Classical statistical methods such as the *t*-Student test, ANOVA *F*-test or mixed models have been applied. However, to address the specific problems of microarray data, new methodologies or variations of classical statistical procedures were developed. In the following we mention some of the most noteworthy aspects especially related to microarray data and the methodologies that address them. Firstly, we introduce classical statistical tests to show the simplest way to handle this analysis and to show the test basis from which the majority of the tools for microarrays have started. Secondly, we mention the main aspects inherent to microarray data. Finally, we describe some specific tools developed to deal with microarray experiments.

2.5.1 Classical statistical tests

Two conditions

The simplest and most frequent microarray experiment involves only two conditions, for instance experiments where cancer is compared with healthy tissue. Two conditions can be directly compared in the same array for each replicate using two-colour cDNA microarrays technology and the hypothesis to test is if the ratio is 1 (or if the log-ratio is 0). An arbitrary level is normally used as cut-off value to declare a gene as differentially expressed. However, this test, named "fold" change is not a statistical test. When replicated data is available, the *t*-test statistic associated with this experiment for each gene is based on:

$$T = \frac{\bar{x}}{s/\sqrt{m}} \sim t_{m-1} \qquad (2.7)$$

being:

$\bar{x}$: Sample average log-ratio.
m: Sample size or number of replicates.
s^2: Sample variance.

This sample distribution assumes normally distributed data or the size of the samples is big enough to apply the central limit theorem (m≥30).

Two conditions can also be compared indirectly by hybridizing the samples with a common reference and testing to see if there are differences in the ratios, that is the same test to use for single-colour technology. As the most usual experiments compare samples indirectly, the developed methods and also the methods described in this chapter can be applied to both technologies. The *t*-test statistic associated with this experiment for each gene is based on:

$$T = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{\frac{(m_1-1)s_1^2 + (m_2-1)s_2^2}{m_1+m_2-2}\left(\frac{1}{m_1}+\frac{1}{m_2}\right)}} \sim t_{m_1+m_2-2} \qquad (2.8)$$

being:

$\overline{x}_i$: Sample average log-ratio of condition i, $i = 1,2$.
m_i: Sample size of condition i, $i = 1,2$.
s_i^2: Sample variance of condition i, $i = 1,2$.

This test is also based on normality assumptions and furthermore equality of the variances between the samples of the two conditions.

More than two conditions

When the object of the study is to compare the effects of *K* treatments of a factor, for example, multiple drug treatments or different doses of the same treatment, ANOVA (ANalysis Of Variance) is the classical tool developed to handle this problem. To explain the differences in gene expression through the different levels of the considered factor, the following model is considered for each gene:

$$x_{ij} = \mu_j + \varepsilon_{ij} = \mu + \alpha_j + \varepsilon_{ij}, \qquad i = 1,\dots,N_j, \qquad j = 1,\dots,K. \tag{2.9}$$

x_{ij} being the *i*-th replication of gene expression at level (or treatment) *j*, μ the global average, α_j the effect of the *j-th* level of the factor, N_j the available data in level *j* and ε_{ij} the residual term.

To determine the existence of differences between the gene expression population average at the different levels, in other words to contrast the null hypothesis $\text{H}_0\colon \alpha_1 = \alpha_2 = \cdots = \alpha_K = 0$ against the alternative hypothesis $\text{H}_1\colon \exists i \neq j \,/\, \alpha_i \neq \alpha_j,\ i,j \in (1,\dots,K)$, the following decomposition of the variance is considered:

$$\begin{array}{ccccc} \sum_{j=1}^{K}\sum_{i=1}^{N_j}\left(x_{ij}-\overline{x}\right)^2 & = & \sum_{j=1}^{K} N_j\left(\overline{x}_j-\overline{x}\right)^2 & + & \sum_{j=1}^{K}\sum_{i=1}^{N_j}\left(x_{ij}-\overline{x}_j\right)^2, \\ SS_T & = & SS_B & + & SS_W, \end{array} \tag{2.10}$$

where $\overline{x}$ is the global sample average and $\overline{x}_j$ the sample average for group *j*, SS_T the total sum of the squares, SS_B the sum of the squares between groups defined for the levels of the factor and SS_W the sum of the squares within groups. From this decomposition the F statistic is formed as $F = \frac{SS_B\ /(K-1)}{SS_W\ /(N-K)}$ that under the null hypothesis and assuming normality and independence of the residuals it is distributed as the F-Snedecor with (K-1, N-K) degrees of freedom.

Multifactor experiments

When two or more factors are available the interest is not only to study the differences between the groups of a factor but the possible interaction between the different factors. Taking the most simple case: an experiment with two factors, the ANOVA model to consider for each gene is:

$$x_{ijh} = \mu + \alpha_i + \beta_j + \gamma_{ij} + \varepsilon_{ijh}, \quad i = 1,\ldots,K_1, \quad j = 1,\ldots,K_2, \quad h = 1,\ldots,N_{ij} \quad (2.11)$$

x_{ijh} being the h-th replication of gene expression at level i of factor 1 and level j of factor 2, μ the global average, α_i the effect of the *i-th* level of the first factor, β_j the effect of the *j-th* level of the second factor and γ_{ij} the interaction effect between the factors at i and j levels, respectively.

By splitting the total variance in variability due to the factors and interaction in a similar way that the decomposition (2.10) three different F-statistics can be developed to test the existence of population average gene expression differences between the groups formed by the considered factors.

2.5.2 Relevant aspects in data analysis of microarray experiments

These classical methods have some limitations when dealing with microarray data. Firstly, these techniques contrast a unique hypothesis and they control the type I error rate. Transcriptomics data imply many tested variables (genes) and the control of this rate is not enough. Furthermore, there are not always enough replications to obtain good estimations of the variance of the estimators. Finally, these techniques require the normality assumption to their success, and this can not always be assumed in microarray data. In this section we discuss these three main issues that are major aspects considered by statisticians when developing new specific techniques to improve statistical analysis.

Multiple testing

The multiple testing problem is one of the most challenging topics. In experiments where there are *n* tests to do the global significance, the so-called family-wise type I error rate, is computed by $FWER = 1-(1-\alpha)^n$. In spite of selecting a very low significance level α for each test, *FWER* increases considerably. For instance, in an experiment with 1000 genes and two conditions there are 1000 statistical tests, taking α =0,01 the $FWER = 0.9999$. In other words, the procedure will almost definitely wrongly conclude that there is at least a difference in one test (when there are no real differences).

A simple solution is the Bonferroni correction that consists in choosing a global significance level and working for each comparison at $FWER/n$ level. Unfortunately this correction is very strong for gene expression analysis due to the large number of comparisons to do. The required significance level for each contrast will be so small that the almost no statistically significant gene would be found in the results, yielding many false negatives. Other procedures exist to control the *FWER* as Sidák procedure and Holm's step-wise correction (Dudoit *et al.,* 2002a). However, the control of the *FWER* is conservative. Instead, it is more appropriate to control the false discovery rate (FDR) that is the proportion of errors among the identified differentially expressed genes.

There are several procedures to control the FDR. The first and most used one is the linear step-up procedure (Benjamini and Hochberg, 1995). This procedure firstly orders the *p*-values associated with the employed statistics for *n* null hypothesis considered: $p_{(1)} \leq \ldots \leq p_{(n)}$. And secondly, for a desired FDR level *q*, the number of null hypothesis to reject is $k = \max\left\{i / p_{(i)} \leq qi/n\right\}$, *i*=1,…,*n*. This method has the drawback that implies independency of tests (gene expression) which is not true. To circumvent this problem, resampling-based FDR controlling procedures have been developed (Reiner *et al*., 2003).

Sharing information

Some authors have developed the idea of moderation or shrinkage to analyse microarray data. Moderating or shrinking gene variances refers moving the variances towards a common value estimated by using the global information of all the genes. In experiments where a small number of replicates are available for each gene, variances can be poorly estimated and therefore the results of the classical *t* or *F* statistics can lead to increased false negatives and positives. However, it is well known that genes do not act alone, therefore there is shared information within microarray data that could be used to improve variance estimates. Borrowing information from the ensemble of genes can assist in the inference about each gene individually. Tusher *et al*. (2001) developed one of the first approaches that uses this idea. These authors proposed a modification of the *t*-statistic by adding a constant in its denominator which improves the estimation of the variance. Another way to do this is through the application of empirical Bayes methods (Efron *et al.,* 2001 and Lönnstedt and Speed, 2002). In this case the information of all the genes is used to estimate the parameters of a prior distribution that is later used for especifically evaluating each gene. Recently, Storey *et al*. (2007) proposed a new approach for performing multiple tests in high-dimensional studies by applying the optimal discovery procedure (ODP). This new approach utilizes the information of all the statistical tests available to improve the existing thresholding methods. Theoretically, the data for each specific test is evaluated by using the ODP statistic. Suppose that there are *n* significant tests or genes: $x_1, x_2,\ldots,x_n$ each one with *m* observations or arrays. Assume n_0 true null hypotheses, $i=1,\ldots,n_0$ and the alternative is true for $i=n_0+1,\ldots,n$. If the significance test *i* has null probability density function f_i and alternative density g_i, they consider the statistic:

$$S_{ODP}(x) = \frac{g_{n_0+1}(x) + \ldots + g_n(x)}{f_1(x) + \ldots + f_{n_0}(x)} \tag{2.12}$$

The null hypothesis *i* will be rejected if $S_{ODP}(x_i)$ is higher than an acceptable threshold. They provide an example in Storey (2005) showing how microarray data contains information shared across genes.

Distributional assumptions

Standard statistical methods rely on the normality hypotheses to give good results. The failure to meet this requirement is not a serious problem when large samples are available because in such cases the "law of large numbers" holds. In microarray experiments high replication is not possible –or very rare- and alternative methodologies, such as permutation or empirical bayes methods, are being applied to estimate the empirical distribution of the statistics of interest or, most typical, the distribution of the FDR (Efron *et al.*, 2001 and Tusher *et al.*, 2001).

2.5.3. Specific statistical methods for microarray data analysis

Significance analysis of microarrays (SAM)

SAM (Tusher *et al.*, 2001) is one of the most popular versions of the *t*-test and it is the first method that applies the idea of moderation introduced in section 2.5.2. SAM uses an empirical distribution by applying permutation techniques and controls the false discovery rate (FDR) to take into account the multiple testing problem. The method can be applied to compare two or more conditions.

For the most simple analysis scenario (two conditions) the SAM approach computes a modified *t*-statistic (see equation 2.13) for each gene adding in the denominator a suitable constant (s_0) for all the genes which avoids those small estimated variances which give false positives for having high values of the *t*-statistic. This constant is chosen to obtain a constant coefficient of variation for the *t*-statistic that is not possible when sample standard deviations change between genes.

$$d(i) = \frac{\overline{x}_1(i) - \overline{x}_2(i)}{s(i) + s_0},\ i = 1,\dots,N \qquad (2.13)$$

SAM orders the computed values of *d(i):* $d_{(1)} \le d_{(2)} \le \dots \le d_{(N)}$ that are the observed values, and compares them with the empirical distribution or expected ones. In each permutation of the data the statistic is again computed and ordered $d^{*r}_{(1)} \le d^{*r}_{(2)} \le \dots \le d^{*r}_{(N)}$. Developing *R* permutations *r*=1,…*R* we can obtain the averages of these values $\overline{d}_{(i)} = \frac{1}{R}\sum_{r=1}^{R} d^{*r}_{(i)},\ \ i = 1,\dots,N$ that are used as the expected values under the null hypothesis of no differences. A common rejection region is defined for all the genes fixing a threshold: $d_{(i)} - \overline{d}_{(i)} \ge \Delta$. The delta value is chosen by computing the FDR for several deltas until one is obtained where $\text{FDR}(\Delta) \le \gamma$, which is prefixed. SAM was very much used in the early 2000´s. Today, it is still a popular method but some stability and granularity problems have been reported. In Zhang (2007) the drawbacks of SAM are analysed by pointing out that the main problem is the poor estimation of FDR and modifications to improve its control are being proposed.

Linear Models for Microarray Data: LIMMA

Lönnstedt and Speed (2002) applied the idea of moderation to develop the B-statistic to compare two conditions using a Bayesian mixture model. The B-statistic is the logarithm of a ratio of probabilities: the probability of being differentially expressed and the probability of invariability. They smoothed gene-specific residual sample variances towards a common value. Smyth (2004) developed the hierarchical model of Lönnstedt and Speed (2002) into a practical approach for general microarray experiments. Smyth proposes linear models to consider the comparisons of interest for each gene, two or more conditions, and applies the empirical Bayes approach to do inference about the regression coefficients of the model. The implementation of this method has been carried out in the statistical language R within the LIMMA package by adding general tools for data exploration and normalization. This package is one of the most used programs for microarray analysis.

ANOVA in transcriptomics

The analysis of variance (ANOVA) model has also been adapted to the analysis of microarray data. One of the first publications by Kerr *et al*. (2000) considered a model for non-normalized data to study the main effects with array signals: array (A), dye (D), variety (V) and gene (G) and two interactions (AG), measuring the gene specific spot bias, and (VG) which is the actual interest of the analysis. Let x_{ijkg}, denote the intensity from the *i-th* array, *j-th* dye, *k-th* variety and *g-th* gene, the considered fixed-effects ANOVA model is:

$$\log x_{ijkg} = \mu + A_i + D_j + V_k + G_g + AG_{ig} + VG_{kg} + \varepsilon_{ijkg} \qquad (2.14)$$

The fixed-effects model is applicable to many microarray experiments, however some of the factors can be considered as random samples from a population as can be the array and the spots of the array, which can be added to the model. Although the mixed model has the same structure showed above, there is a difference in the interpretation of the random effects that are considered as sources of variance. Wolfinger *et al.* (2001) proposed a two-stage approach using a mixed-model

ANOVA. These authors use a linear model in the first step to normalize the data:

$$\log x_{ijkg} = \mu + A_i + D_j + AD_{ij} + r_{ijkg} \tag{2.15}$$

and then fit a gene-specific model to the residuals of the first model:

$$r_{ijk} = G + VG_k + DG_j + AG_i + \varepsilon_{ijk} \,. \tag{2.16}$$

A similar approach in two steps but only with fixed effects was also considered by Kerr *et al*. (2000). This was implemented in a package named MAANOVA (MicroArray Analysis Of Variance), although currently this package includes random effects as well. MAANOVA package also includes normalization and visualization tools and it is implemented in both R and MATLAB programming environments. Specific variations of the classical *F* test are available to deal with microarray data (Cui and Churchill, 2003).

<u>Mixture modelling</u>

Another approach to the study of gene expression is mixture modelling. This model has been used for the study of two condition microarray experiments, (Pan *et al*., 2003), and also for clustering purposes (McLachlan *et al.,* 2002). The proposed method avoids the strong distributional assumptions of the *t*-test and regression approaches considering that the observations are from a Normal mixture distribution. In Pan (2002) a comparison of these three methods can be found which also includes SAM approach. Pan concluded that the main differences among the results of the application of these methods were produced when dealing with small samples. This is due to the fact that with large samples (>30) the standard Normal distribution can be used as the null distribution for the *t*-test. The Normal mixture model is fitted by maximum likelihood using the Expectation-Maximization (EM) algorithm (McLachlan *et al.,* 1997). The available software to apply this technology is called EMMIX and it is described in McLachlan *et al.* (1999).

2.6 Other applications of interest

Although this thesis focuses on differentially expressed gene selection, this is not the only application of microarrays. The use of transcriptomics for class prediction is a major topic in many medical fields, especially in cancer research. A recent review can be found in Boulesteix *et al*. (2008) focusing on the statistical evaluation of microarray-based prediction methods.

Basically, the problem in class prediction using microarrays has two elements. First, a method for variable selection is required, i.e., a subset of the original probes included in microarray matrices is selected. Selection is normally carried out on the basis of a per gene measure, such as by fold change, *t*-test or ANOVA. Genes are ranked by value of differential expression or significance and an arbitrary number is selected for the next step. Dimension reduction techniques, as PCA (Principal Component Analysis) or PLS (Partial Least Squares, Boulesteix, 2004), can be applied as a previous step to summarize the information and then use any classification method to the obtained components. Although this approach takes into account the correlation structure of the data, the components can be difficult to interpret. After a variable selection is obtained, the actual predictor is constructed using a machine learning approach. Larrañaga *et al.* (2006) review machine learning in bioinformatics including class prediction. Popular strategies are logistic regression, the k-nearest neighbour algorithm and support vector machines (Vapnik, 1995). There are also statistical methods based on penalization or shrinkage as the Penalized Logistic Regression (Zhu, 2004) or the Prediction Analysis of Microarrays (PAM) method based on shrunken centroids (Tibshirani *et al.*, 2002). Irrespective of the statistical method used, the methodology for developing class predictor involves the use of a part of the data, the training set to build the predictor, and another part, the test set, to evaluate its prediction power (Medina *et al*., 2007).

Another application of microarray experiments is the study of gene regulatory networks. This type of analysis evaluates the interactions between genes and looks for models to describe or predict gene expression behaviour. Describing molecular processes allows for identifying the genes involved, their relationships and their sequence of action. These models can be useful in many studies, for instance, by characterizing the gene expression mechanisms that cause certain disorders it would be possible to target those genes to block the progress of the disease.

The main approaches to study gene regulatory networks deal with time series microarray data, although there are other inputs. Bayesian networks, Boolean networks, and differential equations and other mathematical models are the most used models to this end (de Jong, 2002 and Bansal *et al*., 2007).

2.7 Method comparison applied to Time Course data

We have applied some of the descriptive and inferential methods described in the previous sections to a time course microarray experiment in order to make a comparative evaluation of the performance of the different analysis strategies and to highlight difficulties in the analysis of this type of data analysis. Hierarchical clustering, SOM, SOTA and PCA are applied to find groups of genes with similar profiles. Next, statistical inferential analysis has been applied using *t*-test and ANOVA modelling to find differentially expressed genes for some proposed questions of interest about differences between some experimental conditions.

2.7.1 Toxicogenomics experiment

In our evaluation we have used data from a toxicogenomics study where the effect of the hepatotoxicant brombenzene in rats was analysed (Heijne *et al.*, 2003). Rats were treated with three doses (low, medium and high) of bromobenzene dissolved in corn oil. Additionally, there were two groups of rats without toxic treatment: an untreated rats group and a group treated only with the drug administration vehicle, corn oil. In total there were five groups denoted by the labels: UT (untreated), CO (corn oil), LO (low dose), ME (medium dose) and HI (high dose). At different time-point measurement (6, 24 and 48 hours) one to three rats were randomly selected from each treatment group. Each individual RNA rat sample was co-hybridized against an external reference and the hybridizations were duplicated by swapping the two labelling dyes. This makes a total of 54 slides (see Table 2.1) and 2665 genes available for statistical analysis. Data pre-processing included background subtraction, calculation of log2 ratios and Lowess normalization.

This example is a Time Course experiment where there are 5 experimental groups, 3 time points and 2 or 6 replications. The original dataset consisted of 2665 genes and 54 chips associated with the conditions detailed in Table 2.1. We have also considered a subset of this data matrix denoted as “reduced data matrix” to simplify the evaluation with some of the studied techniques from which it is difficult to draw conclusions from the entire dataset. The simplification of the original data has been done by taking the averages of the replicates and removing the genes with a flat profile or genes without important changes in their profiles. To remove the flat profiles we used the tools of GEPAS "Gene Expression Pattern Analysis Suite" available at http://gepas.bioinfo.cipf.es. We considered as flat profiles genes whose expression change between -1 and 1. By doing this, the reduced data matrix consists of 487 genes and 30 conditions (5 treatment levels × 3 time-points × 2 dyes).

Slide	Treatment	Slide	Treatment	Group	Time	#Replications
1	Cy3-UT-T6	**28**	Cy5-UT-T6	1	6	2
2	Cy3-UT-T24	**29**	Cy5-UT-T24	1	24	2
3	Cy3-UT-T48	**30**	Cy5-UT-T48	1	48	2
4	Cy3-CO-T6	**31**	Cy5-CO-T6	2	6	2
5	Cy3-CO-T24	**32**	Cy5-CO-T24	2	24	2
6	Cy3-CO-T48	**33**	Cy5-CO-T48	2	48	2
7	Cy3-LO-T6	**34**	Cy5-LO-T6	3	6	2
8,9,10	Cy3-LO-T24	**35,36,37**	Cy5-LO-T24	3	24	6
11	Cy3-LO-T48	**38**	Cy5-LO-T48	3	48	2
12	Cy3-ME-T6	**39**	Cy5-ME-T6	4	6	2
13,14,15	Cy3-ME-T24	**40,41,42**	Cy5-ME-T24	4	24	6
16,17,18	Cy3-ME-T48	**43,44,45**	Cy5-ME-T48	4	48	6
19,20,21	Cy3-HI-T6	**46,47,48**	Cy5-HI-T6	5	6	6
22,23,24	Cy3-HI-T24	**49,50,51**	Cy5-HI-T24	5	24	6
25,26,27	Cy3-HI-T48	**52,53,54**	Cy5-HI-T48	5	48	6
						54

Table 2.1: Treatments assigned to each slide. The indicated dye is the assigned dye to the sample.

When we show gene profiles the order of the abscise axis is the order of the columns of the data matrix. This order coincides with the order of the slides as is described in Table 2.1 and is further indicated in Figure 2.7.

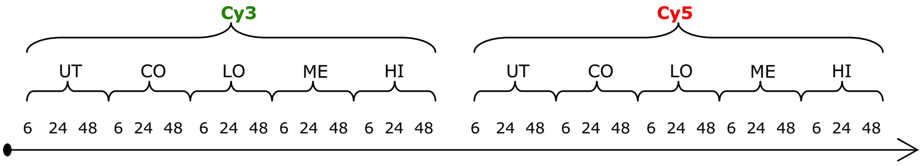


Figure 2.7: Identification of the treatment for the gene profiles representation.

2.7.2 Data visualization

A common plot used to see the overall data is a box-plot for each of the arrays. Figure 2.8 shows the 54 box-plots of our experiment. Visually, we can see that the slides are centred on 0. We can also observe that data ranges are between 2 and -2 and that there are basically equivalent distributions between different arrays, which is an indication of a proper normalization of the data.

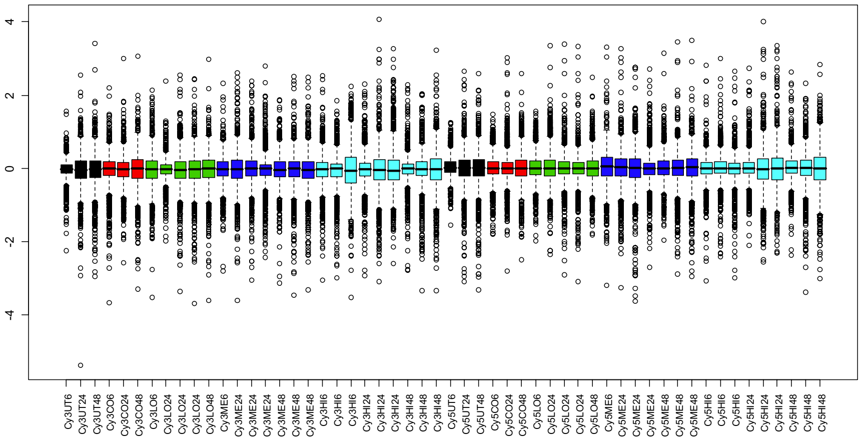
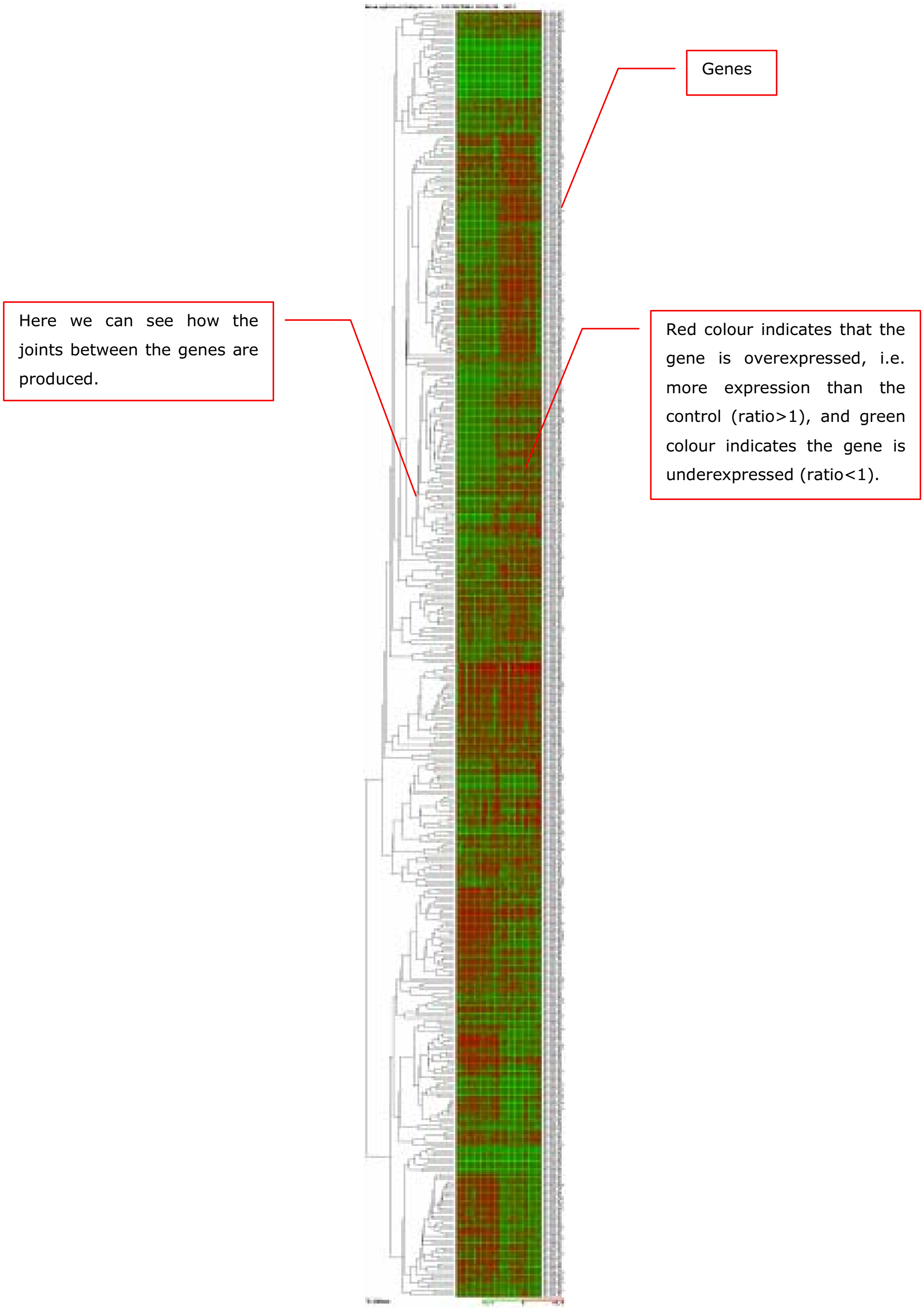


Figure 2.8: Boxplots for each one of the 54 arrays. Colours indicate the treatment group assigned to each array.

<u>Hierarchical clustering</u>

We have applied hierarchical cluster using the GEPAS suite. We chose the correlation distance to measure the relation between the genes and the centroid linkage distance to measure the inter-cluster distances. Firstly, we applied the method to the complete dataset. However, due to the huge dimension of the obtained dendogram and the difficult interpretation we prefer to show only the result with the reduced data matrix, which is shown in Figure 2.9. In this representation we can see the dendogram and a heat map together. The heat map is a graph that assigns a colour to each data, being green for negative values (repressed) and red for positive values (over-expressed). We can see that in spite of the reduction of the data, the interpretation continues to be difficult. The analysis of the dendogram suggested genes can be divided into 16 groups. This number can be taken as reference when applying other visualization techniques such as SOM, SOTA or non hierarchical clustering. The difficulty of the interpretation of this graph is not only due to the number of genes but also to the number of conditions present. If we had two conditions instead of 30, it would be simple to identify the meaning of the identified groups by looking at their colours.



Genes

Here we can see how the joints between the genes are produced.

Red colour indicates that the gene is overexpressed, i.e. more expression than the control (ratio>1), and green colour indicates the gene is underexpressed (ratio<1).

Figure 2.9: 487 genes in a dendogram obtained by applying hierarchical cluster analysis.

SOM

We applied SOM to the reduced data matrix again by choosing 16 clusters, as suggested by the previous hierarchical cluster results, with a rectangular structure (4×4). SOM gives as a result the profiles of the genes classified in each group and the average profile (black line) along the experimental conditions detailed in Figure 2.7 (see Figure 2.10). We realize that the clusters contain genes with similar patterns, for instance the first cluster is formed by repressed genes and the last one by over-expressed genes (see Figure 2.11).

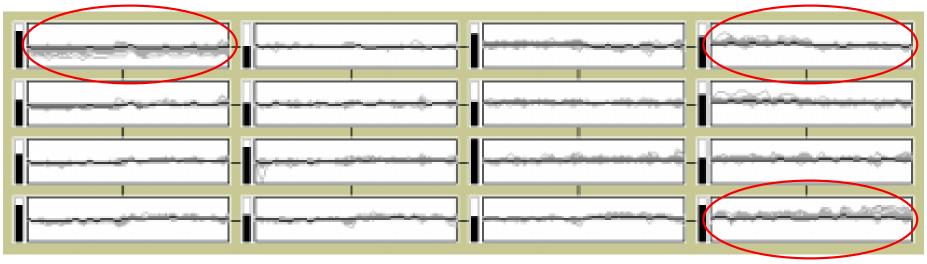


Figure 2.10: Groups obtained by applying SOM to the reduced matrix. Marked clusters show some examples of interest: repressed genes (position 1.1), over-expressed genes (position 4.4) and genes affected by the dye (position 1.4).

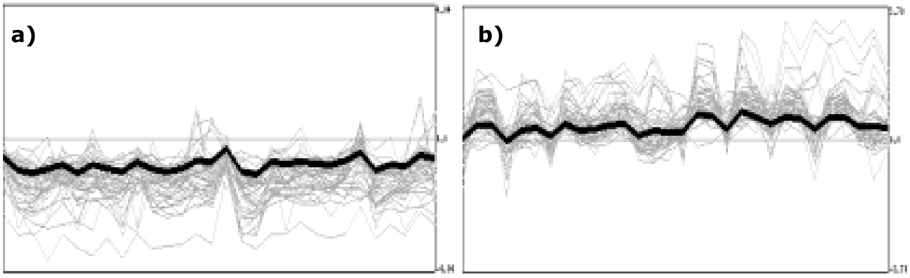


Figure 2.11: Amplified gene expression profiles of **a)** the first cluster that includes repressed genes and **b)** the last cluster that includes over-expressed genes.

We can also observe that clusters with similar profiles are adjacent. An interesting result is that the cluster in position 1.4. contains genes with different expression for the two dyes (Figure 2.12). This result leads us to think that there were genes for which the dye effect had not been removed in the normalization process and that data portioning might have been affected by this dye bias. If we are interested in studying the evolution through the treatments it would be preferable to run two SOM analyses, one with each dye to focus on the changes in trends through treatments not through the dyes (Figure 2.13).

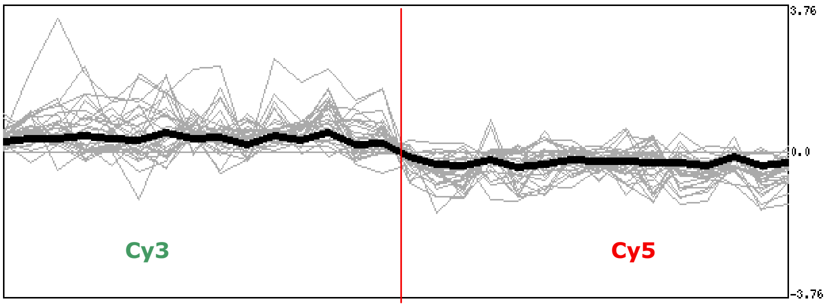


Figure 2.12: Amplified gene expression profiles of cluster 1.4. which includes genes with different expression for the two dyes.

Looking at Figure 2.13 we realize that in both results there are clusters showing changes in expression in high and medium doses of bromobenzene (e.g. third and fourth group of both results). In Figure 2.14 we can see two examples where the profiles are amplified to see the details and illustrate our previous comments.

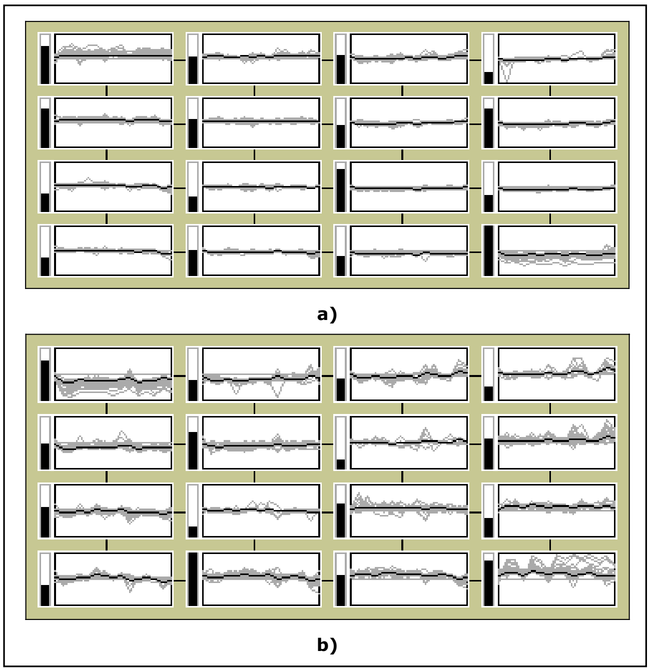


Figure 2.13: Clusters obtained by applying SOM to the samples labelled with green dye **(a)** and with red dye **(b)** of the reduced data matrix.

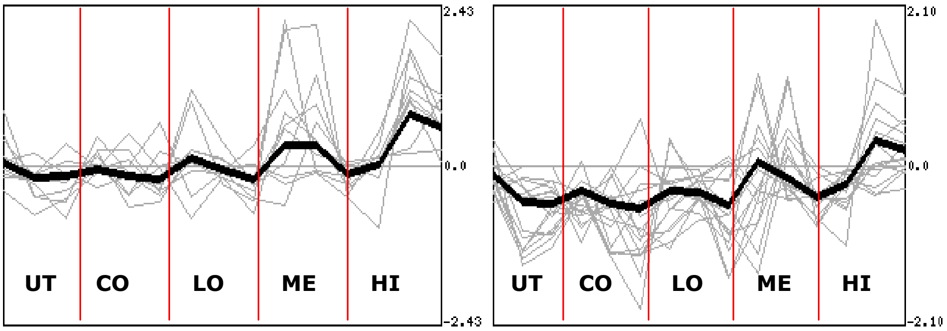


Figure 2.14: Gene expressions of the genes included in the third and fourth cluster obtained applying SOM to the data where samples are labelled with red dye. In both clusters there are changes in expression in high and medium doses of bromobenzene.

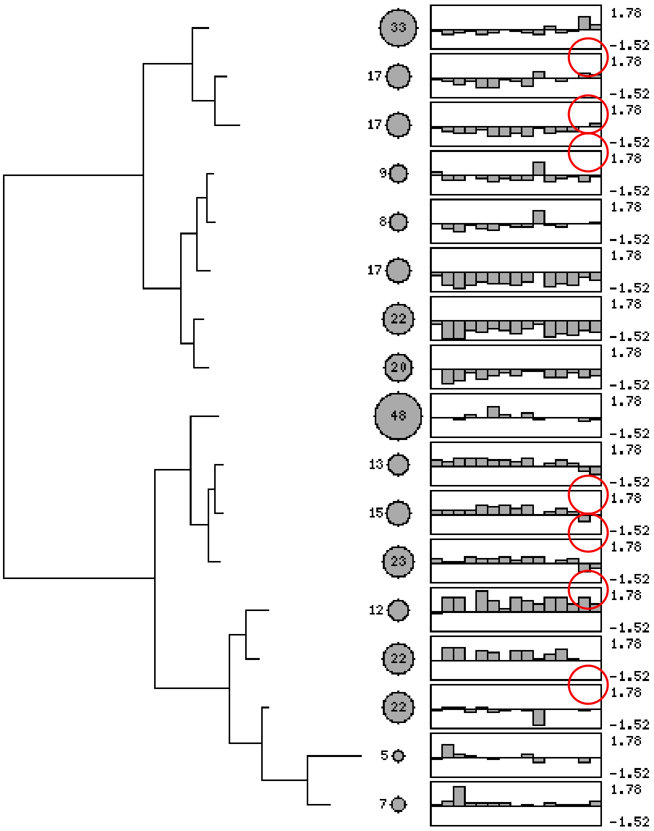


Figure 2.15: Hierarchical structure obtained by applying SOTA to the reduced data matrix choosing the data with the red dye. Graphs represent gene expression average of genes within each cluster. We can identify 7 clusters where gene expression changes for high doses. Red circles indicate the location of these changes.

SOTA

By applying SOTA to the reduced data matrix, a similar effect as in the previous case is observed, making again difficult the interpretation of the clustering results. Therefore we choose to apply the clustering method to the data split by dye orientation. For each subdata matrix it is then possible to identify clusters of genes with up-regulation and clusters of genes where expression decreases. Figure 2.15 shows the clustering results obtained by applying SOTA to the samples with red dye. The graph shows the data partioning, the number of genes in each cluster and the evolution of the profiles of each cluster with bars. Visually we can identify 7 clusters where gene expression changes for high doses. These cases have been marked with red circles: groups 1,2,3,10,11,12 and 14.

PCA

We apply PCA to the initial matrix (2665 genes and 54 slides) considering the samples (conditions) as variables and the genes as individuals. The first three components explained 60% of the total variation. By analysing these first three components we detect the overall behaviour of gene expression. Furthermore, the analysis of the residuals graph of this model (not shown) shows no large values, indicating that genes are well described. Loadings are related to the correlation between samples and principal components. By analysing the loadings of the samples we can interpret the meaning of the principal components (Figure 2.16). The first component, that explains 34% of variation, is correlated positively with all the samples, so this component is related to the basal expression level of each gene. Actually this result is not too important due to the fact that we are looking for changes in expression through the experimental conditions. We could have avoided this first component by previously centering the data gene by gene. The second component had positive correlations with the slides where the sample is labelled with red dye and negative correlations in the cases labelled with green dye. This component shows that the dye effect accounts for an important amount of data variation and identifies the genes most affected by this dye bias. Finally, the third component is negatively correlated with the slides where high doses have been applied and the time period is 24 or 48 hours. This is the effect that we have observed in previous sections. This component will be useful to identify genes related to this treatment effect.

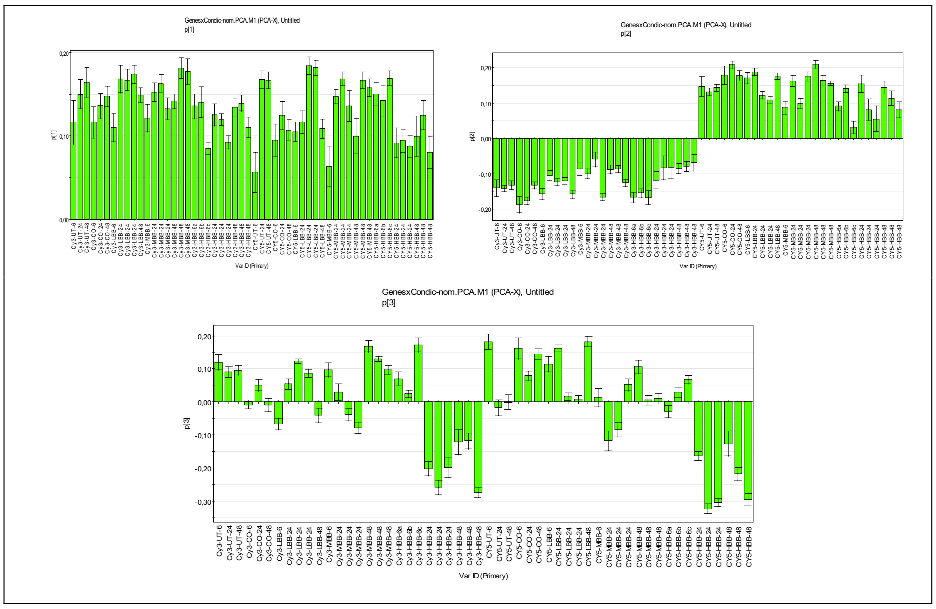


Figure 2.16: Loadings of the initial variables to the three components selected. The first component represents the signal intensity effect, the second one is related to the dye-swap and the third reveals the treatment effect.

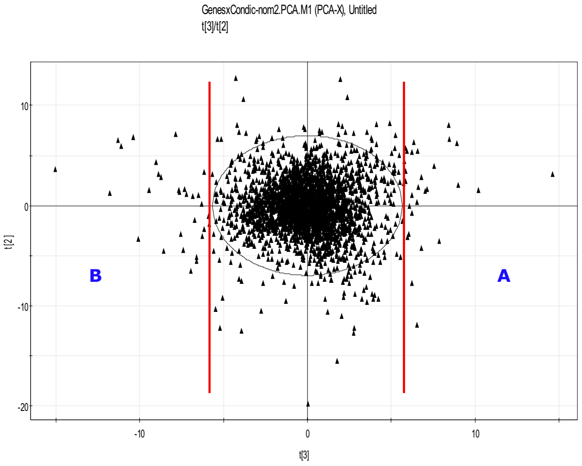


Figure 2.17: Scores values of genes in the second and third components. Genes whose expression in high doses is over-expressed are in area B and genes whose expression is repressed in high doses are in area A. Genes within the black circle are those whose expression is not affected either by high bromobenzene doses nor dye.

Figure 2.17 shows the graphic representation of the scores of the second and third components. The scores are the values of the genes in the new space. Genes with high scores for a component are genes correlated with the profile described for this component. On the other hand, genes with high negative scores are negatively correlated with the component. In our case, focusing on the third component (x-axis), we can detect two interesting groups of genes identified in areas A and B. Genes whose expression in high doses is over-expressed are in area B while genes whose expression is repressed in high doses are in area A.

Conclusions of data visualization

The application of hierarchical clustering, SOM, SOTA and PCA leads us to some interesting conclusions. Firstly, we have seen that the results obtained with hierarchical clustering are very difficult to interpret. Although this method gave us a first idea for the number of clusters to choose, we could not obtain conclusions about the characteristics of each cluster. Secondly, by applying SOM and SOTA we discovered a batch effect associated with the labelling orientation of the samples and also identified a group of genes with differential expression for medium and high doses of bromobenzene. These conclusions were obtained after a close examination of the cluster profiles. However, in the case of complex experimental designs, it is not always easy to draw conclusions from solely the visual inspection of these profiles. Thirdly, we have seen that PCA efficiently deals with the entire dataset while for the clustering methods, reduced and split matrices were needed to achieve interpretability. A PCA model with three components has revealed the data features more compactly: directly showing the signal intensity effect, the problem of the dye-swap and the effect of the treatment. Furthermore, PCA tells us that pre-processing could be improved since size effect and dye bias take more of data variability than the treatment effect that is the effect of interest.

In general, we have seen that the application of these methods is useful to discover general behaviours of the data. By selecting specific clusters or genes with high PC loadings, lists of differentially expressed genes could be generated. However, this is not an inference-based mechanism for feature selection which does not give an indication of the significance (generalization power) of the obtained gene selection. This is better achieved by applying hypothesis testing methodologies, which is discussed in the next section.

2.7.3 Inferential analysis

The experimental design of the toxicogenomic experiment has two factors of interest for each gene; time and treatment with three and five levels, respectively. Moreover, the time factor is of a continuous nature. Therefore, an appropriate analysis method for these data should be able to handle these characteristics. However, when applying the statistical techniques described in the previous sections, it becomes evident that different methods have different application scopes and that comparisons are biased for these differences. In this section, therefore, we tried to reveal these differences and indicate how they limit the analysis of a multifactorial experiment such as the one used in this example. As the dye effect discovered in the previous section could disturb the results, we eliminated this structural noise by centering each gene with its corresponding dye average.

Two conditions

A first limitation comes from approaches which are focused on the comparison of two conditions. To apply these in our cases, we need to either perform multiple pair-wise comparisons or to split data into two conditions that would provide the most informative analysis. The explorative analysis of the data carried out in the previous section suggested that gene expression changes are most pronounced at the high-dose condition. Therefore, we choose to perform a statistical comparison between this condition and the corn oil group that is one of the control groups, using the *t*-test statistics and the SAM methodology.

The application of the *t*-test implied 2665 comparisons between the average of the values with high doses (18 cases) to the average of the values with corn oil (6 cases). We computed the *p*-values associated with each *t*-test, applied multiple test correction and by choosing a FDR=0.01 we obtained 44 genes with statistically significant differences.

We also applied SAM to the same data with the same purpose using the specific application developed by Standford University through the EXCEL spreadsheet available at http://www-stat-class.stanford.edu/SAM/SAMServlet. We chose 1000 permutations and FDR=0.01 obtaining 55 genes as significant. Figure 2.18 shows one of the results that the program offers.

By comparing the *t*-test and SAM results we can observe that the number of genes is similar in both cases. Furthermore, there are 21 genes in common in both solutions.

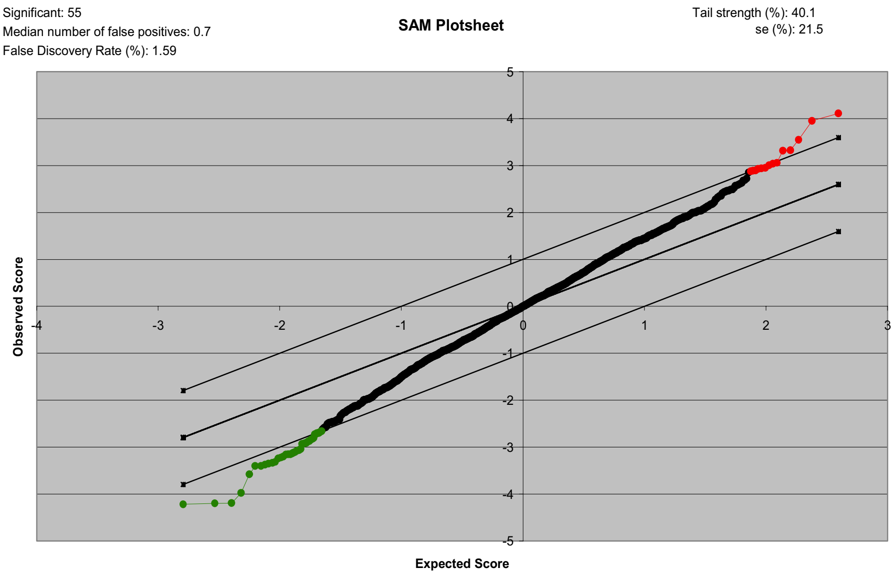


Figure 2.18: SAM plot for delta=1. Significant genes are those plotted outside the bands with red and green colours. Genes in red are over-expressed genes and genes in green are repressed.

More than two conditions

Here we applied the LIMMA R package available in the Bioconductor platform. Considering only the dose effect, first LIMMA estimates a linear model with the 5 groups involved. Then, we can consider multiple contrasts for the comparisons of our interest that are performed from the coefficients of the estimated model. E.g.: the analysis of gene expression differences between the High and the other ones: Medium, Low, CO and UT groups. This implies the analysis of four contrasts: HI.vs.ME, HI.vs.LO, HI.vs.CO and HI.vs.UT. Selecting FDR=0.01, 188 genes are selected.

LIMMA seems to be easy to use when more than two conditions need to be compared. The definition of a contrast matrix facilitates the performance of such cases. However, although t-test and SAM can also be programmed to this goal, it is a more complicated task. In order to compare LIMMA with t-test and SAM results, we focus on the same analysis done previously. This implies analysing the contrast HI.vs.CO. Selecting FDR=0.01, 47 genes were detected. We can observe that the results are very similar in number. Table 2.2 shows the number of genes that the different obtained results have in common. Differences could be related to the different approaches that each method involves. As we have seen in previous sections, the main difference in the approaches is the way in which the distribution needed is obtained to perform the contrasts. SAM uses permutations, LIMMA the empirical bayes approach and the classical *t*-test uses the theoretical *t*-Student distribution.

	t-test	SAM	LIMMA
t-test	**44**	-	-
SAM	21	**55**	-
LIMMA	24	31	**47**

Table 2.2: Comparisons of the selections of genes obtained with *t*-test, SAM and LIMMA by comparing HI with CO doses.

Multifactor experiment

If we consider the two involved factors: time and treatment and their interaction, we can compute the corresponding ANOVA model for each gene. By analysing the statistical significance of any of the mentioned three effects with a FDR=0.01 we obtain 421 genes as statistically significant (without considering the interaction the solution is a set of 281 genes). We can observe the high increase of statistically significant genes with respect to the previous selections. This is due to the fact that ANOVA implies the testing of all possible differences between time, treatment and interactions.

The next step is introducing the gene factor in the ANOVA model. In our data, this is a factor with 2665 levels. This calculation is extremely memory demanding and was not possible to perform with available computer resources (Intel Core 2 Quad processor and 2046 MB of RAM memory).

Finally, we have applied the MAANOVA R-package, also available in BIOCONDUCTOR platform. As we have already mentioned in section 2.5.3, the main benefit of this method is that before applying the ANOVA model it performs a first step to estimate the dye and array effect jointly to all the data as a normalization step. We presume that this will not have an important effect on our data because it has already been normalized. Other interesting features of MAANOVA package is that it allows the control of the FDR and it uses an empirical distribution to carry out the statistical tests associated with the *F*-statistics. Considering the additive model (time + treatment) MAANOVA selected 411 genes by choosing a FDR=0.01. However, the program does not let us include the interaction between time and treatment in the model, as there were not enough degrees of freedom to do the corresponding tests. The estimation of global effects of the data consumes the degrees of freedom for the ANOVA model of interest. Comparing this result with the selection of genes obtained with the general ANOVA model without previously applied interaction (281 genes) we obtained an overlap of 279 genes.

Conclusions of inferential analysis

The application of inferential tools has offered us several lists of selected genes. Our intention here was to study the performance of the available packages and how they adapt to the characteristics of a multiple series time course dataset. The first conclusion is that comparisons are not fully possible since different methods treat data differently and different contrasts are tested in each case. *t*-test and SAM focus on double comparisons, LIMMA in multiple comparisons, and ANOVA and MAANOVA in multifactor comparisons. We used LIMMA in an example with a double comparison to compare its results with those obtained with SAM and t-test (Table 2.2). On the other hand, and as MAANOVA did not permit us to include the interaction between time and treatment in the model, we have neither considered this interaction in the ANOVA model to better compare the results between these techniques. Mainly, we have seen that results are very similar between techniques with the same hypothesis to test. The differences can be due to the different ways in which the empirical distribution needed in each method is estimated.

2.8 Discussion

In this chapter we have described the major techniques to deal with microarray data in different steps of the data analysis process focusing on visualization and inferential tools. By applying some of these techniques to microarray data we have discovered the main benefits and limitations of classical and new specific techniques.

With the application of the visualization tools, we have mainly seen the difficulties in obtaining an informative representation of all the available data. This is related to the multifactorial nature of the experiment considered. In these cases and if a different number of replicates per condition are present, profile-based representations of the data are hard to interpret. In spite of this, we have detected some interesting global effects in our data such as the dye effect and the differences in expression in high doses of bromobenzene. Moreover, we have seen that PCA is able to deal with the complete data, dissecting sources of variation and identifying genes related to these main effects. In any case, these selections of genes are obtained from descriptive methods without statistical significance to evaluate their interest.

With the application of inferential tools we have mainly analysed their use and performance. Different selections of genes have been obtained depending on the comparisons done and the method employed. Here we can see the importance that the number of comparisons has in the results. For this reason researchers must be clear and concise in biological questions they want to address with their microarray study when planning the experimental set-up. For instance, the illustrated real example is a Time-Course experiment that has been used with several techniques planning different comparisons. However, the most typical objective when dealing with Time-Course data is to compare the gene expression pattern through time for the different treatments. For this goal, neither of the techniques shown are adequate as they do not analyze the dynamics of the data. Following chapters address this important issue.

All these techniques, as with other microarray data analysis methods, provide us with lists of differentially expressed genes. The next step in microarray data analysis is the biological interpretation of the gene selection. For this, the available knowledge on gene function is extracted from public databases and combined with the statistical results. A widely used approach relies on the utilization of structured vocabularies to represent the functionalities associated with genes. The use of these controlled terminologies eases the quantification of gene-wise results and hypothesis-testing analysis from a functional perspective. The most widely used vocabulary to describe gene function is the Gene Ontology (GO, http://www.geneontology.org/). GO describes gene products by their molecular function, the biological process they participate in and the cellular component they localize, and it is a general schema that can be applied to describe any biological domain. Annex 8 includes the description of this ontology and the available methods to extract biological meaning from genomics data using the GO.

Chapter 3

Review of Statistical Analysis of Time Course Microarray Data

3.1 Introduction

The main goal of the research presented in this thesis is the development of new statistical methods which are able to identify patterns of gene expression variation in time course transcriptomics data. In this chapter we review the current research topics which are addressed by time series microarray experiments and the statistical methodologies that have been applied or specially developed for this type of data.

The classification of time series experiments presented in Chapter 1 considers three criteria: i) The length of the series: depending on the number of time points, experiments can be regarded as short time courses (3-6 time points) and long series (> 6 time points); ii) The number of series: experiments can evaluate time evolution in one series (single series) or investigate time associated differences along different series (multiple series); iii) Dependence on observations: measurements can be linked by the individual (sampling the same individuals at different moments), referred to as longitudinal or repeated measurement data, or be obtained from different and independent individuals at each time point, i.e. independent data. In longitudinal series the correlation between the measures for the same individual, called autocorrelation, must be taken into consideration, meanwhile in the independent series this problem does not exist. Additionally we can consider if there is replication at each experimental condition (time-point × series) or not and if the expected pattern should obey some kind of cyclic behaviour.

Time course experiments are used to address a wide variety of biological questions. Typically, time course data relate to the study of the dynamics of biological systems, developmental processes, gene regulatory networks and responses to stimuli. Depending on the type of study, the corresponding data will fit into one of the specific types described above.

For example, the study of periodic time courses such as the cell cycle is the most popular study of biological systems where regular periodic patterns are expected. These series are long, longitudinal and single, and normally non-replicated for each time-point. Spellman *et al*. (1998) and Cho *et al.* (1998) are examples of the study of the cell cycle system in yeast.

In developmental time series studies there is an interest in understanding the temporal profile of an organism in its natural state or under a specific condition. These series are also normally long, longitudinal or not, depending on the biological source, and are single but usually replicated. Examples of developmental studies are the works by Himanen *et al*. (2004) in *Arabydpsis thaliana* to characterize the early molecular regulation induced by auxin; the study by Cercós *et al*., 2006 on fruit ripening in citrus; or on organ development studies in Drosophila (Arbeitman *et al*., 2002 and Tomancak *et al*., 2002).

In gene network analysis the objective is the study of the interactions between genes. These studies are associated with single and long series of time-course data under a specific experimental condition. For these studies cell cycle time series microarray data has been used as Spellman *et al*. (1998) dataset, although there are also specific experiments designed for the study of gene networks as in Rangel *et al*. (2004). We have to mention that time course data is not the unique type of data used for the study of gene network interactions. Regulatory relationships can also be inferred using steady-state gene expression data. The steady-state data are obtained by altering specific gene activities, such as deleting or over-expressing genes. Bansal *et al.,* 2007 claim that time-series data contain less information than steady-state data for the study of gene networks. This is due to the fact that in steady-state data multiple perturbations of the cell are available while time series are measured following only few perturbations in time.

Finally, studies that evaluate responses to stimuli normally use, independent, short and multiple series. The goal of multiple series time course (MSTC) experiments is to analyse the differences in gene expression between the various experimental groups of interest: different treatments or doses of the same treatment to inspect their evolutions. These reactions or possible changes are usually expected in a predefined period of time included in the design, therefore these series are typically short. In Heijne *et al*. (2003) we can find the toxicogenomics experiment used in this thesis, where there are 5 experimental groups and 3 time-points.

The analysis of time course microarray data requires specific treatment to adequately deal with variable time in each case. The statistical analysis of microarray time course data has been reviewed by Bar-Joseph (2004) and Tai and

Speed (2005). Bar-Joseph presents the review considering four sections: experimental design, data analysis, pattern recognition and networks, discussing the challenges and proposed methods but focusing on the analysis of long series. Tai and Speed (2005) focused on clustering and gene selection techniques for developmental or non periodic time course series that are also long series.

In this chapter we review the existing and novel methodologies applied to the analysis of time course experiments in a broad sense. We classify statistical approaches according to the different possible discovery aims in associated studies: methods for clustering of gene expression patterns, methods for identifying differentially expressed genes and specific methods for the inference of gene regulatory relationships. We use this classification to emphasize the major current lines of research, although it does not match exactly with the biological purposes of the previously mentioned studies. This is due to the fact that clustering and gene selection methods can tackle the studies of biological processes (associated with long series) and also the studies of MSTC (normally short series), although the effectiveness of the techniques is not always the same in both cases. A large number of currently available methods are devoted to the clustering of gene expression patterns, and for the deciphering of gene regulatory networks. Curiously, few methodologies can be found that directly address the problem of finding statistical profile differences between experimental groups. The analysis methodologies developed in this thesis deal with this last case. Therefore we will mainly focus on it, after briefly reviewing the other topics.

3.2 Clustering

Clustering is the most frequently used multivariate technique to analyse gene expression data for the assumption that genes with similar expression profiles could be involved in similar biological processes. In the previous chapter we have described the different types of cluster analysis and their limitations with large multifactorial datasets. Here we mention the specific problem of clustering in microarray time course data.

The first applications of clustering techniques to microarray data also include applications to microarray time course data in the works of Spellman *et al.* (1998) and Cho *et al.* (1998) that have become classical experiments. The goal in these studies was to discover gene expression patterns related to the cell cycle and therefore used clustering techniques in long time series. Applying hierarchical clustering, the authors succeeded in identifying co-expressed genes associated with specific biological categories. These studies showed the biological importance of gene expression in time. Furthermore, these datasets have been used and continue to be used in many other statistical developments to show the performance of new algorithms and techniques dealing with time course microarray data.

However classical clustering methods are not the best choice for the analysis of time course data. As we have seen in previous chapter, when dealing with multilevel experimental factors the difficulty in extracting general conclusions about the effects of interest were pointed out as the main limitation of classical clustering methods. Obviously, as time can be considered as a multilevel factor, this problem also appears in microarray time course data. Furthermore, levels in time factor involve a specific order and magnitude that must be taken into consideration.

More recently, dedicated clustering algorithms have been envisaged where the particular temporal property of gene expression is considered. The continuous and recent publications in this field show that the development of this type of methodology is an active topic of research. There is still a need for efficient methods that exploit the nature of these data, taking into account the inherent between time-point relationships present in the observations. Several proposals have appeared that exploit different mathematical strategies. Here we cite some examples:

- Yeung *et al*. (2001) proposed to use Gaussian mixture models assuming that the data is generated by a finite mixture of underlying probability distributions such as multivariate normal distributions. In this sense they formulate a statistical model to select an adequate number of clusters and a suitable clustering method.

- Ramoni *et al*. (2002) introduced a Bayesian model-based clustering algorithm representing temporal profiles by autoregression equations and grouping the gene profiles with the highest posterior probability of having been generated by the same process.

- Luan and Li (2003) considered a mixed-effects model for time course gene expression data using B-splines. Under this model they apply the

Expectation-Maximization (EM) algorithm to cluster genes and demonstrate the use of the Bayesian inference criteria (BIC) to decide the number of clusters. Similar algorithms have been developed by Bar Joseph *et al.* (2003) by applying cubic splines.

- Schliep *et al*. (2003) used Hidden Markov Models to account for time dependency. They develop an iterative algorithm to assign genes to clusters maximizing the likelihood of clustering and models.

- Ma *et al*. (2006) introduced a method using a smoothing spline model to estimate the gene expression curves avoiding the application of the EM algorithm that, as they say, is costly for large-scale data.

- Kim *et al.* (2006) used Fourier series approximation to model periodic patterns of gene expression profiles and to apply the EM algorithm to cluster genes from their corresponding estimated Fourier coefficients.

Although these approaches have been successfully applied to long series, the effectiveness in short time course is not so clear because these algorithms require a high number of time-points to capture the evolution of the profiles. Moreover, there is an increasing tendency to do experiments with short series for which the developed clustering techniques are inadequate. For this reason, the most recent works in clustering time course microarray data try to deal with short series. For instance:

- Ernst *et al.* (2005) developed a cluster algorithm for short time series gene expression. This algorithm establishes representative profiles independently of the data, and then assigns genes to the profile that is most correlated and determines the most statistically significant profiles by using a permutation test. This method is implemented in the STEM package (Ernst and Bar-Joseph, 2006).

- Kim and Kim (2007) proposed an algorithm for cluster short series taking into account the replicates of each time-point. This algorithm is based on the evaluation of the statistical significance of the differences between each pair of adjacent time-points computed with a *t*-statistic. However, this method does not describe the dynamics of gene expression evolution.

Although both methodologies are specific for single short series, they do not include adaptations to multiple series. The series used in studies involving multiple conditions are normally short. The consideration of the differences between the same time-points through the different series is the most interesting way of clustering the genes and this is not specifically treated by conventional clustering methods.

In general, we can conclude that the specific clustering methods for time-course deals with aspects related to the time factor which are valid mainly for long series as they are based on modelling approaches that require a sufficient number of data-points to be estimated. However, few clustering studies deal with short series and there are no suitable methodologies for the extension to multiple series. Summing up, the mentioned approaches are efficient in finding groups of co-expressing genes but when the experimental set-up is complex (different numbers of series, replicates, dye-swaps, etc.) the evaluation of the results on the basis of the clustering can still be rather complicated. On the other hand, these clustering methods tend to equally weigh all samples while deriving gene partition, which could not be the most convenient approach when expression changes are located to a restricted period of the analyzed time course.

3.3 Gene regulatory networks

Another very interesting application of time course microarray experiments is the unravelling of gene regulatory networks. This type of analysis evaluates the interactions between genes and look for models to describe or predict the gene expression behaviour. Describing molecular processes implies not only the identification of the genes involved but also infers possible casual relationships and the sequence of gene action. These models have many applications, for instance by characterizing the gene expression mechanisms that cause certain disorders it would be possible to target those genes to block the progress of the disease. Several approaches have been proposed to describe the genetic regulatory networks from time series microarray data such as Bayesian networks, Boolean networks, differential equations and other mathematical models (for an overview see de Jong, 2002 and, more recently, Bansal *et al*., 2007). Lähdesmäki *et al.* (2006) stated that the two most often used large-scale modelling frameworks are Boolean and Bayesian networks. They demonstrate the relationships between both approaches that are very useful to extend the developed tools for both models.

Bayesian Networks (BNs) is a very effective method to describe interacting processes by inferring causal relationships from the derived models and efficiently handling noisy and missing data (Friedman *et al.*, 2000). However BNs can not construct cyclic networks. This limitation can be avoided by applying Dynamic Bayesian Networks (DBNs). Since Murphy and Mian (1999) proposed the use of DBNs for modelling time series expression data, many papers have appeared applying and discussing this approach. In Kim *et al*. (2003) the methodology of estimating gene networks from time series microarray data using discrete and continuous DBNs models is reviewed. However DBNs has been applied mainly to small datasets and the computational costs of the application of DBNs to the genome-wide dataset are very high. New approaches are trying to deal with this challenge. For example, Beal *et al.* (2005) build a DBN model considering hidden variables or unmeasured genes to simplify the network structure and Geier *et al*. (2007) exploited the benefit of using knock-out data and prior knowledge to reconstruct gene networks.

The Boolean network model considers that the genes of interest are deterministically predicted by the so-called input genes and the defined Boolean functions. However, modelling the uncertainty inherent to genetic regulation in the biological level by a deterministic model is not appropriate. To incorporate this uncertainty in the model Probabilistic Boolean networks (PBNs) have been developed (Schmulevich *et al*., 2002a). In a PBN each gene can have more than one Boolean function and there are probabilities associated with the possible network. Random gene perturbations can also be considered in this model (Schmulevich *et al*., 2002b). Recent works in PBNs try to study the steady-state probability distribution (Ching *et al.,* 2007 and Brun *et al*., 2007) which is critical to identify the influent genes in a network and understand how to control some of these genes.

Methods based on differential equations have been successfully applied to small networks (de Hoon *et al*., 2003; Bansal *et al.,* 2006), but their use in larger datasets are not appropriate. In these cases the underlying biological system is considered to be known to avoid computational costs. By doing this, unknown regulatory relationships are not estimated from the data and alternative modelling approaches have to be considered.

Most of the available methods for gene network reconstruction are normally applied to a single dataset of time-course data under a specific experimental condition. New trends to the analyses of networks propose data integration from multiple datasets. For example, many researchers are interested in combining microarray data with protein interaction data and binding site information. Works dealing with multiple datasets under different experimental conditions are considerated more adequate to meet the goal in gene networks. In this way, Shi *et al.* (2007) consider the task of combining diverse time series datasets for pairwise lagged regulatory relationship inference and Wang *et al*. (2006) combine multiple time-course microarray datasets from different conditions for inferring gene regulatory networks applying linear programming and a decomposition procedure.

In general, it can be concluded that although the task of unravelling the complete cellular gene regulatory network is still far from being solved, new approaches that consider hidden variables and combine different datasets and information sources are providing interesting progress into this challenging field.

3.4 Identifying differentially expressed genes

As we have already mentioned, the primary goal in many transcriptomic experiments is the identification of genes with a change in expression over the conditions of the study (see Chapter 2). In multiple TCM experiments the objective is to find genes with different trends in time across several experimental groups or series. The most widely used approaches in this context address this problem by applying general methods described in Chapter 2 designed for replicated microarray experiments for two or more independent groups such as the Student's *t*-test and its related algorithms SAM (Tusher *et al*., 2001) and LIMMA (Smyth, 2004). For instance, de Hoon *et al.* (2002) applied Student's *t*-test to select genes by making comparisons for each time point separately and then analysing gene expression evolution for selected genes by using regression splines. This type of approach, although conceptually easy, is based on pair-wise comparisons (e.g. between all consecutive pairs of time points, or all possible pairs of time points) and when applied to microarray time courses especially when multiple series are present, it might be tedious and ineffective to capture the dynamic nature of the temporal data.

A first possible approach to consider time variable is applying classical

ANOVA to independent time series or mixed-effect ANOVA to longitudinal data. ANOVA-models can easily study multilevel factors and their interactions, and by evaluating the associated *F*-statistic, the statistical significance of each effect can be determined. Due to the success of the results of the classical *F*-statistic relying on the assumptions of normality, homoscedasticity and independence of the measurements, some variants of this statistic have been developed. For instance, Park *et al*. (2003) proposed a permutation test based on the ANOVA-model which does not need the normality assumption to deal with independent time course experiments. Similarly, ANOVA variants of Cui and Churchill (2003) have been successfully applied to time series in Fischer *et al*. (2007) to background-normalized simulated data. As TCM experiments normally are designed experiments, the time levels are pre-fixed and equal and a sufficient number of replicates for each time-point can be obtained, and the application of the ANOVA model seems to be a suitable approach. However, when analysing models containing quantitative variables or experiments with unbalanced designs, traditional ANOVA procedures are not appropriate and specific modifications have to be incorporated.

Another approach could be to consider specific tests to perform contrasts about the mean vector on the different time-points to study differences in the evolution through time. In this line, Tai and Speed (2006) proposed one and two sample multivariate empirical Bayes statistics (the MB-statistic) using shrinking covariance estimates to make inference about the average vector of a gene's expression levels. The algorithm contrasts, for each gene, the null hypothesis of constant vector of means along the time component (invariability), to the alternative hypothesis of non-invariability. The MB-statistic can be used to rank genes in the order of evidence of non-constancy. The method has been implemented in the R language in the timecourse package for the study of longitudinal data of one, two or multi-sample problems. However this method, similarly to ANOVA-models, requires replications for each combination of time-point/experimental group for sensitive estimates and also synchronized sample times for multiple time series.

Bar-Joseph *et al*. (2003) obtained a selection of differentially expressed genes between two cell-cycle microarray datasets by computing a difference measure between the continuous representations of the two time series expression data using B-splines that are defined as linear combinations of a set of basis polynomials. Storey *et al.* (2005) also proposed the use of B-splines to fit the same dimensional model to each gene where the coefficients are estimated by applying standard least squares regression techniques. They develop their method to deal with both independent and longitudinal data. And by using bootstrap techniques to find the empirical distribution of the *F*-statistic they detect genes with changes in expression over time and rank them by their *p*-values. There is an implementation of this method in the software EDGE (Leek *et al*., 2006). B-spline based approaches seem to be one of the most suitable methods to represent the gene expression evolution. However they work well with long time series (>10) and their adequacy for shorter time course experiments is not clear.

Regression approaches appear to be a more straightforward and flexible solution for the analysis of this type of data. Regression methods treat time as a quantitative variable, and therefore not only differentially expressed genes can be detected, but changes in trends can also be discovered and their magnitude can be studied by analysing the coefficients of the model. A regression model approach was used by Xu *et al*. (2002) to identify differential gene profiles in an inducible transgenic model. Their model includes the time as variable and specific covariates to identify differences in expression between two series. This tailor-made approach was claimed to be useful to evaluate specific gene expression behaviour but it implies redefining the variables for other biological systems. However, the complexity of the model shows the thorough knowledge that the researchers have in the experiment. A simpler modelling was proposed in Liu *et al*. (2005) where a quadratic regression model with time variable is fitted for identifying changes in expression in a short single time-series. Based on a linear regression model Guo *et al.* (2003) constructed a variant of the robust Wald-Statistic for studying longitudinal data. Due to the fact that asymptotic distribution of the Wald-statistic is not adequate when the number of subjects is small, they propose to use the permutation methods developed in Tusher *et al.* (2001) and Pan *et al.* (2003) to make inferences. Recently, DeCook *et al.* (2006) applied regression models to the study of multiple series. The method is based on the *F*-statistic of several pre-defined regression models to choose the best model for each gene. The authors emphasize the advantage of no need for replication in each time-point condition to apply regression models. Although this statement is true, it can be misleading for a non-statistician user. We have to bear in mind that we need enough data to estimate the polynomial model for each particular treatment around the period of interest. Mathematically, to estimate the coefficients of a polynomial model of *p* degree, we need at least *p+1* time-points without the need of having replicates in each time-point. Obviously, the more replicates, the better estimations can be obtained.

3.5 Discussion

In this chapter we have reviewed the main statistical approaches and the related papers dealing with TCM. We have presented the methods classified in three categories responding to analysis purposes: co-expression (clustering), gene networks and differential expression (gene selection). We have seen that, both in clustering and gene selection approaches, the majority of the methodologies are developed for long series. However, there is an increasing tendency to use microarray technology to explore the response to different stimuli by using short series.

The wide variety of mathematical and statistical approaches applied to deal with this data led us to conclude that there are no general rules for the researcher to obtain the best choice to analyse his/her data. For instance, the ANOVA model is one of the most criticized methodologies, used in many papers as a comparative method to introduce a novel technique. It is argued that the ANOVA model does not take the temporal ordering into account. However, other authors highly recommend its use. For instance in Fischer *et al*. (2007) we can find a comparison of several tests for identifying genes in experiments with different types of normalization and ANOVA is recommended to analyse background-normalized data. We think that the application of the ANOVA model could be a suitable approach as TCM experiments normally are designed experiments, which means that time levels are fixed, and the number of replicates for each time-point is constant. However, when analysing models containing quantitative variables or experiments with unbalanced or longitudinal designs, traditional ANOVA procedures are not appropriate and specific modifications need to be incorporated.

The use of B-splines to represent the evolution of gene expression in long series seems to be effective and it is being used for clustering and gene selection purposes. However, to deal with short series the use of regression models is more adequate, especially when multiple series are present. Fitting regression models with multiple variables that can be correlated may cause multicollinearity problems. Taking this problem into account, we have developed **maSigPro** (microarray

Significant Profiles, Chapter 4 and also in Conesa *et al*., 2006), a general regression-based approach for the analysis of short, single or multiple microarray time series. The procedure is a two-step regression strategy with adjustable model parameters. First the method fits a global polynomial regression model with all the defined variables to each gene to pre-select differentially expressed genes, and second a variable selection strategy is applied to find more suitable models for each gene and to study the different profiles between genes trying to avoid the multicollinearity problem. Furthermore, this second step allows for the ranking of genes through the value of the *R*-squared statistic that is a measure of the goodness of fit model or from the *p*-value of its specific stepwise model.

Most of the methods described above are univariate in the way they approach analysis: one gene is considered at a time. This implies that many informative correlation structures within the data are simply ignored. The **ASCA-genes** methodology presented in Chapter 5 (published in Nueda *et al.,* 2007) is an approximation to a multivariate consideration of time course data. This strategy combines ANOVA-modelling and a dimension reduction technique to discover the general targeted trends in time-course. The methodology is valuable for identifying the principal and secondary responses associated with the experimental factors and spotting relevant experimental conditions, and for identifying differentially expressed genes that follow specific variation patterns. This method is particularly interesting for the way it treats noise. ASCA can be used to extract the latent structure of the data and to filter out the systematic noise, thereby enhancing the statistical power of the maSigPro methodology. In Chapter 6 we describe this particular application of **ASCA** as a **filtering** method. There are other methods that consider the correlation structures to estimate the underlying distribution of the gene expression, such as the mentioned timecourse method (Tai and Speed, 2006). However, these methods focus on the average vectors and do not take into account the separation between the sources of variation involved in the experiment.

The ultimate purpose of microarray experiments is to generate biological knowledge. However, the results of the statistical methods for the analysis of microarray data are lists of differentially expressed genes (d.e.g.). A more difficult challenge for researchers is to understand the biological phenomena behind these gene lists. *A priori* knowledge stored in public databases on gene functions is habitually used along the statistical treatment of the data to provide biological meaning to transcriptome analysis. The most widely applied method is to assess the enrichment of functional categories within the group of the d.e.g. This analysis is also applied in time course microarrays analysis. However, there are no methods which incorporate the *a priori* knowledge in an efficient way into the dynamic of the time series. In Chapter 7 we propose a novel strategy to deal with this topic. We apply data analysis methods to groups of genes belonging to the same biological functional category. In this sense we have applied maSigPro, ASCA and PCA to develop three new methodologies called **maSigFun**, **PCA-maSigFun** and **ASCA-functional**.

Chapter 4

maSigPro: a Method to Identify Significantly Differential Expression Profiles in Time-Course Microarray Experiments

Conesa A., Nueda, M.J., Ferrer, A. and Talón, M. (2006) maSigPro: a Method to Identify Significantly Differential Expression Profiles in Time-Course Microarray Experiments. Bioinformatics, 22 (9), 1096-1102.

4.1 Introduction

Multiple series time-course (MSTC) microarray experiments are useful approaches for exploring biological processes. In this type of experiments, the researcher is frequently interested in studying gene expression changes along time and in evaluating trend differences between the various experimental groups. The large amount of data, multiplicity of experimental conditions and the dynamic nature of the experiments poses great challenges to data analysis.

As we have seen in previous chapter most of the currently available methods are devoted to the identification and clustering of gene expression patterns, and for the deciphering of gene regulatory networks. However, few methodologies can be found that address the problem of finding statistical profile differences between experimental groups.

In this chapter, we propose a statistical procedure to identify genes that show different gene expression profiles across analytical groups in time-course experiments. The method is a two-regression-step approach where the experimental groups are identified by dummy variables. The procedure first adjusts a global regression model with all the defined variables to identify differentially expressed genes and secondly, a variable selection strategy is applied to study differences between groups and to find statistically significant different profiles. The proposed method has been successfully applied to several experiments. In this work the procedure is illustrated both on simulated data and a public domain toxicogenomics dataset.

The method has been implemented in the statistical language R and is freely available from the Bioconductor contributed packages repository and from the personal webs of the authors (http://www.ua.es/personal/mj.nueda or http://bioinfo.cipf.es/aconesa) and it has also been included in the gene expression pattern analysis suite (http://www.gepas.org). In Annex 1 we have included a scheme with the main functions of the maSigPro package that summarizes the process to obtain the results.

4.2 Methods

4.2.1 Definition of the model

In the problem we are considering there are normally two or more variables of interest. One of them is typically the time, which is a quantitative variable (in the type of experiments considered for this approach, time is usually the independent variable, however the methodology would accept as well other experimental continuous variables, such as a quantified physiological parameter). The others variables are usually qualitative variables (e.g. different treatments, strains, tissues, etc.) and represent the experimental groups for which temporal gene expression differences are sought. For clarity in the exposition, only one qualitative variable or factor will be considered here.

Let be *J* experimental groups described by the qualitative variable evaluated at I time points for each particular condition *ij* (*i*=1,…,I and *j*=1,…,*J*). Assume that gene expression is measured for *N* genes in R_{ij} replicated hybridizations.

We define *J*-1 dummy variables (binary variables) to distinguish between each group and a reference group (Table 4.1).

Group	D_1	D_2	**…**	D_{J-1}
1 (Ref.group)	0	0	0	0
2	**1**	0	0	0
3	0	**1**	0	0
…	0	0	**…**	0
J	0	0	0	**1**

Table 4.1: Definition of experimental groups with dummy variables.

Let x_{ijr} denote the normalized and transformed expression value from each gene in the situation *ijr* (*r*=1,…,R_{ij}). To explain the evolution of *x* along the time (*T*) we consider the following polynomial model, where simple time effects and interactions between the dummies and the time have been modelled. In principle, the maSigPro methodology allows a polynomial model of I-1 degree as the model described in Equation (4.1).

$$\begin{aligned} x_{ijr} &= \beta_0 + \beta_1 D_{1ijr} + \ldots + \beta_{(J-1)} D_{(J-1)ijr} \\ &\quad + \delta_0 T_{ijr} + \delta_1 T_{ijr} D_{1ijr} + \ldots + \delta_{(J-1)} T_{ijr} D_{(J-1)ijr} \\ &\quad + \gamma_0 T_{ijr}^2 + \gamma_1 T_{ijr}^2 D_{1ijr} + \ldots + \gamma_{(J-1)} T_{ijr}^2 D_{(J-1)ijr} \\ &\qquad \ldots \\ &\quad + \lambda_0 T_{ijr}^{I-1} + \lambda_1 T_{ijr}^{I-1} D_{1ijr} + \ldots + \lambda_{(J-1)} T_{ijr}^{I-1} D_{(J-1)ijr} + \varepsilon_{ijr} \end{aligned} \tag{4.1}$$

$\beta_o, \delta_o, \gamma_o \ldots \lambda_0$: are the regression coefficients corresponding to the reference group.

β_j, δ_j, $\gamma_j \ldots \lambda_j$: are the regression coefficients that account for specific differences (linear, quadratic, cubic,etc.) between the (*j*+1)-th group profile and the first group (reference) profile, *j*=1,...,*J*-1.

ε_{ijr} : is the random variation associated with each gene in each hybridization *ijr* owing to all sources other than those that have already been incorporated into the model.

This model defines implicitly as many models as experimental groups. For example, the model for the first group 1 is $x_{i1r} = \beta_0 + \delta_0 T_{i1r} + \gamma_0 T_{i1r}^2 + \ldots + \lambda_0 T_{i1r}^{I-1} + \varepsilon_{i1r}$, since in this group all the dummies are 0; and for the second group is $x_{i2r} = (\beta_0 + \beta_1) + (\delta_0 + \delta_1) T_{i2r} + (\gamma_0 + \gamma_1) T_{i2r}^2 + \ldots + (\lambda_0 + \lambda_1) T_{i2r}^{I-1} + \varepsilon_{i2r}$. In this example β_1, δ_1, $\gamma_1 \ldots \lambda_1$ measure the differences between the second and first (reference) groups related to linear, quadratic, etc. and (*I*-1)-th time order effects; respectively.

4.2.2 First regression model: gene selection

The first step of the maSigPro approach applies the least-squares technique to estimate the parameters of the described general regression model for each gene. This means that we are testing the following null and alternative hypotheses:

$$\begin{aligned} &H_0 : \beta_1 = \ldots = \beta_{J-1} = \delta_0 = \delta_1 = \ldots = \delta_{J-1} = \gamma_0 = \gamma_1 = \ldots = \gamma_{J-1} = \ \ldots \ = \lambda_0 = \lambda_1 = \ldots = \lambda_{J-1} = 0 \\ &H_1 : \exists j \,/\, \beta_j \neq 0 \left(j = 1,\ldots,J-1 \right) \vee \delta_j \neq 0 \vee \gamma_j \neq 0 \vee \ldots \vee \lambda_j \neq 0, \ \left(j = 0,\ldots,J-1 \right) \end{aligned} \tag{4.2}$$

This first analysis generates *N* ANOVA tables as shown in Table 4.2, one for each gene. A gene with different profiles between the reference group and any other experimental group will show some statistically significant coefficient, and its corresponding regression model will be statistically significant. The *p*-value associated with the *F*-Statistic in the general regression model is used to select significant genes. This *p*-value is corrected for multiple comparisons by applying the linear step-up (BH) false discovery rate (FDR) procedure (Reiner *et al.*, 2003). Therefore, genes with a FDR lower than a predetermined threshold will be selected.

Source	**Sum of squares (SS)**	**Degrees of freedom**	**Mean square error**	***F*-Statistic**
Regression (R)	$SS_R = \sum_{ijr} (\hat{x}_{ijr} - \bar{x})^2$	p	$\frac{SS_R}{p}$	$\frac{SS_R / p}{SS_E / (\sum_{i,j} R_{ij} - (p+1))}$
Error (E)	$SS_E = \sum_{ijr} (x_{ijr} - \hat{x}_{ijr})^2$	$\sum_{i,j} R_{ij} - (p+1)$	$\frac{SS_E}{\sum_{i,j} R_{ij} - (p+1)}$	
Total (T)	$SS_T = \sum_{ijr} (x_{ijr} - \bar{x})^2$	$\sum_{i,j} R_{ij} - 1$		

Table 4.2: ANOVA table. $\hat{x}$ is the predicted expression value, $\bar{x}$ is the average expression value and *p* is the number of variables in the model, (polynomial order *+1)J-1=IJ-1.*

4.2.3 Second regression step: variable selection

Once statistically significant gene models have been found, the regression coefficients of the models can be used to identify the conditions for which genes shows statistically significant profile changes. To do this, a new model is obtained only for selected genes, applying a variable selection strategy (stepwise regression, Draper and Smith, 1998). Stepwise regression is an iterative regression approach that selects from a pool of potential variables the "best" ones (according to a specified criterion) to fit the available data. In this process, the statistical significance of the regression coefficients of the variables present in the model at each iteration is computed and only those variables with a *p*-value under a given threshold (type I risk) are maintained. In this case, applying FDR for multiple comparisons is not easy due to the fact that *p*-values associated with each coefficient vary as the model evolves. Therefore, we apply a threshold that must be fixed by the researcher. We recommend correct the desired level of significance for the total possible number of variables in the model. The variables included in these new models will be those that indicate the differences in profiles. The maSigPro package provides different types of stepwise regression: backward, forward, stepwise backward and stepwise forward. This variable selection approach has a double effect: on one hand it provides the significant differences between experimental groups, and on the other hand, it generates an adequate regression model for the data. This implies that for each gene and experimental group, polynomial regressions of different degree (up to the maximum initially given in the formulation of the model) can be obtained. The method will therefore generate a matrix with so many rows as significant genes and so many columns as parameters in the complete regression model (Equation (4.1)). This results matrix contains information (estimated coefficient and its *p*-value) for those variables that remained in the model of each gene. Table 4.3 is an illustrating example of such a results matrix.

	1	***2***	***3***	**…**	***J***	***1***	***2***	***3***	**…**	***J***		***1***	***2***	***3***	**…**	***J***
geneID	β_0	β_1	β_2	…	β_{J-1}	δ_0	δ_1	δ_2	…	δ_{J-1}	…	λ_0	λ_1	λ_2	…	λ_{J-1}
	Intercept	D_1	D_2	…	D_{J-1}	**Time**	**Time×**D_1	**Time×**D_2	…	**Time×**D_{J-1}	…	**Time**$^{I-1}$	**Time**$^{I-1}$**×**D_1	**Time**$^{I-1}$**×**D_2	…	**Time**$^{I-1}$**×**D_{J-1}
gene1	β_{01}	β_{11}	NA	…	NA	NA	δ_{11}	NA	…	NA	…	NA	λ_{11}	NA	…	$\lambda_{(J-1)1}$
gene2	β_{02}	NA	NA	…	NA	δ_{02}	NA	δ_{22}	…	$\delta_{(J-1)2}$	…	λ_{02}	NA	λ_{22}	…	NA
gene3	NA	NA	NA	…	$\beta_{(J-1)3}$	NA	δ_{13}	NA	…	$\delta_{(J-1)3}$	…	NA	λ_{13}	NA	…	$\lambda_{(J-1)3}$
gene4	NA	β_{14}	β_{24}	…	$\beta_{(J-1)4}$	δ_{04}	NA	δ_{24}	…	$\delta_{(J-1)4}$	…	λ_{04}	NA	λ_{24}	…	NA
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
geneN	β_{0N}	NA	β_{2N}	…	$\beta_{(J-1)N}$	δ_{0N}	NA	δ_{2N}	…	NA	…	λ_{0N}	NA	λ_{2N}	…	$\lambda_{(J-1)N}$

Table 4.3: Results matrix of regression coefficients for the variable selection fit. Genes are shown in rows and model parameters in columns. Regression coefficients exclusively associated with the same experimental group are labelled with the same number. NA value for regression coefficients indicates that the variable was not statistically significant for that gene (under a given threshold, type I risk).

This matrix provides the framework for selecting significant genes for each variable of the complete model and for each experimental group. For example, to find genes that have significant differences in group 2 respect to the reference group, those genes having statistically significant coefficients for the variables associated with the Dummy1 (D_1, Time×D_1,…, Time^{I-1}×D_1) must be selected, i.e. genes which have a significant coefficients (column labelled as 2 in Table 4.3). Additionally, the study of individual model variables allows focusing on the evaluation of specific pattern differences. For example, the analysis of the regression coefficients of the variable Time×D_1 allows the classification of genes for their different behaviour in the linear model component (i.e. induction or repression) of group 2 with respect to the reference group. The maSigPro package includes functions to easily perform different types of gene selection at this stage.

Until now, the goodness of fit (*R*-squared) of the new models has not been considered. This means that all significant genes are selected genes. The researcher might however be interested only in genes with clear trends as this may reflect biologically meaningful behaviours. In such case, maSigPro allows an additional gene selection step based on the *R*-squared value of the second regression model.

4.2.4 Visualization

The maSigPro package provides a number of functions for the visual analysis of the results. Individual plots of expression profiles by experimental group can easily be generated for each significant gene. Computed regression curves can also be superimposed to visualize the modelling obtained for the data. When the number of selected genes is large, cluster algorithms may be used to split the data into groups of similar expression patterns. maSigPro incorporates a number of traditional clustering algorithms to do so. These algorithms typically use gene expression data to compute clusters. Additionally, maSigPro provides a clustering alternative that uses the estimated regression coefficients rather than the original data. This option will group genes on the basis of their statistically significant profiles changes, discarding the noise of the data that has been removed by the estimated model. Once clusters have been obtained, maSigPro displays both the continuous expression profile along all experimental conditions and the average expression profile by experimental group for each cluster. The first representation helps to analyze the homogeneity of the clusters while the second provides a useful visualization of the between-groups differences for the genes of each cluster.

4. 3 Results

4.3.1 Case 1: Toxicogenomics dataset.

The maSigPro method has been applied to the analysis of the published dataset described in Chapter 2. As we have mentioned, it is a toxicogenomics study where the effect of the hepatotoxicant brombenzene in rats was studied. In this example there are 5 experimental groups (j=1,…,5), 3 time points (i=1,2,3), 2 or 6 observations, r=1,…,R_{ij} (2 or 6) for each case ij, and 2665 genes (n=1,…,N). The corn oil group was taken as reference group as this provides the true control for the treatments. Consequently, we defined four dummy variables D_{UT}, D_{LO}, D_{ME} and D_{HI} to introduce in the model the experimental groups in an analogous way as described in Table 4.1. We considered for each gene the model given in Equation (4.3) where linear and quadratic time effects and their interactions with the dummies have been modelled.

$$\begin{aligned} x_{ijr} = {} & \beta_0 + \beta_1 D_{(UT)ijr} + \beta_2 D_{(LO)ijr} + \beta_3 D_{(ME)ijr} + \beta_4 D_{(HI)ijr} + \\ & \beta_5 T_{ijr} + \beta_6 D_{(UT)ijr} \times T_{ijr} + \beta_7 D_{(LO)ijr} \times T_{ijr} + \beta_8 D_{(ME)ijr} \times T_{ijr} + \beta_9 D_{(HI)ijr} \times T_{ijr} + \\ & \beta_{10} T_{ijr}^2 + \beta_{11} D_{(UT)ijr} \times T_{ijr}^2 + \beta_{12} D_{(LO)ijr} \times T_{ijr}^2 + \beta_{13} D_{(ME)ijr} \times T_{ijr}^2 + \beta_{14} D_{(HI)ijr} \times T_{ijr}^2 + \varepsilon_{ijr} \end{aligned} \tag{4.3}$$

Applying maSigPro to these data a total of 155 significant genes were selected at a FDR=0.01 and *R*-squared threshold=0.6. The FDR gives the expected number of false positives among the selected genes, in this case 1.5, and the *R*-squared criterion selects for genes which are statistically well modelled. All these 155 genes showed statistically significant differences in the comparison between the high dose and the CO reference group. Out of these, 28 and 91 genes showed also significant differences at the low and medium dose, with respect to the reference CO group, respectively.

Visualization of these expression profiles differences can be performed through the clustering and plotting functions available in the package. A more directed visualization to specific gene expression behaviours is also possible by using the values of the estimated regression coefficients. For example, we identified genes having either an induction or repression response upon the HI treatment (with respect to the reference group) by selecting genes with positive or negative values on the estimation of regression parameter, respectively. This variable gives the slope difference between HI and CO groups when variable $D_{HI}\times T^2$ is not significant, or the slope difference at Time=0 between these two groups when the quadratic term is significant. Thus, we obtained 59 genes grouped in the "induction response" and 86 genes in the "repression response". Each of these groups was then subjected to clustering for visualizing the differences between experimental groups (Figure 4.1 and Figure 4.2). Figure 4.1 shows the experiment-wide gene expression profiles, whereas Figure 4.2 gives the mean profile by groups of each cluster. Figure 4.1 is useful to evaluate the homogeneity of the obtained clusters but the actual profile differences between experimental groups can be better analyzed on Figure 4.2.

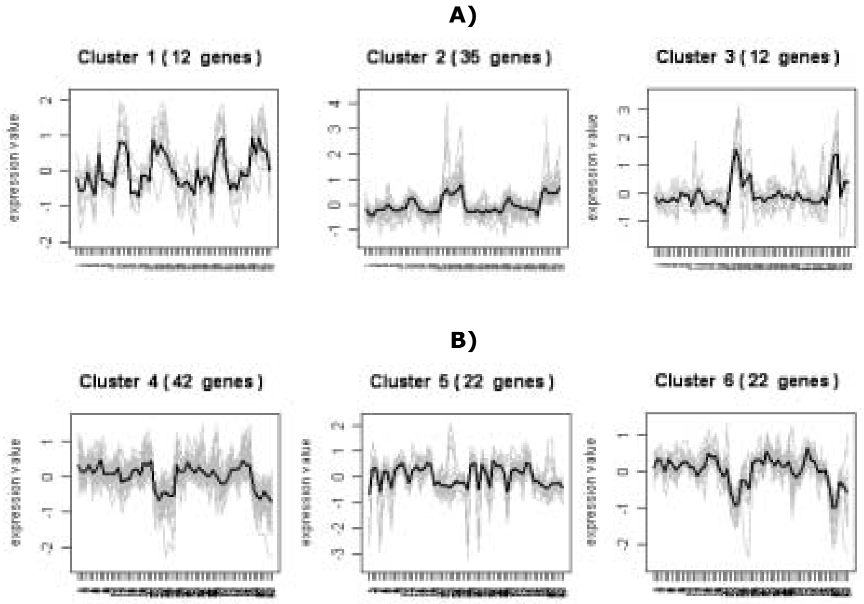


Figure 4.1: Data visualization by cluster analysis. The gene expression profile along all 54 experimental conditions (see Table 2.1 for array labelling) is displayed. A) Genes with a positive $D_{HI} \times T$ coefficient (induced). B) Genes with a negative $D_{HI} \times T$ coefficient (repressed). Average expression profile is showed (black line) together with the expression profiles of the genes in the cluster (grey lines).

Functional classification of the significant genes showed a high proportion of genes involved in functions related to a toxicological response. Cluster 1 contained a high number of genes related to drug-response, while clusters 2 and 3 were populated by genes involved in protein synthesis, and degradation and maintenance of cell structure. Among the down-regulated genes, many were participating in acute-phase, fatty acid metabolism or had oxidative properties. Interestingly, cluster 4 contained many retinol-signalling and tumorogenesis genes, most of them not found in the original paper analysis (Heijne *et al.*, 2003). Overall, maSigPro detected 104 new genes that showed statistically significant differences between experimental groups compared to Heijne *et al.*' results. These authors used a two-tailed Student´s *t*-test per gene on the comparison BB treated (joining LO, ME and HI experimental groups) and CO control with no FDR correction.

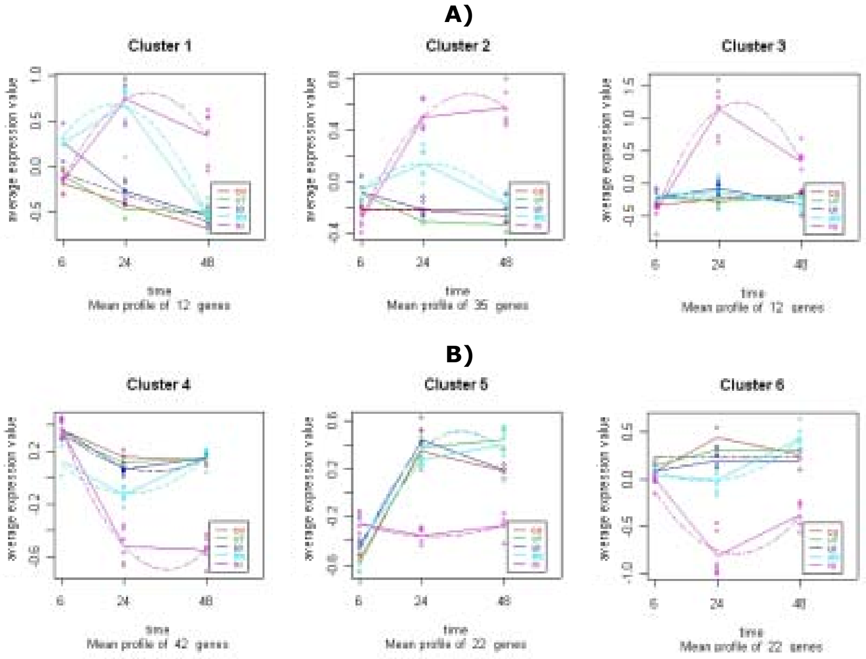


Figure 4.2: Data visualization by cluster analysis. Each plot shows the cluster average expression profile by experimental group. A) Genes with a positive $D_{HI} \times T$ coefficient (induced). B) Genes with a negative $D_{HI} \times T$ coefficient (repressed). Dots show actual expression values. Solid lines have been drawn joining the average value of gene expression at each time point for each experimental group. Fitted curves are displayed as dotted lines.

To further evaluate the performance of maSigPro we compared our results with those generated by using the R package LIMMA. We chose LIMMA for this comparison for being a widely used methodology for the statistical analysis of microarray experiments. LIMMA performs a linear fit of the data on the experimental variables and allows setting multiple contrasts for the comparison of the experimental conditions. When applying LIMMA to the bromobenzene study, it became immediately notorious the high number of pair-wise comparisons that had to be set to mimic the maSigPro analysis. We focused on the analysis of gene expression differences between the High dose and the CO group. This implied to analyze the contrasts HI_6h.vs.C0_6h., (HI_24h.-HI_6h.) vs. (CO_24h.-CO_6h.), (HI_48h.-HI_24) vs. (CO_48-CO_24h.) and (HI_48h.-HI_6) vs. (CO_48-CO_6h.), and gather the results in one unique gene list. Using this approach, LIMMA selected 63 significant genes at an FDR of 0.01 while maSigPro detected 155. A total of 53 genes were selected by both methodologies, 10 additional genes were called significant by the LIMMA approach and 102 were solely found by maSigPro. LIMMA exclusive genes showed a greater data variability than those selected with maSigPro. These genes were actually found significant also by maSigPro at the first regression fit but had low *R*-squared values and were consequently not selected. On the other hand, genes detected with maSigPro and not with LIMMA show clear differences between the high doses and corn oil groups. The reasons for the different detection might be due to the different criterion for significance between maSigPro and LIMMA. LIMMA applies FDR on the estimated coefficients while maSigPro controls false positives on the significance of each gene model.

4.3.2 Case 2: Simulated data.

Since "live" experimental data cannot tell which genes are truly differentially expressed, we evaluated the power detection of maSigPro on a simulated dataset resembling the structure of the bromobenzene experiment. The dataset contained 600 genes with profile differences which could be classified into 3 expression patterns; single group continuous induction (A), single group transitory repression (B) and differential multi-group induction (C) (see examples of these situations in Figure 4.3). Additionally, there were 2000 flat profile genes without differences between experimental groups, making a total of 2600 genes. The replicates for each gene were produced as independent observations from a distribution $N(\mu_{ijn},\sigma^2_{ijn})$, i=1,2,3; j=1,…,5; n=1,…,2600. The data was generated considering higher variance to the cases with high gene expression and introducing outliers. We performed 100 independent simulations and computed the number of false positives detected with both maSigPro and LIMMA using an FDR of 0.05. Our results show that maSigPro was successful in controlling the number of false positive at the given FDR, while LIMMA exceeded this threshold in many cases (see Figure 4.4). The difference between the FDR obtained with LIMMA and maSigPro is statistically significant at 95% confidence level (0.02±0.002). Furthermore, no type-II errors (false negatives) were present within maSigPro solutions whereas 16% of the simulations analyzed by LIMMA did contain at least one false negative. Further analysis of maSigPro estimates showed that all the significant effects were included in the models and there were approximately 2.8% of the significant genes with some additional variable in the model, which indicates an adequate control of the false positives at this step of the analysis.

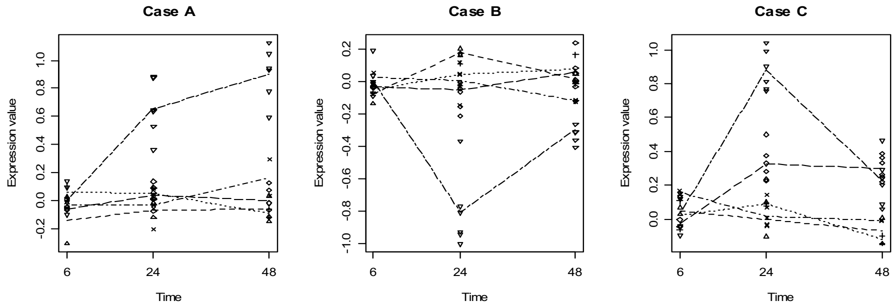


Figure 4.3: Three simulated data examples. Different points correspond to the data at different groups. Different lines join mean expression values at each GroupxTime combination for the five different groups.

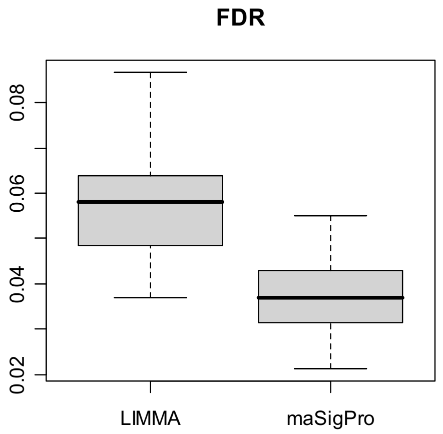


Figure 4.4: Results on simulated microarray data. Box-plots summarizing FDR applying LIMMA and maSigPro to 100 simulated datasets.

4. 4 Discussion

In this work we present a statistical procedure to identify genes that have different expression profiles among experimental groups in microarray time-course experiments. The method is a two-step regression approach where experimental groups are defined by dummy variables. The first regression fit adjusts a global model and serves to select differentially expressed genes, while in the second step a variable selection strategy is applied to identify statistically significant profile differences between experimental groups. The way variables are defined in the model provides a versatile procedure for studying specific pattern differences among experimental groups and genes.

The choice of using a two-regression steps approach instead of fitting a unique model had a number of reasons and consequences. In principle, it is possible that a model including all the available variables would be statistically significant but would not have any statistically significant coefficient. This situation is possible in multicollinear scenarios. Therefore, it appears more adequate to apply a variable selection strategy to obtain gene-specific models containing only statistically significant variables and where correlated variables had been removed. However, this way of building the models is not very recommendable for the purpose of selecting significant genes. Firstly because the time necessary to obtain models by steps is much longer than the time needed to estimate a unique model. With datasets including thousands of genes this can become highly time consuming and practically unfeasible. Consequently, it appears much more effective to first fit a global model for all genes, use the ANOVA *p*-values of these global models to find significant genes and apply then stepwise variable selection fit to only this selection of genes. A second reason is based on the out-come of some studies that have shown that regression models created by stepwise approaches yield *p*-values biased towards low values (Harrell, 2002). These *p*-values do not have a proper meaning and their appropriate correction is still a problem. We checked how this circumstance would affect gene expression analysis by applying both the maSigPro approach and solely stepwise regression, with their corresponding *p*-value corrections for multiple comparisons, on different datasets. This experiment showed that Harrell´s assertion was true when the goodness of fit of the models (*R*-squared) was not considered, but as the *R*-squared of the estimated models increased, normally above 0.5, both approaches converged (see the case of the Bromobenze data in Figure 4.5). When gene selection uses a high *R*-squared threshold (e.g. 0.6 as used in this example), both approaches yield similar results, but the two-step procedure is computationally less intensive.

Gene selection based on the goodness of fit criterion (high *R*-squared) provides the possibility of selecting genes for which good models could be obtained. This can be in many cases a very interesting option when the researcher is mainly interested in finding biologically meaningful expression trends and in detecting evenly meaningful profiles differences. In this case, high *R*-squared gene models might be successful in capturing these behaviours. In other cases, the aim of the analysis may be the detection of any possible gene expression difference and low *R*-squared models showing some significant coefficients could be allowed. The knowledge of the researcher and the objectives of study in each experiment will help to take a decision about the *R*-squared threshold to use.

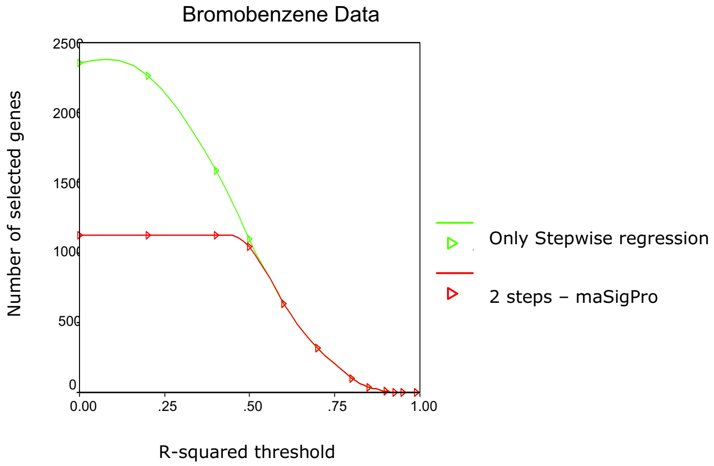


Figure 4.5: Gene selection evolution for maSigPro and only stepwise for different levels of *R*-squared.

Regression approaches rely on a number of assumptions such as independence of the observations, homoscedasticity and normality. Since microarray data might not always meet these requirements, validation of the models would be pertinent. In the simulation study maSigPro successfully detected the existing gene expression profile differences despites the heteroscedasticity and influential values present in the data, indicating that the method is valid for the detection of profile differences in such cases. The maSigPro package provides a series of tools for evaluating the presence of influential data, which is given as one of the results of the analysis process.

In the toxicogenomics example analysed, observations were independent because each rat was removed from the experiment after RNA extraction and therefore the measurements had been obtained from different individuals. However, in experiments where gene expression is measured over time on the same subjects the assumption of independence of the observations will not be satisfied. In these cases it would be more recommendable to analyse the data via repeated measures or longitudinal studies (Vittinghoff *et al.*, 2005).

Although we have presented the method with (I-1)-th time order effects, in experiments where simple gene expression responses are sought or expected and a reduced number of time points are evaluated (< 6), quadratic or cubic models would usually be sufficient to analyse the data (note that the polynomial degree is always a maximum, the variable selection step will create in the end models that "best" fit the data). As already discussed above, it is likely that the researcher is mostly interested in genes which follow biologically meaningful patterns like induction/repression, saturation kinetics, or transitory responses, which can easily be modelled with low degree polynomials. In experiments expanding a larger number of time points, more complex expression patterns could be expected. In this case, simple polynomial models may fail to capture the evolution of gene expression. For such scenarios a piece-wise regression or splines regression approach could be applied (Marsh and Cormier, 2001). The inclusion of a splines regression alternative within the maSigPro approach is in principle quite straightforward, as it would simply imply to introduce new dummy variables to define time intervals. The feasibility of this strategy will be addressed in future studies.

The results presented in this work show that maSigPro is a powerful method for the analysis of time course microarray data. The method detects significant profiles differences without carrying out tedious multiple pair-wise comparisons, allowing for unbalanced designs and heterogeneous sampling times. The variable definition of the models does permit not only to find genes with temporal expression changes between experimental groups, but also to analyze the magnitude of these differences. The proposed method can easily be extended to include additional variables (e.g. dye) or reduced by removing variables (e.g. to study the evolution over time for one unique group). The availability of the maSigPro methodology as an R package makes this analysis approach easily accessible to the research community.

Chapter 5

ASCA-genes: Discovering gene expression patterns in Time Course Microarray Experiments by ANOVA-SCA

Nueda, M.J.; Conesa, A.; Westerhuis, J.A.; Hoefsloot, H.C.J.; Smilde, A.K.; Talón, M. and Ferrer, A. (2007) Discovering gene expression patterns in Time Course Microarray Experiments by ANOVA-SCA. Bioinformatics, 23 (14), 1792-1800.

5.1 Introduction

Designed microarray experiments, as multiple series time-course (MSTC) microarray experiments, are used to investigate the effects that controlled experimental factors have on gene expression and learn about the transcriptional responses associated with external variables. In these datasets signals of interest coexist with varying sources of unwanted noise in a framework of (co)relation among the measured variables and with the different levels of the studied factors. Discovering experimentally relevant transcriptional changes require methodologies that take all these elements into account.

Multiple series time-course (MSTC) microarray experiments are designed experimental set-ups in which gene expression is measured at various points of a given time interval on samples that correspond to different levels of other experimental factor(s), such as treatment, tissue or strain. As in many other functional genomics datasets, MSTC data contain information about a large number of variables (genes) measured on a relatively small number of samples (experimental conditions). The analysis of this kind of data is usually addressed either as the identification of co-expressing genes clusters, or as the detection of differentially expressed genes. Traditional clustering methods have been applied to the analysis of microarray time course data (Spellman *et al*., 1998, Lukashin and Fuchs, 2001) and more recently dedicated clustering algorithms have been developed that particularly consider the temporal property of gene expression (Bar-Joseph *et al*., 2003). These approaches are efficient in finding groups of co-expressing genes but when the experimental set-up is complex (different numbers of treatments, replicates, dye-swaps, etc.) the evaluation of the results on the basis of the clustered patterns can become a rather complicated task. Furthermore, clustering methods tend to equally weight all samples while deriving gene partition, which could not be the most convenient approach when expression changes are only present in a subset of conditions. A second type of methodologies aims at the identification of genes whose expression vary across experimental conditions in a statistically significant manner (Conesa *et al*., 2006, Storey *et al*., 2005, Tai and Speed, 2006). These approaches are frequently univariate and as such do not provide the adequate framework for generating a global understanding of the information contained in the data.

In general, when managing large amounts of noisy but correlated data, such as in the case of microarray experiments and especially when various experimental factors and levels combine, data analysis can greatly benefit from approaches that generate information about major and secondary patterns of variability present through the experimental set-up. Such explorative approaches are effective in providing a global understanding of the effects that the different factors cause on gene expression, help in identifying most relevant experimental conditions and can shed light on how to address subsequent statistical analysis, e.g., which would be the contrasts of greatest interest. Dimensionality reduction techniques, such as Principal Component Analysis (PCA) are suited for explorative and summarizing analyses in these datasets as they are able to model the relationships between genes by analysing the correlation structure of the data (Quakenbush, 2001). PCA in microarray data was introduced by Raychaudhuri *et al*., (2000) for the analysis of Chu's yeast sporulation dataset (Chu *et al*. 1998). These authors showed the basis of Principal Component (PC) interpretation in the gene expression framework indicating the possibilities and difficulties of using PCA as a clustering technique. Other studies have applied PCA and related dimension reduction techniques in microarray data analysis for the purpose of classifying samples (Landgrebe *et al*., 2002; Nguyen and Rocke, 2002; Dai *et al*., 2006), finding co-expressing genes (Yeung and Ruzzo, 2001) or identifying odd data (Hilsenbeck *et al*., 1999). Methods have been developed to introduce statistical significance in the choice of PCs or for selecting relevant genes (Landgrebe *et al*., 2002, Roden *et al*., 2006). However, these approaches generally do not take the underlying experimental design into account. Therefore, the different sources of variation are confounded in the PCA model and this can seriously hamper the interpretation of the principal components (Jansen *et al*., 2005). The typical methodology for the analysis of designed experiments is Analysis of Variance (ANOVA), which focuses on the separation of the different sources of variability. Smilde and coworkers (Smilde *et al*., 2005) proposed an adaptation of the SCA (Simultaneous Component Analysis) algorithm that incorporates experimental design information through ANOVA modelling. The so-called ASCA (ANOVA-SCA) basically applies PCA to the estimated parameters in each source of variation of an ANOVA model. This methodology has been successfully applied to data analysis problems in psychology (Timmerman and Kiers, 2003) and metabolomics (Jansen *et al*., 2005).

In this chapter, we study further the applicability of the ASCA approach to the analysis of high-dimensional microarray data from a designed experiment involving several factors. In particular, we use ASCA to explore gene expression trends and differences in MSTC microarray experiments. We show how ASCA is an effective approach for separating the data variability present in a complex MSTC experimental set-up to i) extract the signal of interest from noisy data, ii) reveal major expression patterns associated with the different experimental factors and iii) identify most relevant experimental conditions. We develop further the original methodology by incorporating algorithms for identifying significant signals and selecting genes that behave according to the detected patterns. We use a published MSTC dataset for illustrating the methods presented in this work and synthetic data to provide a deeper understanding of the working of the methodology. The methodology, denoted by ASCA-genes, has been implemented in the statistical language R and it is available at http://www.ua.es/personal/mj.nueda and also at http://bioinfo.cipf.es/aconesa.

5.2 Methods

5.2.1. Model definition

We will consider the general case of a multiseries time-course microarray experiment, where the experimental design is defined by two main types of variables: the time component and the experimental groups for which temporal gene expression differences are sought. Consider I time points ($i=1,\dots,I$), J experimental groups ($j=1,\dots,J$), R_{ij} replications, $r=1,\dots,R_{ij}$ for each case ij, and N genes ($n=1,\dots,N$). For each gene, we will denote by x_{ijr} the gene expression measure at the time i, under condition j and for replicate r.

The analysis of this experiment with the ASCA approach (Smilde *et al.*, 2005), implies the definition for each gene of the ANOVA model given in Equation (5.1)

$$x_{ijr} = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij} + (\alpha\beta\gamma)_{ijr} \qquad (5.1)$$

where μ is an offset term, α_i is the model parameter for factor time on level i, β_j measures the *j-th* group effect, $(\alpha\beta)_{ij}$ represents the interaction effect between the *i-th* time and *j-th* group, and the individual variation is indicated by $(\alpha\beta\gamma)_{ijr}$ instead of ε_{ijr} to avoid confusion with the error term in the subsequently derived ASCA model. The terms in Equation (5.1) can be estimated by least squares under certain constraints as indicated in Table 5.1.

ANOVA FACTOR	CONSTRAINTS	ESTIMATE
μ	-	$x_{...}$
α_i	$\sum_i \alpha_i = 0$	$x_{i..}$-$x_{...}$
β_j	$\sum_j \beta_j = 0$	$x_{.j.}$-$x_{...}$
$(\alpha\beta)_{ij}$	$\sum_i (\alpha\beta)_{ij} = \sum_j (\alpha\beta)_{ij} = 0$	$x_{ij.}$-$x_{i..}$- $x_{.j.}$+$x_{...}$

Table 5.1: ANOVA factor estimates. The value *x...* is the overall average of variable *x*; $x_{i..}$ is the average value of all measurements of variable *x* corresponding to level *i* of the factor α, $x_{.j.}$ is the average value of all measurements of variable *x* corresponding to level *j* of the factor β and $x_{ij.}$ is the average value of all measurements of variable *x* corresponding to the levels *i* and *j* of the factors α and β respectively.

When we consider a microarray experiment with *N* genes and $M = \sum_{i,j} R_{ij}$ samples, a matrix ***X*** of dimensions (*M*×*N*) can be defined containing the entire gene expression dataset. Similarly, the estimates of the ANOVA parameters on the right hand side of Equation (5.1) can be obtained for all genes and collected into matrices, where rows represent samples and columns represent genes. Therefore expression (5.2) can be obtained:

$$\boldsymbol{X} = \mathbf{1}\boldsymbol{m}^{\mathrm{T}} + \boldsymbol{X}_{\mathbf{a}} + \boldsymbol{X}_{\mathbf{b}} + \boldsymbol{X}_{\mathbf{ab}} + \boldsymbol{X}_{\mathbf{abg}} \qquad (5.2)$$

where, **1** is a size *M* column vector of ones, $\boldsymbol{m}^T$ is a size *N* row vector containing estimates of μ for each gene, matrices $\boldsymbol{X}_a$, $\boldsymbol{X}_b$ and $\boldsymbol{X}_{ab}$ contain the estimates of parameters α_i, β_j and $(\alpha\beta)_{ij}$ respectively, and $\boldsymbol{X}_{abg}$ contains the residuals named $(\alpha\beta\gamma)_{ijr}$. The rows of matrices $\boldsymbol{X}_a$ and $\boldsymbol{X}_b$ are highly structured. All rows related to one level *i* of factor α are equal in $\boldsymbol{X}_a$ and analogously all rows of $\boldsymbol{X}_b$ are equal for each level *j* of factor β. Figure 5.1 shows schematically the structure of the matrices ***X***, $\boldsymbol{X}_a$, $\boldsymbol{X}_b$ and $\boldsymbol{X}_{ab}$. For illustrative purposes the non-replicated case is shown. Besides the high dimension of these matrices, they have low rank. These ranks are related to the number of levels of each factor of the experimental design and the number of constrains involved. The rank of matrices $\boldsymbol{X}_a$, $\boldsymbol{X}_b$ and $\boldsymbol{X}_{ab}$ are I-1, J-1 and (I-1)(J-1).

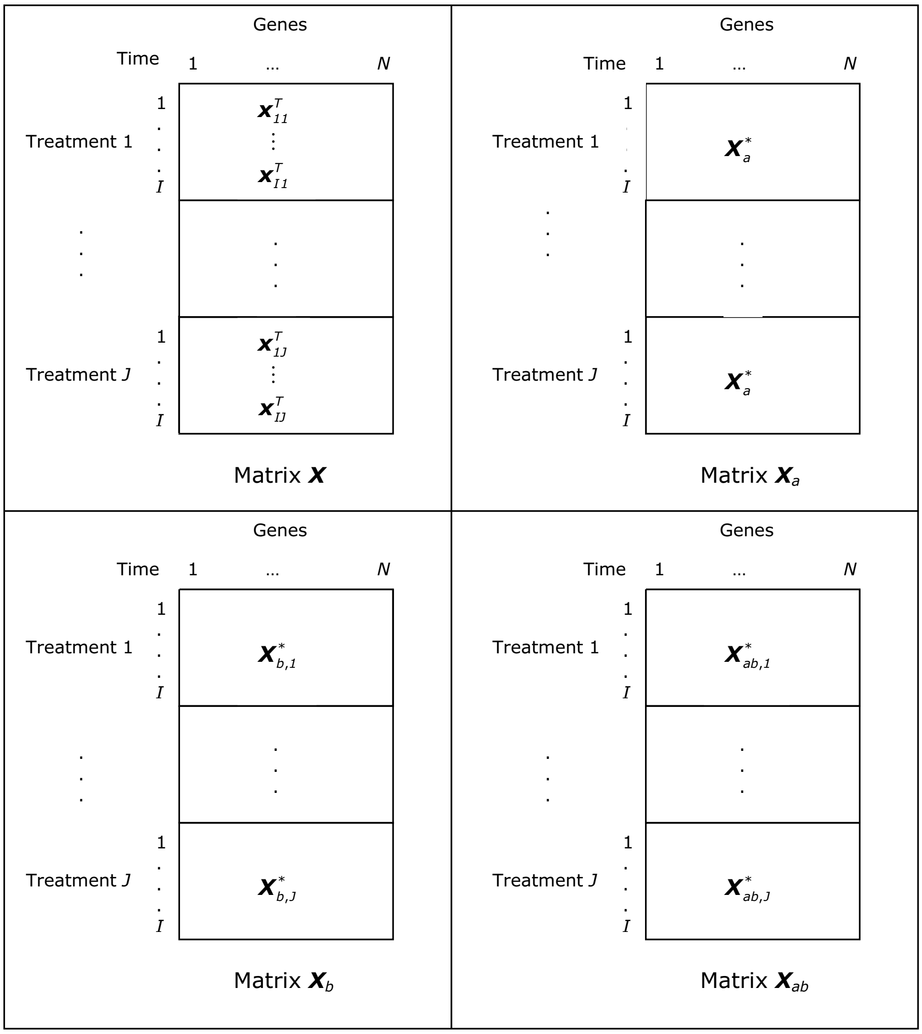


Figure 5.1: Structure of the matrices $\boldsymbol{X}$, $\boldsymbol{X}_a$, $\boldsymbol{X}_b$, $\boldsymbol{X}_{ab}$ for the non-replicated MSTC microarray experiment with I time-points, J experimental groups (treatments) and N genes.

$\boldsymbol{x}_{ij}^T$ is a row vector that contains the expression value for all genes on level i of factor Time and level j of factor Experimental Group.

$\boldsymbol{X}_{\boldsymbol{a}}^{*} = \begin{bmatrix} \boldsymbol{x}_{a,1}^{T} \\ \vdots \\ \boldsymbol{x}_{a,I}^{T} \end{bmatrix}$ where $\boldsymbol{x}_{\boldsymbol{a,i}}$ is a size *N* vector containing estimates of time effect on level *i* for all the genes.

$\boldsymbol{X}_{\boldsymbol{b,j}}^{*} = \begin{bmatrix} \boldsymbol{x}_{b,j}^{T} \\ \vdots \\ \boldsymbol{x}_{b,j}^{T} \end{bmatrix}$ where $\boldsymbol{x}_{\boldsymbol{b,j}}$ is a size *N* vector containing estimates of group effect on level *j* for all the genes.

$\boldsymbol{X}_{\boldsymbol{ab,j}}^{*} = \begin{bmatrix} \boldsymbol{x}_{ab,1j}^{T} \\ \vdots \\ \boldsymbol{x}_{ab,Ij}^{T} \end{bmatrix}$ where $\boldsymbol{x}_{\boldsymbol{ab,ij}}$ is a size *N* vector containing estimates of the interaction effect on level *ij* for all the genes.

When experimental data contains numerous variables in which correlation relationships are present as it would normally be expected in a microarray dataset, the data matrix ***X*** contains redundant information, and therefore also the matrices $\boldsymbol{X}_a$, $\boldsymbol{X}_b$, $\boldsymbol{X}_{ab}$ and $\boldsymbol{X}_{abg}$. In this case, information can be summarized by applying multivariate projection techniques that reduce data dimensionality. These methods project the data into a subspace with the minimal loss of information, and this subspace represents the principal directions of data variability. The analysis of the data in the new subspace allows the identification of the major signals. Given the data decomposition obtained by the ANOVA model, it is possible that each source of variability has different principal directions. It is therefore convenient to apply dimension reduction separately to each matrix $\boldsymbol{X}_a$, $\boldsymbol{X}_b$, $\boldsymbol{X}_{ab}$ and $\boldsymbol{X}_{abg}$. Consequently, the ASCA model corresponding to ANOVA Equation (5.2) is given in Equation (5.3),

$$\boldsymbol{X} = \boldsymbol{1m}^{T} + \boldsymbol{T}_a\boldsymbol{P}_a^{T} + \boldsymbol{T}_b\boldsymbol{P}_b^{T} + \boldsymbol{T}_{ab}\boldsymbol{P}_{ab}^{T} + \boldsymbol{T}_{abg}\boldsymbol{P}_{abg}^{T} + \boldsymbol{E} \qquad (5.3)$$

where the SCA component scores of each submodel are given by the matrices indicated by $\boldsymbol{T}_a$, $\boldsymbol{T}_b$, $\boldsymbol{T}_{ab}$, $\boldsymbol{T}_{abg}$, and the submodel loadings are given by the matrices $\boldsymbol{P}_a$, $\boldsymbol{P}_b$, $\boldsymbol{P}_{ab}$, $\boldsymbol{P}_{abg}$, where $\boldsymbol{P}_x^{T}\boldsymbol{P}_x = \boldsymbol{I}$ for *x=a, b* or *ab*, without loss of generality, (Jansen *et al*., 2005). ***E*** is a matrix in which the residuals of all submodels of the ASCA-model are collected: $\boldsymbol{E}=\boldsymbol{E}_a+\boldsymbol{E}_b+\boldsymbol{E}_{ab}+\boldsymbol{E}_{abg}$. The structure of the matrices $\boldsymbol{X}_a$, $\boldsymbol{X}_b$, $\boldsymbol{X}_{ab}$ and $\boldsymbol{X}_{abg}$ defines the structure of matrices $\boldsymbol{T}_a$, $\boldsymbol{T}_b$, $\boldsymbol{T}_{ab}$, $\boldsymbol{T}_{abg}$. For example, for the case showed in Figure 5.1 the structure of these matrices of scores is as follows:

$$\boldsymbol{X} = \mathbf{1}\boldsymbol{m}^{T} + \begin{bmatrix} \boldsymbol{T}_{\boldsymbol{a}}^{*} \\ \vdots \\ \boldsymbol{T}_{\boldsymbol{a}}^{*} \end{bmatrix} \boldsymbol{P}_{a}^{T} + \begin{bmatrix} \boldsymbol{T}_{\boldsymbol{b},\mathbf{1}}^{*} \\ \vdots \\ \boldsymbol{T}_{\boldsymbol{b},\boldsymbol{J}}^{*} \end{bmatrix} \boldsymbol{P}_{b}^{T} + \begin{bmatrix} \boldsymbol{T}_{\boldsymbol{ab},\mathbf{1}}^{*} \\ \vdots \\ \boldsymbol{T}_{\boldsymbol{ab},\boldsymbol{J}}^{*} \end{bmatrix} \boldsymbol{P}_{ab}^{T} + \begin{bmatrix} \boldsymbol{t}_{\boldsymbol{abg},\mathbf{11}}^{\boldsymbol{T}} \\ \vdots \\ \boldsymbol{t}_{\boldsymbol{abg},\boldsymbol{IJ}}^{\boldsymbol{T}} \end{bmatrix} \boldsymbol{P}_{abg}^{T} + \boldsymbol{E} \qquad (5.4)$$

where $\boldsymbol{T}_{a}^{*}$ is repeated inside $\boldsymbol{T}_{a}$ J times, the I rows of each matrix $\boldsymbol{T}_{b,j}^{*}$ are equal and the rows of each matrix $\boldsymbol{T}_{ab,j}^{*}$ are different, although if there were replicates the rows of the same cases *ij* would be equal.

In the rest of this work, the ASCA submodels in Equation (5.3) will be indicated as "submodel a", "submodel b", "submodel ab" and "submodel abg", respectively. Further details on the ASCA approach can be found in Jansen *et al.* (2005).

Once the ASCA model has been derived and computed, data analysis proceeds, as in regular PCA, with the exploration of the loadings in a selected number of PCs, which are typically obtained on the basis of the percentage of the explained variability or by a cross-validation criterion. In the case of ASCA, the number of components to retain has to be decided for each of the submodels. We propose a PC selection procedure based on the scree-plot of each submodel and the explained variability and interpretation of the associated patterns. Main PCs are identified as those previous to the slope inflexion on a scree-plot representation of the accumulated variability and additional PCs are retained when they describe interesting expression patterns. Finally, the graphical analysis of the score profiles of the selected components in the different submodels (time, experimental groups and interactions, $\boldsymbol{T}_{a}$, $\boldsymbol{T}_{b}$ and $\boldsymbol{T}_{ab}$ respectively) allows to extract conclusions on the effects that the different experimental factors have on gene expression.

5.2.2 Gene-wise analysis

Once major variability patterns have been identified, one step further in the analysis is to identify both those genes that more closely follow the detected trends as well as those that clearly diverge from the general model. The first ones would represent those genes that most coordinately respond in the experimental context; and the second ones would include odd behaviours or outlier data. For this goal we analyze the loadings ($\boldsymbol{P}_{a}$, $\boldsymbol{P}_{b}$, $\boldsymbol{P}_{ab}$) and the residuals ($\boldsymbol{E}_{a}$, $\boldsymbol{E}_{b}$, $\boldsymbol{E}_{ab}$) of the genes in the components selected in each model. Genes with high absolute loading in a specific component are those that follow the behaviour described by this component and genes with high residuals are genes poorly modelled by this component. We propose the use of two statistics, the leverage and the squared prediction error (SPE) to quantify these two aspects.

The leverage is a measure of the importance of a variable (in this case gene) in the PCA model. This is computed according to Equation (5.5) (Martens and Næs, 1989):

$$\boldsymbol{h}_x = diag[\boldsymbol{P}_x \boldsymbol{P}_x^T], \quad x = a,\ b \text{ or } ab \tag{5.5}$$

Where $diag[\boldsymbol{P_x P^T_x}]$ is a vector with the diagonal of matrix $\boldsymbol{P_x P^T_x}$ and $\boldsymbol{h}_x$ the vector containing the leverage values for all the genes in submodel *x* (*x*=a, b or ab). A threshold for statistical significance of leverage can be defined by resampling methods. In our case we have chosen a permutation approach in which a number of row permutations of matrix ***X*** are generated to create a reference distribution where the designed structure of the data has been destroyed. ASCA is then applied to each permuted matrix with the same number of components as taken for the original data and gene leverages are computed in each case. The leverage threshold value at a given confidence (1-α) is obtained as the average of the (1−α)% quantiles computed for all the genes.

The SPE associated with a particular gene is a measure of the goodness of fit of the model for that specific gene. Genes not following the general structure defined in the fitted model will have high SPE. The vector containing the values of this statistic for all the genes can be computed in each submodel according to Equation (5.6):

$$\boldsymbol{SPE}_x = diag[\boldsymbol{E}_x^T \boldsymbol{E}_x] \quad x = a,\ b \text{ or } ab \tag{5.6}$$

The SPE for a particular gene in a submodel is a quadratic form of the errors associated with that gene. Assuming that these errors are well approximated by a multivariate normal distribution, Box (1954) showed that the SPE is well approximated by a weighted chi-squared distribution ($g\chi_h^2$). We have used this approximation to establish the (1- α) confidence SPE threshold. We estimate *g* and *h* by matching moments of the $g\chi_h^2$ distribution: the mean and variance ($\mu = gh, \sigma^2 = 2g^2h$) are equated to the sample mean (m) and variance (v) of the

SPE sample obtaining the next expression as SPE threshold at α level of significance:

$$SPE_{\alpha} = \frac{v}{2m}\chi^{2}_{\frac{2m^{2}}{v},\alpha} \tag{5.7}$$

Therefore, by combining leverage and SPE criteria allows genes can be categorized in relation to modelling and interest. Most relevant genes in the derived ASCA model will be those showing high leverage and low squared prediction error. Poorly modelled genes will be identified by high values in their SPE, while those genes having low leverage and low SPE will be regarded as not affected by the experimental factors (Table 5.2).

	Low SPE	**High SPE**
Low leverage	*Not responsive*	*Badly modelled* *Possibly odd data*
High leverage	*Well modelled* *Follow main trends*	*Influential but poorly modelled.*

Table 5.2: Criteria for gene categorization. Shaded categories provide genes for further analysis.

5.2.3 Comparative analysis

The ASCA-genes approach was compared with four different methodologies described for the analysis of time course microarray data: the clustering methods SOTA and *K*-means, and the hypothesis testing based approaches timecourse and maSigPro. SOTA is a hierarchical unsupervised growing neural network which adopts the topology of a binary tree and offers a statistical criterion for cluster division (Herrero *et al*., 2001). *K*-means is a non-hierarchical partition based algorithm widely used in microarray data analysis (Hartigan and Wong, 1979). *K*-means uses a minimum "within-class sum of squares from the centers" criterion to select the clusters and requires the number of partitions to be fixed in advanced. Timecourse applies a multivariate empirical Bayes statistic (the MB-statistic) to the analysis of replicated time course data. The algorithm contrasts, for each gene, the null hypothesis of constant vector of means along the time component, to the alternative hypothesis of non invariability. The MB-statistic can be used to rank genes in the order of evidence of non constancy (Tai and Speed, 2006). Finally, maSigPro is a model-based univariate method in which different temporal series are modelled by binary variables. The method assesses significant differences in gene expression profiles between time series through the significance of the estimated model parameters (Conesa *et al*., 2006).

SOTA analysis of time course data was done at the GEPAS server (www.gepas.org) taking Euclidean distance as similarity metric and a threshold of 90% node variability as stop growing tree condition. *K*-means, timecourse and maSigPro analyses were performed with the corresponding R packages available at the Bioconductor repository (www.bioconductor.org). The default Hartigan and Wong algorithm and 25 partitions were taken as parameters of the *kmeans* function. A criterion of positive MB-statistics was used for the feature selection in the timecourse package while a significance value of 0.01 was applied in the maSigPro approach. Additionally, Principal Component Analysis computations were done using the princomp function of the stats R package.

5.3 Results

The proposed method has been applied to two datasets. The first one is the toxicogenomics study described in Chapter 2 that involves different treatments and time points. The second one is a synthetic dataset which reproduces the structure present in the real toxicogenomics experiment. In the latter dataset, signals and noise sources have been simulated to resemble real data. The synthetic data was used to analyze how different sources of variability are treated by the ASCA-genes approach while the real dataset was used to study the biological interpretation of the ASCA-genes results.

As we have explained in Chapter 2, the experimental dataset comes from a toxicogenomics study by Heijne *et al*. (2003) where the effect of the hepatotoxicant bromobenzene is studied. In this study there are 3 time points (i=1,2,3), 5 experimental groups (j=1,…,5), 2 or 6 replicates, r=1,…,R_{ij} (2 or 6) for each case ij, and 2665 genes (n=1,…,N).

The simulated data was created to reproduce the experimental set-up of the previous dataset -5 experimental groups and 3 time points- introducing responsive genes and noise in a controlled manner. The synthetic dataset contained 410 genes with profile changes classified into 5 expression patterns: 100 genes with continuous induction for all the groups (A), 100 genes with continuous induction for group 5 (B), 100 genes with continuous repression for group 5 (C), 100 genes with continuous induction for group 4 (D) and only 10 with transitory induction for group 3 (E) (see examples in Figure 5.2). The reason for including a responsive group with few genes, group (E), is the interest in analysing the behaviour of the method on a minor trend. Additionally, there were 2500 flat profile genes without differences between experimental groups, making a total of 2910 genes. The replicates for each gene were produced as independent observations from a normal distribution. It is however, known that in real experiments residuals do not necessarily follow a normal distribution. Non-random sources of technical variability such as spatial bias, not corrected dye-swap labelling, or mixture of data from different labs or experimenters can deviate data from Gaussian distributions. Therefore, to better reproduce real microarray data and in order to analyze how ASCA-genes behaves with this type of variability, systematic noise was introduced to the dataset by splitting the dataset in two replicates and adding two opposite normal distributions to each half.

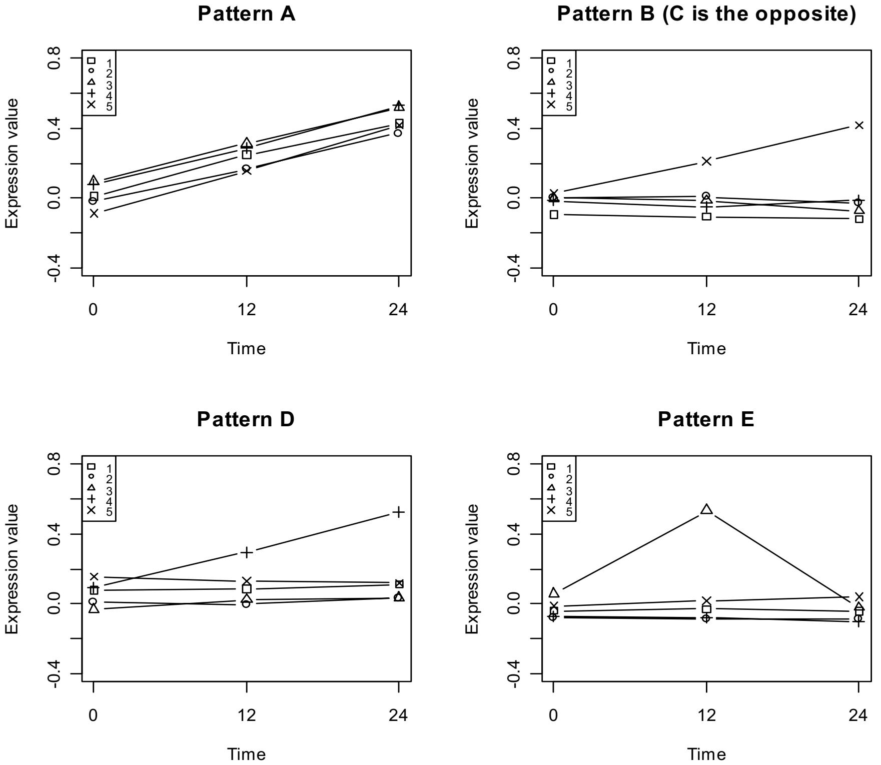


Figure 5.2: Examples from each expression pattern simulated. Lines join the averages for each group and time-point.

As the purpose of this study is to identify gene expression profile differences between experimental groups, when applying ASCA-genes the choice is made to join for each gene β_j and $(\alpha\beta)_{ij}$ effects in Equation (5.1) and analyze them in one submodel as it is shown in Equation (5.8) (Jansen *et al.*, 2005).

$$x_{ijr} = \mu + \alpha_i + \left[\beta_j + (\alpha\beta)_{ij}\right] + (\alpha\beta\gamma)_{ijr} = \mu + \alpha_i + (\beta + \alpha\beta)_{ij} + (\alpha\beta\gamma)_{ijr} \qquad (5.8)$$

Out of these effects, $(\beta + \alpha\beta)_{ij}$ is the most important for biological interpretations: it represents the effect of the treatment group on the gene expression measured as deviations from the common time effect α_i for each gene. The terms in Equation (5.8) can be estimated using Table 5.1. And by arranging all the estimates of the ANOVA parameters for all the genes into matrices, Equation (5.9) is obtained.

$$\boldsymbol{X} = \mathbf{1}\boldsymbol{m}^{\boldsymbol{T}} + \boldsymbol{X}_{\boldsymbol{a}} + \boldsymbol{X}_{\boldsymbol{b}} + \boldsymbol{X}_{\boldsymbol{ab}} + \boldsymbol{X}_{\boldsymbol{abg}} = \mathbf{1}\boldsymbol{m}^{\boldsymbol{T}} + \boldsymbol{X}_{\boldsymbol{a}} + \boldsymbol{X}_{\boldsymbol{b+ab}} + \boldsymbol{X}_{\boldsymbol{abg}} \qquad (5.9)$$

where "submodel a" describes the variation due to the factor time, "submodel b+ab" describes all the variation related to the factor treatment group, and "submodel abg" describes the residual (random and non-random) variation in the data. Therefore, the ASCA-model applied to the described datasets is given by Equation (5.10):

$$\boldsymbol{X} = \mathbf{1}\boldsymbol{m}^{\boldsymbol{T}} + \boldsymbol{T}_{\boldsymbol{a}}\boldsymbol{P}_{\boldsymbol{a}}^{\boldsymbol{T}} + \boldsymbol{T}_{\boldsymbol{b+ab}}\boldsymbol{P}_{\boldsymbol{b+ab}}^{\boldsymbol{T}} + \boldsymbol{T}_{\boldsymbol{abg}}\boldsymbol{P}_{\boldsymbol{abg}}^{\boldsymbol{T}} + \boldsymbol{E} \qquad (5.10)$$

5.3.1 Case 1: Simulated data.

By fitting the ASCA model to the synthetic dataset and using the PC selection criterion described, one and two components were selected for "submodel a" and "submodel b+ab" respectively.

The score profiles of the components of the submodels reveal the most common expression patterns in the genes. The first component of the "submodel a" (Figure 5.3) shows a positive linear effect through time for all experimental groups. This pattern is the case to the simulated pattern A, where no time-experimental group interactions were modelled, only a time effect in all the groups. The first component of the "submodel b+ab" (Figure 5.4) shows different behaviour for group 5 and group 4 with respect to the other groups: a clear positive linear effect through time for group 5 (related to patterns B and C) and a light negative one for group 4 (related to pattern D). The second component of the same submodel (Figure 5.4) shows positive differential behaviour of group 4 and group 5 (related to patterns D, B and C) and also signals different behaviour of group 3 (related to pattern E), even this is less pronounced.

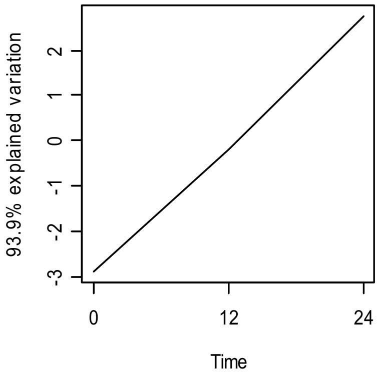


Figure 5.3: Score profiles of component 1 in "submodel a".

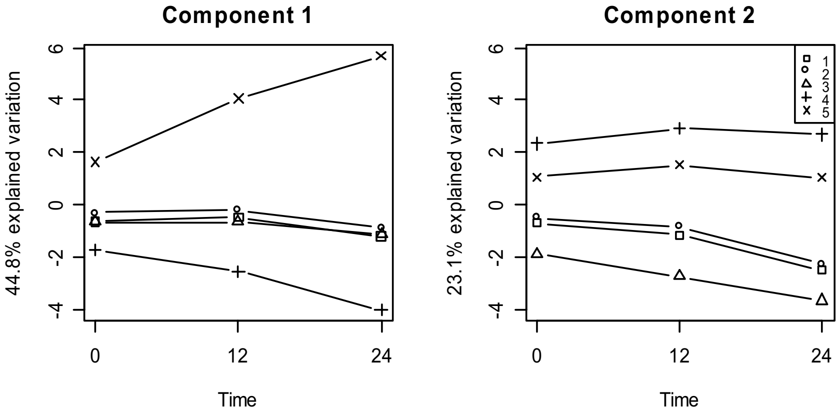


Figure 5.4: Score profiles of component 1 and component 2 in "submodel b+ab".

By analysing the loadings of the genes, we can detect genes that follow the trends shown in the score profiles of the components of the ASCA-model. For example, if we focus on "submodel b+ab" (Figure 5.5), genes with a high positive loadings value for the first component of this submodel correspond to genes which were simulated in pattern B, while genes with a high negative value in the same component were genes simulated as pattern C. As pattern D is detected negatively by the first component and positively for the second one, genes from pattern D have negative loadings for the first component and positive loadings for the second. Another interesting result is that genes without interactions between time and groups (simulated pattern A and modelled by "submodel a") and flat profile genes have low loadings in this model.

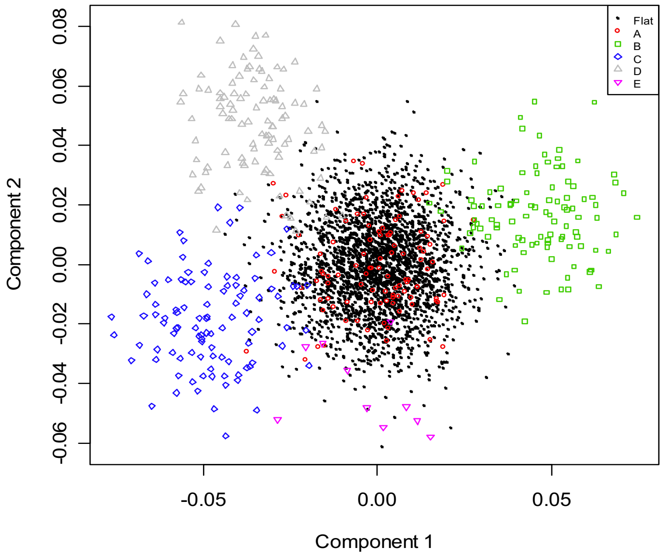


Figure 5.5: Gene loadings in the two components of "submodel b+ab".

The analysis of the leverages and SPE in both models shows interesting results (Figure 5.6). First of all, "submodel a" shows high leverages and good modelling for case A, which is the case where the time effect is the same for all the groups. Secondly, "submodel b+ab" in general shows high leverages and good modelling for cases B, C and D, low leverages for case A and Flat profiles, and high leverages and high SPE for case E. This indicates that case E is bad represented by the model. The reason for this is presumably due to the fact that the variability associated with case E represents a small percentage of the gene set, and hence this source of variation is not included in the model. This example illustrates that when there is a small group of genes whose behaviour is not described in the model, they can be detected by residuals analysis or in extra components. The thresholds shown in Figure 5.6 have been computed with 100 permutations and α=0.01 using the gene-wise analysis described previously. Taking into account that genes of interest have either high leverages or high SPE the method points to 426 genes. This selection means a sensitivity (defined as the proportion of detections above true positives) of 91% and a specificity (defined as the ratio between false calls and true negatives) of 98% (Table 5.3).

		Real positive	**Real negative**

Sensitivity = $\frac{TP}{TP+FN}$
Specificity = $\frac{TN}{TN+FP}$
	Test positive	*TP*	*FP*

	Test negative	*FN*	*TN*

Table 5.3: Sensitivity and specificity definitions.

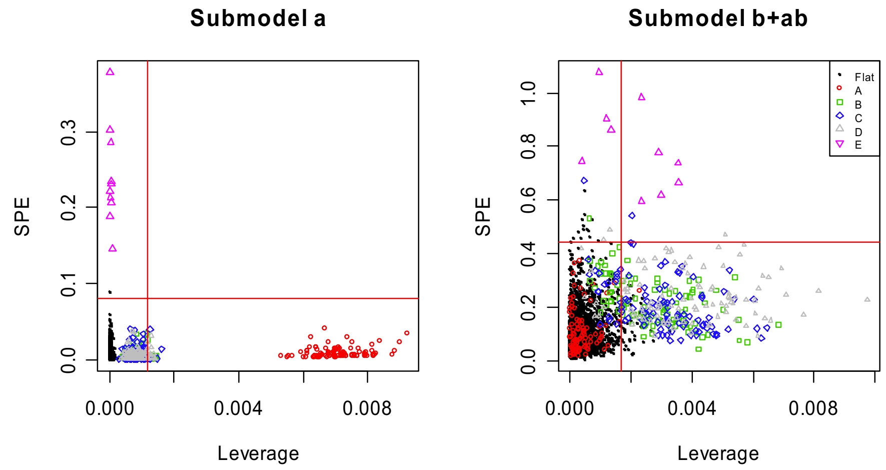


Figure 5.6: Leverages and SPE of the genes for the simulated dataset in "submodel a" and "b+ab".

To illustrate how ASCA-genes analysis is able to filter non-random or unwanted sources of noise and extracts the variability associated with the experimental factors we decided to compare the method with a general projection technique: Principal Component Analysis. We applied PCA to this synthetic dataset and analyzed the variability and meaning of the components similarly to the procedure followed with ASCA-genes. Two components were extracted according to the scree-plot criterion. We observe that the first PCA component explains 82.1% of the total variation. However, as it can be seen on the first plot of Figure 5.7 this component can not be associated with the behaviour of any of the simulated patterns (A, B, C, D or E). When sample scores were analyzed for this first component, we observed that this PC represents the technical noise introduced to the data. The second component (3.94% explained variation) identifies the pattern B and the opposite C. Inspection of a possibly interesting third component (2.6% explained variation) revealed a general linear tendency in all the groups (patterns A) and the pattern simulated in D (not shown). Although these results are interesting, they pose a difficulty to interpretation as the by far main direction of variability turns out to be associated with a technical feature rather to the experimental factors which are the goal of the study. When the identification of these sources of non-random noise is not possible or at least easy – as could be the case in real data-, we would probably fail to give a biological interpretation to the results obtained by this approach. In short, because PCA does not impose the experimental design when doing variability analysis, targeted and non targeted effects can show up and confound.

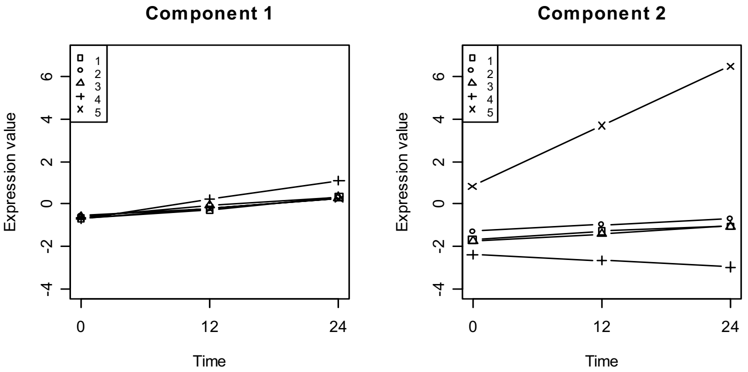


Figure 5.7: Score profiles of the two components of PCA model in the simulated dataset. Lines join the averages for each group and time-point.

		PCA (2 PC's)		ASCA (1& 2 PC's)	
Model	**Initial**	**Explained**	**%**	**Explained**	**%**
a (time)	272	32	11.76	255.3	**93.86**
b+ab (group+ interaction)	1053	461	43.78	713.9	67.80
abg (residual)	10815	9954	**92.04**	0	0
TOTAL	12140	10447	86.05	969.2	7.98

Table 5.4: Variation of the submodels explained by PCA and ASCA-genes. Values collecting most variability in each approach are given in bold.

This problem is directly addressed by ASCA. By taking into account the experimental design, the methodology focuses on the signals of interest avoiding the interference of non-random noise effects. Table 5.4 shows the amount of variation associated with each experimental factor and the percentage explained in PCA and ASCA in this example. In PCA, the variation has been computed using ANOVA on the fitted principal components. This is the so-called PC-ANOVA (Jansen *et al.*, 2005). We can observe that while PCA mainly recovers the variability structure present in the residuals ("submodel abg"), the ASCA-genes procedure focuses in the variation of "submodels a" and "b+ab" directly.

5.3.2 Case 2: Toxicogenomics dataset.

Once the technical aspects of the ASCA-genes procedure has been revealed on a synthetic dataset, we study the methodology on a real experiment to analyze the biological meaning of the results generated by this approach. We have used for this the toxicogenomics study described earlier.

Firstly, data exploration by PCA showed that the first component of variability was associated with the dye labelling of the samples, and only on the second PC revealed a distinct behaviour for the high bromobenzene dose Figure 5.8). This result was already detected in Chapter 2 and illustrates the structural noise problem described in the previous section.

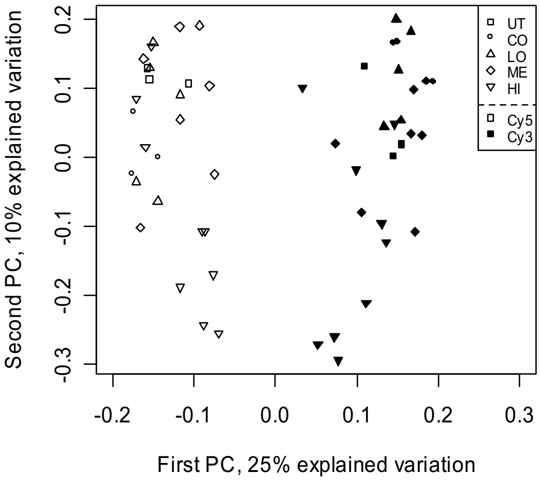


Figure 5.8: PC scores of bromobenzene data.

For ASCA analysis of these data, Equation (5.9) was applied to model the gene expression response. The component selection procedure gave as a result one and three components for "submodel a" and "submodel b+ab", respectively. The score profile of the first component of the "submodel a" (submodel for the time factor, explaining 75.15% of variability) shows that in general, gene expression is mostly affected after 24 hours of treatment, followed by a slight reversion at 48 hours (Figure 5.9). This result reveals the time frame in which treatments have the strongest effect on gene expression and might indicate either a recovery of the biological systems or a loss of drug action at 48 hours. The score profiles of the three components of the "submodel b+ab" (treatment plus interaction submodel) describe the different responses in time (gene expression patterns) for the treatment groups (Figure 5.10). These score profiles show a different effect of the bromobenzene doses on gene expression through time. The score profile of the first component identifies a marked effect of the high bromobenzene dose in gene expression which is different to the rest of treatments. As this component represent almost 50% of the variability associated with this model, it can be interpreted that high bromobenze is, by far, the treatment that most affects gene expression. The next two components have a lower weight (they explain around 10% of the data variability each) and thus represent behaviours of lower impact. Second component identifies a gene expression pattern characterized by a difference in the gene expression response between the HI and ME treatments at 24 hours, while the third component detects basically an effect of differential expression at 6 hours for the HI and ME doses. In general, the ASCA-genes analysis indicates that the major gene expression response of this study is focused on the HI doses at 24 hours and there is a lower response to medium bromobenzene doses. Interestingly similar conclusions about the patterns of bromobenzene toxicity were obtained by Jansen *et al.*, 2005 on the analysis of metabolic changes present in the rats used for this study.

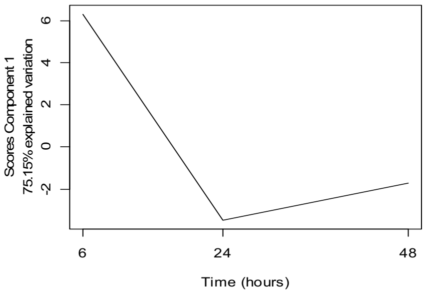


Figure 5.9: Score profiles of component 1 of "submodel a".

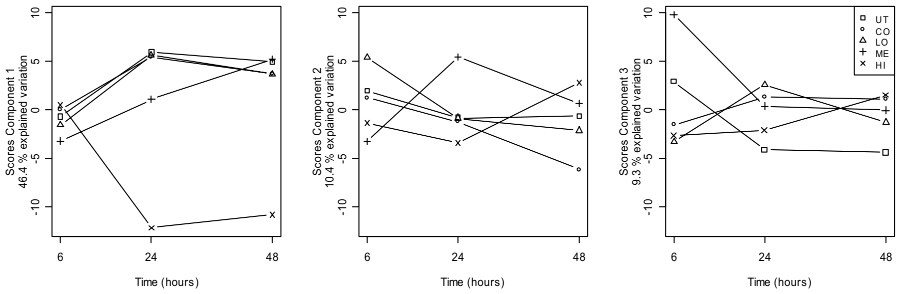


Figure 5.10: Score profiles of the first three components of "submodel b+ab".

The identification of genes following the patterns given by the score profiles of the components is done by loading analysis. Genes with high absolute loading values for the first component of the "submodel b+ab" are genes for which the bromobenzene produces an effect in the high doses very different to the rest of the doses: in case of positive loading, in the high doses gene expression decreases (repression) at 24 hours and it is maintained at 48 hours while the remaining groups do not vary very much over time; on the other hand, genes with negative loadings for this component have the opposite described effect (induction). Finally, genes with loadings on this component near 0 have a pattern different to that described by the first component. Similar reasoning can be applied to the second and third components.

Genes well modelled by the ASCA-model will have high leverages; whereas poorly modelled genes will show high SPE values. Figure 5.11 shows SPE and leverage values computed for all the genes in "submodels a" and "b+ab". In general, genes with high leverage exhibit low SPE value, which means that significant model contributions are also well modelled genes. Cut off values for SPE and leverage were computed as described in the methods section taking α =0.05. In total, 345 genes were found with SPE and leverage values beyond their respective thresholds, of which 157 in "submodel a", 247 in "submodel b+ab" and 59 in both submodels. Additionally 28 genes were identified as high SPE genes.

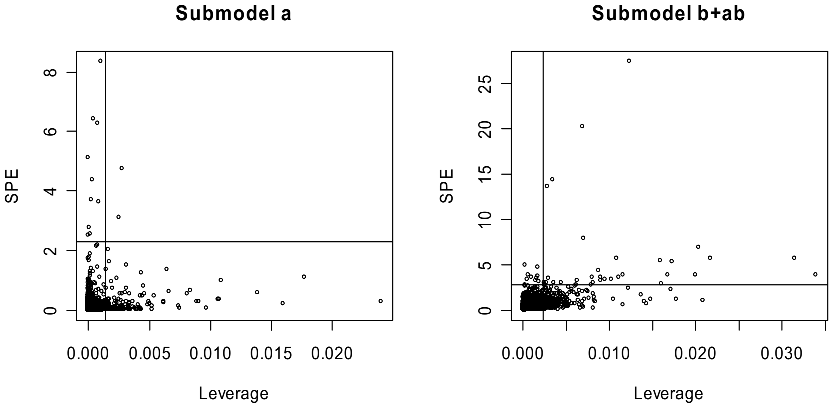


Figure 5.11: Squared prediction error (SPE) and leverage statistics of the genes in "submodel a" (left) and "submodel b+ab" (right). SPE and leverage cut-off values are indicated by horizontal and vertical lines, respectively.

To investigate the biological meaning of the gene selection provided by this approach, Gene Ontology (GO) annotations were fetched for the collection of genes present in the rat chip and functional class enrichment analysis was executed using the software Blast2GO (Conesa *et al.*, 2005) taking a false discovery rate control level of 0.05. The ASCA-genes selected gene pool was significantly enriched for functional categories such as glutathione transferase activity, oxidorreductase activity, microsome, heme binding, fatty acid metabolism, steroid biosynthesis, xenobiotic metabolism, ferric ion binding, response to stimulus, secondary metabolism, fibrinogen complex and structural component of ribosome. These categories are in agreement with a detoxification response described by Heijne *et al*. (2003) that includes upregulation of redox enzymes, such as microsomal glutathione-S transferase or heme oxygenase, which act in the degradation of xenobiotic compounds, the induction of ribosomal constituents and the regulation of acute-phase related proteins such as ferritin and fibrinogen. Steroid biosynthesis and fatty acid metabolism are pathways reported to be induced by corn oil administration trial animals (Takashima *et al.*, 2006). This result indicates a meaningful biological content in the genes associated with the main transcriptional responses detected by the proposed procedure. Additionally, high SPE genes included epoxide hydrolase and alfatoxin B1 inhibitor, activities also reported to be triggered by bromobenzene treatment in rats, already at medium bromobenzene doses, differently from most differentially expressed genes which responded only to the highest dose.

To understand the added value of ASCA-genes above other MSTC analysis methods we compared our approach with four different methodologies described for the analysis of time course microarray data: two clustering methods, SOTA and *K*-means, widely used in many gene expression profiling studies, and two hypothesis testing based approaches, timecourse and maSigPro, especially conceived for time series data. For this, the toxicogenomics dataset was analyzed with each of the alternative methods and results were compared to ASCA-genes in terms of provided overall information and biological meaning of feature selection.

The first noticeable difference when comparing ASCA-genes to the methods described above, is the absence in the latter of an explicit view of major gene expression features related to the experimental factors. While ASCA-genes gives at first glance a representation of the main gene expression changes occurring along time and across series (Figure 5.9 and Figure 5.10), the extraction of this information in the other methods is at least not immediate. The SOTA analysis provided a tree of 26 end nodes in which the gene expression profiles on the different time series and their relative importance were difficult to reveal (Annex 2). Similar conclusions were drawn from the *K*-means partitioning (Annex 2). On the other hand, timecourse and maSigPro packages produced lists of genes with an associated value of statistical significance for differential expression and again without a summarizing representation of experimental factor effects related to major gene expression trends. Next, for comparing the feature selection aspect of ASCA-genes, a selection of genes has to be made for the alternative approaches. In the case of the clustering methods this implied the selection of "important" clusters and here we encountered the difficulty of not having an objective tool for this selection. We therefore made a selection on the basis of visual inspection of the cluster profiles, selecting according to a series associated pattern of differential expression and by the magnitude of the change. In the case of SOTA, this resulted in the selection of the upper 11 and lower 9 clusters, summing up a total of 155 genes. Upper clusters corresponded basically to genes induced by high and medium doses of the drug, while the lower 9 clusters contained repressed genes (Annex 2). In the case of *K*-means this resulted in the selection of 5 clusters (136 genes) which also included up and down regulated genes at high and medium bromobenzene doses (Annex 2). In the case of maSigPro, a selection of 155 genes was obtained when applying a significance level of 0.01. Timecourse, on the contrary, does not provide a selection of significant features but a rank of genes ordered by the value of the MB-statistic. Taking genes with a positive value for the

MB-statistic, a total of 256 genes were selected for comparisons. It should be mentioned that the results of the timecourse approach changed with the scale of the data: when data values were multiplied by 100, the gene associated MB values greatly increased, becoming positive in many cases (data not shown), which drastically affects the results of any absolute feature selection criterion. This did not happen with any of the other statistical approaches.

Table 5.5 shows the results of comparing the ASCA-genes feature selection with the alternative approaches. A complete list of selected genes by the different approaches is given in a supplementary file of the published paper available at http://bioinformatics.oxfordjournals.org/cgi/content/full/btm251/DC1 and also at http://www.ua.es/personal/mj.nueda. In all cases, ASCA-genes provided a greater gene selection. Moreover, 70-85% of the genes selected with the alternative method were present in the ASCA-gene pool, except for the timecourse method where overlap was only of 50%. Analysis of the biological meaning content in the groups of genes that were different in each pairwise comparison indicated the presence of significantly enriched GO terms in any of the groups of genes selected by ASCA-genes and not detected by the other approaches. In contrast, for the opposite comparison no enriched GO terms could be detected, except for the *K*-means vs. ASCA-genes comparison which revealed the presence of ribosomal proteins not selected by the ASCA-genes approach.

Taken together, these results suggest a stronger discovery power of the multivariate approach and illustrate another distinct feature of the ASCA-genes, i.e. categorization according to values of leverage and SPE statistics. High leverage and low SPE genes are genes that vary according to the main trend and correspond to major molecular functions affected by the treatment. High SPE genes are model diverging data and would correspond to responsive genes with a minority pattern. Low leverage genes show low variance and encode functions less specific in the bulk response. Although the commonality or not of a given gene expression pattern can also be derived through cluster analysis (i.e. high SPE genes can be found in the two most upper clusters of the SOTA result), the explicitness and magnitude of this kind of information is best obtained in the ASCA approach.

ASCA-genes (373)	**SOTA (155)**	***K*-means (136)**	**Timecourse (256)**	**maSigPro (155)**
Common with ASCA-genes	106	106	119	132
ASCA-genes vs. alternative method	biosynthetic process translation fibrinogen complex RNA binding	Lipid-metabolic process monooxygenase activity oxidoreductase activity retinoid metabolic process steroid dehydrogenase activity response to xenobiotic stimulus glutathione transferase activity	microsome fibrinogen complex ribonucleoprotein complex cytoplasm biosynthetic process translation lipid metabolic process	oxidoreductase activity steroid biosynthetic process fibrinogen complex
Alternative method vs. ASCA-genes	none	Ribosome	none	none

Table 5.5: Gene selection comparison between ASCA-genes and alternative methods. GO term enrichment analysis of differential set was done with the software Blast2GO (Conesa *et al*., 2005). Numbers in brackets indicate the selection of genes in each method.

5.4 Discussion

This chapter addresses the development and application of the ASCA-genes methodology for the analysis of MSTC microarray data. ASCA combines ANOVA and SCA techniques. Basically, ASCA applies PCA to the estimated parameters in each source of variation of an ANOVA model. The method estimates two gene statistics, leverage and SPE that provide information on the adequateness of the model for each gene. This methodology analyses data from a multivariate prospective, taking into account the experimental design and focusing on the sources of variability associated with each of the experimental factors.

The application of the ASCA approach to transcriptomics data has two practical uses. On the one hand, the analysis of the score profiles of the components of the submodels of interest (time, experimental groups and interactions) helps to understand the shared behaviours of gene expression under the studied conditions. On the other hand, the study of the loadings allows the identification of the genes that follow the discovered patterns. Finally, by analysing the residuals small group of genes following different expression profiles as those modelled by the ASCA-model can be identified.

One follow up question is how to identify most interesting genes in such approach. We have proposed a criterion for feature selection based on the combined use of two statistics: the leverage, as a measure of the importance of a gene contribution to the multivariate fitted ASCA-model, and the squared prediction error (SPE) as an evaluation of the goodness of fit of the model to a particular gene. Threshold values for these parameters can be derived by resampling methods or using a weighted chi-squared distribution proposed by Box (1954). The simulated data shows that we can use these measures to categorize genes as (1) genes with a variability pattern associated with the main effects of the experimental factors, (2) genes that diverge from the most abundant behaviour but can display a minor gene expression response, and (3) genes which basically do not respond to the factors evaluated in the study. Taking as "interesting" genes those falling into the first two classes, our simulation example also showed that these selection criteria provide specificity and sensitivity values above 90%. Furthermore, the results obtained with experimental microarray data, show that the score profiles derived by ASCA-genes are consistent in the context of a dose-related toxicological response and that the feature selection provided by the leverage and SPE parameters is biologically significant. Comparison of the ASCA approach to other existing methodologies for the analysis of time course data shows the potentials and uniqueness of the proposed method for extracting major and secondary patterns of gene expression associated with the factors of study and for highlighting meaningful gene pools.

Another important benefit of the ASCA-genes approach is the possibility of isolating non-random or unwanted sources of noise. These sources of noise can originate from different labs, experimenters or labelling conditions which associate to only a subset of the data. These non-modelled sources of variability can pollute data and hamper the identification of the signals of interest. We show that when PCA is applied to the gene expression signals in such cases, principal components can pop up that do not associate to the factors of the study and can obscure the interpretation of the results. In fact, complains on the poor interpretability of PCA principal components in microarray analysis has been reported (Raychaudhuri *et al*., 2000). As ASCA-genes focuses only on the variability associated with these experimental factors, it clearly outperforms PCA in terms of interpretability and feature selection.

The ASCA methodology relies on standard ANOVA for extracting the variability in the data associated with the experimental factors. ANOVA, however, might not be the most adequate strategy for the analysis of time series as it does not takes into account possible autocorrelation of serial data. In this work we have considered datasets where this problem was not present, as biological and simulated samples were truly independent at each time point. Application of ASCA-genes to longitudinal data, therefore, should be cautious. In any case, the strategy ASCA (PCA restricted to the variability associated with experimental factors) might be extended to other models such as repeated-measures ANOVA or linear mixed models, which would deserve further research.

Posterior studies revealed that the criterion described in this chapter to select genes is not adequate for experiments with high number of genes and high structural noise. In Annex 3 we include one of these studies. We consider that finding an independent criterion of the signal-noise ratio level is an interesting topic for future research.

Chapter 6

Using ASCA as data pre-processing technique in TCM

6.1 Introduction

As we have already stated in previous chapters, microarrays are a noisy technology where both random and systematic sources of variation can obscure the biological signal. Most of the normalization methods are devoted to centre and scale the data assuming general invariability for all observations and greatly ignoring the particular sample hybridized in each array. It is arguable that when systematic noise is associate with an array or sampling effects, the consideration of the experiment design could provide some means to correct this noise. On the other hand, the co-regulation mechanism that underlies gene expression implies that transcriptomics data has an inherent structure of correlation. Taking this covariance structure into account is an effective way, as we have seen in Chapter 5, to enhance data analysis. In this chapter we propose a strategy to transform the original data by considering the relationships within gene expression and the information of the experimental design. We use the ASCA (ANOVA-SCA, Analysis of Variance - Simultaneous Component Analysis, Smilde *et al*., 2005) approach described in Chapter 5.

ASCA is mainly used as an exploratory tool for the analysis of multivariate data with an underlying experimental design. The approach is effective in identifying variability patterns associated with experimental factors and for the descriptive analysis of complex multivariate dataset which follow a designed set-up. ASCA has been applied in metabolomics (Jansen *et al.,* 2005) and also in transcriptomics (ASCA-genes, Chapter 5) where the methodology was expanded to include a gene selection feature. ASCA-genes was developed for a model with two factors and their interaction, although it could easily extend to more complex models. Although these could be any factors, we associated them with time and treatment to specifically consider the Time Course Microarray (TCM) experiments.

The data decomposition offered by the ASCA and ASCA-genes approach is an interesting means of isolating targeted effects (time and treatment effects) from noise effects –through the ANOVA decomposition- and identifying the gene expression features associated with these effects –through the PCA on the ANOVA factors–. In this study we propose to exploit this pre-processing data to generate a data filter strategy that extracts the information of interest by removing both the noise present in the gene expression signal and the signal (or structure) present in the noise of the data. The processed data can then be used for statistical analysis with any dedicated methodology for TCM. We have analyzed the effect of the ASCA-based filter in the performance of maSigPro (described in Chapter 4) and also in other recently developed methodologies, i.e. timecourse and EDGE. The main reasons for choosing these last two methodologies were, firstly, their availability and, secondly, their ability of dealing with multiple time series.

Simulations studies have been carried out to investigate the performance of the ASCA filter in different situations. In a first study, datasets were simulated with a structure of independent observations and short time-course experimental design. Different amounts and types of noise were added resulting in distinct scenarios. In the second case, datasets with longitudinal data structures were simulated introducing likewise different types of noise. maSigPro, timecourse and EDGE packages were applied to simulated data before and after the ASCA-filtering and rates of false positive and negative discoveries were recorded. maSigPro is indicated for independent observations, while timecourse deals especially with longitudinal data. EDGE claims to be suited for both longitudinal and independent data. Our results indicate that the ASCA filter is effective in improving the detection power of TCM analysis methods, especially when high levels of structural noise are present. Interestingly, although maSigPro is not a longitudinal data analysis method, it does provide an efficient selection of d.e.g. in combination with the ASCA filter.

6.2 Material and methods

6.2.1 The proposed filter

We consider the ASCA model described in Chapter 5, where PCA is applied to the matrices collecting the ANOVA estimated effects of time factor ($\boldsymbol{X}_a$), treatment ($\boldsymbol{X}_b$), interaction ($\boldsymbol{X}_{ab}$) and error ($\boldsymbol{X}_{abg}$) for all genes. As Equation (6.1) indicates, the model can be divided in two parts: one corresponding to the gene expression feature the experiment is aimed at and the other corresponding to the experiment noise which is captured in the model residuals.

$$\boldsymbol{X} = \mathbf{1}\boldsymbol{m}^T + \overbrace{\overbrace{\boldsymbol{T}_a\boldsymbol{P}_a^T + \boldsymbol{E}_a}^{\boldsymbol{X}_a} + \overbrace{\boldsymbol{T}_b\boldsymbol{P}_b^T + \boldsymbol{E}_b}^{\boldsymbol{X}_b} + \overbrace{\boldsymbol{T}_{ab}\boldsymbol{P}_{ab}^T + \boldsymbol{E}_{ab}}^{\boldsymbol{X}_{ab}}}^{\text{PART I: Signal of interest}} + \overbrace{\overbrace{\boldsymbol{T}_{abg}\boldsymbol{P}_{abg}^T + \boldsymbol{E}_{abg}}^{\boldsymbol{X}_{abg}}}^{\text{PART II: Residuals}} \tag{6.1}$$

As discussed in Chapter 5, the ASCA approach provides the means of identifying relevant behaviours within each source of variability considering the correlation structure of the data. These behaviours are represented in the scores of the main components of each submodel (***T*** matrices). Furthermore, gene loadings (***P*** matrices) are the representations of individual expressions in the extracted components and can be used to select genes that more closely follow the discovered patterns.

However, as ASCA is based on the ANOVA model, it does not directly exploit the quantitative nature of the time factor and it could be argued that other methods that do consider this aspect could be more suited for identifying differentially expressed genes in TCM data analysis. In order to benefit from the decomposition properties of ASCA while keeping a significant analysis under a quantitative-based statistics, we propose to use the former method as a pre-processing or filtering strategy to the later methodology. This previous step would help to reduce noise and focus on the main structures present in the data, as the filtered data will be the projections on the principal directions of variation. Hence, final results would be improved in two ways, firstly we focus on genes with shared behaviours and secondly we remove noise. A filtered data matrix can be obtained by removing from the original data matrix, the residual matrices of time, group and interaction submodels obtained by data decomposition, such as given by Equation (6.2).

$$\tilde{\boldsymbol{X}} = \boldsymbol{X} - \boldsymbol{E_a} - \boldsymbol{E_b} - \boldsymbol{E_{ab}} \tag{6.2}$$

By removing the residual matrices $\boldsymbol{E}_{a}$, $\boldsymbol{E}_{b}$ and $\boldsymbol{E}_{ab}$ from matrix $\boldsymbol{X}$, the noise embedded in the signal, i.e. the noise contained in each of the experimental factors (time, group and their interaction), will be filtered out. Note that in this model, we are still leaving all the noise not captured by the ASCA model in the data, i.e matrix $\boldsymbol{X}_{abg}$, which would provide the noise variation required to carry out an effective inferential analysis. However, in experimental datasets, noise is frequently a mixture of random and systematic or structural noise components. Sources of systematic noise are, for example, dye effects in two colour technologies, or a batch (lab, time schedule, technician, etc.) effect affecting subsets of arrays differently. Formally, this systematic noise can be modelled as the latent structures present in the component $\boldsymbol{X}_{abg}$ of the ASCA model and can be identified by applying PCA to this matrix. Therefore our filtering strategy can be improved by subtracting $\boldsymbol{T}_{abg}\boldsymbol{P}_{abg}^{T}$ in Equation (6.2), resulting in a data processing according to Equation (6.3). In this formulation we generate a filtered dataset $\tilde{\tilde{\boldsymbol{X}}}$ where putatively both the noise of the signal (***E*** matrices in ASCA submodels) and the signal of the noise (***TP*** element of the ANOVA error) are removed.

$$\tilde{\tilde{\boldsymbol{X}}} = \boldsymbol{X} - \boldsymbol{E_a} - \boldsymbol{E_b} - \boldsymbol{E_{ab}} - \boldsymbol{T}_{abg}\boldsymbol{P}_{abg}^{T} \tag{6.3}$$

We therefore suggest using $\tilde{\tilde{\boldsymbol{X}}}$ instead of ***X*** as filtered data to be subjected to further analysis by any statistical methodology for time course data, as explained in the next section.

Following the same rationale discussed in Chapter 5, we concentrate analysis of TCM data in the differences between experimental groups ($\boldsymbol{X_b}$) over time and therefore we choose to join for each gene this factor with the interaction ($\boldsymbol{X_{ab}}$). Arranging parameter estimates of the ANOVA models for all genes into matrices, equation (6.4) is obtained.

$$\boldsymbol{X} = \boldsymbol{1m^T} + \boldsymbol{X_a} + \boldsymbol{X_b} + \boldsymbol{X_{ab}} + \boldsymbol{X_{abg}} = \boldsymbol{1m^T} + \boldsymbol{X_a} + \boldsymbol{X_{b+ab}} + \boldsymbol{X_{abg}} \tag{6.4}$$

Therefore, the applied ASCA model is al follows:

$$\boldsymbol{X} = \boldsymbol{1m}^T + \overbrace{\boldsymbol{T}_a\boldsymbol{P}_a^T + \boldsymbol{E}_a}^{\boldsymbol{X_a}} + \overbrace{\boldsymbol{T}_{b+ab}\boldsymbol{P}_{b+ab}^T + \boldsymbol{E}_{b+ab}}^{\boldsymbol{X_{b+ab}}} + \overbrace{\boldsymbol{T}_{abg}\boldsymbol{P}_{abg}^T + \boldsymbol{E}_{abg}}^{\boldsymbol{X_{abg}}} \tag{6.5}$$

The number of components criterion

The ASCA approach requires that a number of PC will be selected to generate the different submodels. The number of components taken affects the amount of signal and noise retained and therefore the goodness of fit of the solution depends on the correctness of this selection.

There are several methods to choose from: analysis of the scree-plots, cross-validation, choosing a predefined threshold of variability, etc. As the goal is to select the variation of interest, described by models $\boldsymbol{X_a}$ and $\boldsymbol{X_{b+ab}}$, and to remove the possible structural noise, included in model $\boldsymbol{X_{abg}}$, we decided to use different criteria for each part. For retaining signal, we choose a number of components that explain a high quantity of variation of $\boldsymbol{X_a}$ and $\boldsymbol{X_{b+ab}}$ submodels, fixed in more than the 75% of the variation in each case. For removing structural noise the situation is different. If $\boldsymbol{X_{abg}}$ has structural noise, PCA will capture it and there will be some eigenvalues of the covariance matrix of $\boldsymbol{X_{abg}}$, *Var*($\boldsymbol{X_{abg}}$), which are noticeably higher compared with the remaining ones. Therefore we can select components associated with the highest eigenvalues. A classical criterion is to take the eigenvalues higher than the average:

$$\lambda_k \geq \frac{\sum_{k=1}^{rank(X_{abg})} \lambda_k}{rank(X_{abg})} \tag{6.6}$$

However, when there is only random noise, eigenvalues of variance matrix of $\boldsymbol{X_{abg}}$ will be approximately equal and half of them will be higher than the average. Therefore we prefer to weigh the previous average by a coefficient $\beta > 1$:

$$\lambda_k \geq \beta \frac{\sum_{k=1}^{rank(X_{abg})} \lambda_k}{rank(X_{abg})} \tag{6.7}$$

In our results we have taken $\beta = 2$. The validity of this strategy was studied by comparing the gene selection given in this case, with those obtained when other selections of components were data-filtered. This study has been carried out with only one of the statistical methods, maSigPro methodology, in one of the simulation studies. The results, included in Annex 4, are discussed in the following section. As we have previously mentioned, there are other methods to select the number of components whose suitability in the performance of our filter method merits future research.

6.2.2 Datasets

6.2.2.1 Synthetic data

Two simulation studies have been performed to evaluate the ASCA filter. The first study is an independent time-course experiment while the second case simulates longitudinal data. Different scenarios have been considered in each case to resemble situations with different types and quantities of noise.

Independent time-course data. We have simulated datasets in several scenarios with the same structure, 3 experimental groups (1, 2 and 3), 3 time points and 4 replications. The datasets contained 410 genes with profile differences classified into 5 expression patterns: 100 genes with continuous induction for the three groups (pattern A), 100 genes with continuous induction for group 3 (pattern B), 100 genes with continuous repression for group 3 (pattern C), 100 genes with transitory induction for group 2 (pattern D) and a minority group, a group with only 10 genes with transitory induction for group 1 (pattern E). Additionally, there were 9590 flat profile genes without differences between experimental groups, making a total of 10000 genes. The replicates for each gene were produced as independent observations from a distribution $N(\mu_{ijn}, \sigma^2_{ijn}),\ i = 1,2,3;\ j = 1,2,3;\ n = 1,\ldots,10000$.

Although all the datasets have the same structure there are some differences among them regarding the noise. Different random noise has been considered with σ=0.05 and 0.1 for the simulated replicates. Systematic noise was introduced to the datasets in the same way that the simulated dataset in Chapter 5. This noise simulates a dye-swap experiment by splitting the dataset in two replicates and adding two opposite normal distributions to each half. Furthermore this structural noise has been added in two different quantities: low: ±N(0.1,0.05^2) and high: ±N(0.25,0.1^2). Therefore, six different scenarios have been studied and we refer them by the enumeration indicated in Table 6.1.

	Random noise	
Structural noise	σ =0.05	σ =0.1
None	1	2
Low: ±N(0.1,0.05^2)	3	4
High: ±N(0.25,0.1^2)	5	6

Table 6.1: Simulated scenarios.

Longitudinal data. We have also simulated longitudinal data to evaluate the performance of the ASCA filter with this type of time course data. Furthermore, as timecourse and EDGE methods have been designed for the analysis of longitudinal data, we could better examine the suitability of the ASCA filter with these techniques.

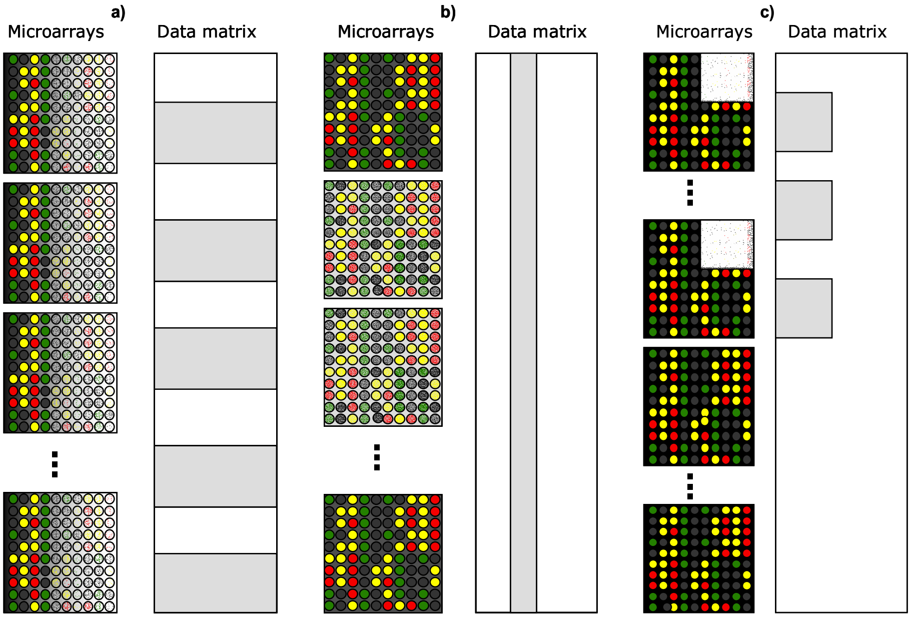


Figure 6.1: Examples of the alterations in the simulated chips. Shaded parts of the data matrix indicate wich data data will be affected by the alteration of the chip. a) Spatial differences inside all the chips. b) Several chips altered in the same way. c) Combination of both effects a) and b).

We decided to use the structure of the simulated longitudinal data described in the timecourse package documentation. This is a dataset with 5 time points, 3 experimental groups with 4, 3 and 2 repetitions each one of them making a total of 9 individuals and 45 chips. We simulated 500 changing genes, with the functions detailed in the timecourse package and 9500 flat genes yielding a total of 10000 genes. These functions generate the dataset by simulating multivariate normal data considering the same normal distribution for the average profile and the same correlation matrix between the individuals of the same experimental group. This produces a series of data with similar behaviour inside each experimental group which is different to the other groups. However, this method does not target selection to genes with the same time pattern for all groups. In our analysis we do consider this behaviour as differential expression and only label as non-interesting those genes which have flat profiles across all treatments.

Structural noise was added in different ways to resemble different problems that may appear in microarrays. In the first scenario no structural noise is added. The second case is a scenario where spatial differences are consistently present across all arrays (Figure 6.1 a). The quality of the glass or systematic hybridization problems can cause different intensities in all the chips of an experiment. As a result several genes will have higher or lower gene expression values in a constant value and therefore several rows of the data matrix will be altered by this constant. We have simulated this scenario by adding a normally distributed variable, N(3,1) to 5000 genes (50% of the total: 250 significant and 4750 flat genes). The third scenario simulates the experiment for which there are several chips affected globally by a constant value. This is the typical batch effect, which arises when different labs, persons or time-scheduling take part in the elaboration of the chips. Here, several columns of the data matrix will be modified (Figure 6.1 b). We have simulated this scenario by adding a normally distributed variable, N(3,1), to 15 chips. In the fourth scenario both effects are combined: spatial effects are only present in a subset (batch) of the arrays (Figure 6.1 c). We have simulated this case by adding to 5000 genes (50% of the total: 250 significant and 4750 flat genes) a normally distributed variable, N(10,2.25), to the expression of 5 individuals of different experimental groups. Finally, the fifth scenario simulates a dye swap as it has been described in the simulated study 1 with a normally distributed variable ±N(3,1).

6.2.2.2 Experimental datasets

We also show the application of this filter to two real experiments: the toxicogenomics study analysed in previous chapters and a larger transcriptomics experiment of abiotic stress on a plant system.

Toxicogenomics dataset. The first experimental dataset is the toxicogenomics study by Heijne *et al*. (2003) where the effect of the hepatotoxicant bromobenzene is studied. In this study there are 3 time points (6, 12 and 48 hours after the administration of the drug), 5 experimental groups (2 placebos and 3 different doses of bromobenzene) and 2665 genes. This dataset is described in detail in Chapter 2.

NSF potato stress dataset. The second experimental dataset corresponds to a stress study in plants that investigates the transcriptional response to three different abiotic stressors (Salt, Cold and Heat) in potato using the NSF 10k potato array (Rensink *et al*., 2005). A common reference design is also used in this case. The dataset has 4 series (1 Control and 3 types of stress: Heat, Salt and Cold), 3 time points, three replicates per experimental condition and 9993 genes.

6.2.3 The evaluation approach

The general strategy to evaluate the proposed filter was to apply a statistical method for TCM data (maSigPro, EDGE and timecourse) before and after applying the filter, and compare results in terms of feature selection. We have used sensitivity (true positives detected/real true positives) and specificity (true negatives detected/real true negatives) to quantify the performance of each pipeline analysis. A good selection of genes is obtained when both measures are close to 1. In the case of experimental data, as the true differentially expressed genes are not known, these metrics cannot be used. Instead, we have applied a functional enrichment analysis to evaluate the biological consistency of the results.

In the following, the time course methodologies used to demonstrate the performance of the proposed filter (maSigPro, timecourse and EDGE) are described. Since maSigPro is extensively described in Chapter 4, we will only include here a brief mention of basic characteristics. Timecourse and EDGE methods were introduced in Chapter 3 but not explained in detail. A more exhaustive description of these comparing methods is provided in this chapter.

maSigPro

The maSigPro approach fits a polynomial regression model that describes the evolution of gene expression over time. The methodology follows a two-step strategy. First, a global regression with the complete polynomial model is estimated for each gene applying least-squares technique. Differentially expressed genes are identified based on the FDR-controlled *p*-values of the *F*-statistic associated with the regression ANOVA. Secondly, a variable selection strategy is applied to identify gene-specific expression differences among experimental groups and to find statistically significant different profiles. At this stage genes might also pass a selection filter based on the *R*-squared value of the second regression model. This value measures the goodness of fit and therefore allows for the selection of genes with consistent expression trends. As a result, a refined gene selection is obtained, together with the set of significant regression coefficients of each selected gene.

TimeCourse

This methodology, proposed by Tai and Speed (2006), considers one and two sample multivariate tests about the mean vector $\boldsymbol{\mu}$ on different *k* time-points to study differences in the evolution through time in longitudinal time series. The hypothesis testing applied is the invariability of gene expression profile against non-constancy for one biological condition ($H_0 : \boldsymbol{\mu} = 0 \ vs \ H_1 : \boldsymbol{\mu} \neq 0$), and equality of the gene's mean expression levels versus the alternative that they are not ($H_0 : \boldsymbol{\mu_A} = \boldsymbol{\mu_B} \ vs \ H_1 : \boldsymbol{\mu_A} \neq \boldsymbol{\mu_B}$) under two biological treatments or conditions.

Based on the likelihood ratio statistic they develop a moderated likelihood ratio statistic and a moderated Hotelling T^2-statistic taking into account the information in all the available data to estimate the variance-covariance matrix by applying a multivariate empirical Bayes procedure. They consider the posterior odds for each gene that is the probability that H_1 is true divided by the probability that H_0 is true. Furthermore, they introduce a MB-statistic in both problems as the log base 10 of the posterior odds, which is the equivalent to the *B*-statistic in the univariate model (Lönnstedt *et al*., 2002 and Smyth, 2004) and is also called in genetics LOD score. When MB-statistic is positive it indicates that H_1 is more probable than H_0, therefore this gene will be of interest.

The methodology has been implemented in the R package timecourse available from Bioconductor. Additionally, the program also includes the corresponding implementation for multi-sample problems, generalizing the hypothesis of the two conditions case.

The method focuses on gene ranking, using the MB or T^2-statistics, and there is no significant assessment or a *p*-value computation. The gene ranking is obtained from the order of evidence of non-constancy for one condition and from the order of evidence of differences in evolution between conditions. Therefore, and for comparison purposes with the other statistical approaches, we defined the feature selection criterion with the timecourse methodology as positiveness of the MB statistics in the case of experimental data. In the case of simulated datasets the selection applied was the *n* first features, *n* being the number of genes simulated as significant. This was due to the fact that the MB values were negative in the first simulated study and there were big differences in positive cases between the original and filtered data in the second simulation study. Our purpose here is not to demonstrate the performance of this method but to investigate improvements in gene selection by ASCA filtering when timecourse is used as a statistical test and the ranking of genes is the main output of this method.

EDGE

Storey *et al.* (2005) proposed the use of B-splines to fit the same dimensional model to each gene where the coefficients are estimated by applying standard least squares regression techniques. They develop their method to handle both independent and longitudinal data. The method is implemented in the statistical language R in the software EDGE (Leek *et al.*, 2006).

Let x_{ij} be the relative expression level of gene *i* in individual *j.* For independent data they propose the model:

$$x_{ij} = \mu_i(t_j) + \varepsilon_{ij} \qquad (6.8)$$

where $\mu_i(t_j)$ is the population average time curve for gene *i* evaluated at time t_j and ε_{ij} the random deviation from this curve. $\mu_i(t)$ is parameterized in terms of an intercept plus a *q*-dimensional linear basis:

$$\mu_i(t) = \alpha_i + \boldsymbol{\beta}_i^T \boldsymbol{s}(t) = \alpha_i + \beta_{i1}s_1(t) + \ldots + \beta_{iq}s_q(t) \qquad (6.9)$$

where $\boldsymbol{s}(t)$ is a prespecified *q*-dimensional basis, α_i is the unknown gene-specific intercept and $\boldsymbol{\beta}_i^T = [\beta_{i1}, \ldots, \beta_{iq}]^T$ a *q*-dimensional vector of unknown gene-specific parameters. The authors indicate that a natural choice for this basis is a polynomial of degree *q*, that was effective in their studies, although a natural cubic spline basis is more flexible, and this option was chosen in our analysis. The curve is estimated by minimizing the sum of squares between the curve and the observed values. To decide between the null hypothesis of no differential expression ($\mu_i(t)$ constant) against the alternative of differential expression ($\mu_i(t)$ a curve) they consider a statistic for each gene analogue to the *F*-statistic used in maSigPro that compares the variation explained for the model with the residual variation. Finally they use bootstrap techniques to find the empirical distribution of the *F*-statistic and they compute a *Q*-value for each gene which estimates the FDR incurred when calling the gene significant (Storey and Tibshirani, 2003).

For longitudinal data, x_{ijk}, the relative expression level of gene *i* in individual *j* at the *k*-th time point, they use the model:

$$x_{ijk} = \mu_i(t_{jk}) + \gamma_{ij} + \varepsilon_{ijk} \tag{6.10}$$

being γ_{ij} the individual deviations from the average time curve $\mu_i(t)$. The term γ_{ij} can be modelled as a curve or as a constant. However, to model a curve enough observations need to be available.

6.3 Results

6.3.1. Simulation study 1: independent time-course data

In this simulation study, we first asked if the proposed criterion for the selection of a number of components in each ASCA submodel was adequate. We studied this in a pilot simulation where only one dataset was created for each of the proposed scenarios. Table 6.2 describes the percentage of variation simulated in each submodel and the explained percentage of variation captured by each submodel when the proposed criterion for component selection is applied (the actual number of components obtained by this criterion is indicated in the table). We can observe how the residual variation increases with more noisy scenarios. When gene selection on these simulated data is obtained by the ASCA-genes approach we observed that selection becomes unsatisfactory when high levels of structural noise and low signal-ration values are present. This study is provided as supplementary material in Annex 3 and will not be discussed further here, although results indicated the necessity of adapting ASCA-genes feature selection strategy to high noise / low signal scenarios.

	%Variation			Number of	% Explained in each model		
SCENARIO	Xa	Xbab	Xres	Components	Xa	Xbab	Xres
1	33.6	39.8	26.6	1,2,0	89.7	89	0
2	19.4	28.2	52.3	1,3,0	83.5	71.7	0
3	12.3	18.9	68.8	1,2,1	89.7	89.9	79.8
4	10.5	18.3	71.2	1,3,1	83.7	79.5	57.1
5	3	7.8	89.2	1,2,1	89.6	83.5	92.6
6	3.3	8.6	88.1	1,2,1	83.7	75	86

Table 6.2: Percentage of variation simulated in each submodel with independent data and % explained variation by applying ASCA with the number of components indicated in each submodel.

maSigPro

As indicated before, we used this pilot study to evaluate the adequacy of the proposed criterion for a number of component selection. We include the different solutions in Annex 4, in terms of FN, FP, sensitivity and specificity, obtained with the proposed and other numbers of components for each submodel and with different *R*-squared levels (0.6, 0.7 and 0.8). We can observe that the proposed criterion provides an efficient selection of differentially expressed genes in all scenarios although there are also other good solutions. We can also observe in Annex 4 that as the number of components of submodel b+ab or submodel abg increases the number of selected genes also increases. This is related to the definition of the *F*-statistic used for gene selection in maSigPro:

$$F = \frac{\text{Explained variability with the model}}{\text{Residual or non explained variability}}$$

As the maSigPro model tries to explain the variability in $\boldsymbol{X}_a$ and $\boldsymbol{X}_{b+ab}$, if this variability increases, the *F*-statistic also increases and there will be more genes selected because the associated *p*-value exceeds the prefixed threshold. On the other hand, by increasing the number of components in the submodel *abg*, the considered residual variability in the associated regression model decreases, thereby the *F*-statistic will increase and consequently so will the number of selected genes.

Table 6.3 shows the results obtained with the proposed number of components taking *R*-squared=0.8 (results with *R*-squared of 0.6 and 0.7 are included in Annex 4 and lead to similar conclusions). The election of the cut-off levels for a method depends on the proportion of genes that it is expected to be obtained and the level of noise in the dataset. In the following we try to maintain the same levels for the different techniques, but this not always result in comparable analysis scenarios. Considering the sensitivity and specificity indicators we can state that in all scenarios the selection of genes applying maSigPro to the filtered data is equal or better than the selection with maSigPro to the original data. In scenarios where there is no systematic noise, maSigPro behaves efficiently both in the original and the filtered data, thus the ASCA filter does not affect the original good results. On the other hand, in scenarios with high noise, the ASCA filter clearly improves sensitivity, specificity being unaffected. Taken together we can conclude that this pilot study suggests that the ASCA filter is an efficient way of improving the detection of differentially expressed genes in high noise scenarios and it has no significant effect when noise is limited.

SCENARIO	DATA	SELECTION	FP	FN	SENSIT	SPECIF
1	Original	410	0	0	1.000	1.000
	Filtered	**410**	**0**	**0**	**1.000**	**1.000**
2	Original	410	0	0	1.000	1.000
	Filtered	**399**	**0**	**11**	**0.973**	**1.000**
3	Original	369	0	41	0.900	1.000
	Filtered	**409**	**1**	**2**	**0.995**	**1.000**
4	Original	281	0	129	0.685	1.000
	Filtered	**389**	**0**	**21**	**0.949**	**1.000**
5	Original	8	0	402	0.020	1.000
	Filtered	**558**	**206**	**58**	**0.859**	**0.979**
6	Original	4	0	406	0.010	1.000
	Filtered	**311**	**3**	**102**	**0.751**	**1.000**

Table 6.3: maSigPro results with original and filtered data for the 6 simulated scenarios with independent data by using FDR=0.05 and *R*-squared=0.8.

We confirmed the results of this pilot experiment by running an extended simulation study in which each scenario was simulated 10 times summing up 60 simulations. Again, we applied maSigPro to each original dataset and to the ASCA-filtered datasets at different *R*-squared threshold levels (0.6, 0.7 and 0.8) and collected selected genes, making a total of 360 analyses. We applied ANOVA to the data obtained in this study to assess the statistical significance of the differences in specificity and sensitivity observed for the involved factors: Data (original or filtered), Scenario (1 to 6) and *R*-squared (0.6, 0.7 or 0.8). ANOVA showed that all factors and some interactions (mainly Scenario x Data) were statistically significant (*p*-values < 0.05, Annex 5). The ASCA-filter pre-processing improves the average sensitivity by 0.23 when maSigPro is used as a method for gene selection. However, although the average decrease of the specificity is also statistically significant, its magnitude is very limited (0.01), and is not considered relevant.

The most important result is that filtered data is less sensitive to noise than original data and there is no practical difference in specificity. This is shown in Figure 6.2 where 95% Least Significance Difference (LSD) intervals and average sensitivity and specificity evolution through the different scenarios obtained with all the levels of *R*-squared for filtered and original data are shown. In Annex 5 details of these results have been included. The extended study confirmed the results and pattern observed in the pilot study: maSigPro sensitivity is good at low noise levels (scenarios 1 and 2) but decreases when high structural noise is present (scenarios 5 and 6). Sensitivity is restored in these cases by applying the ASCA filter while this pre-processing has little effect in scenarios where maSigPro alone is already effective.

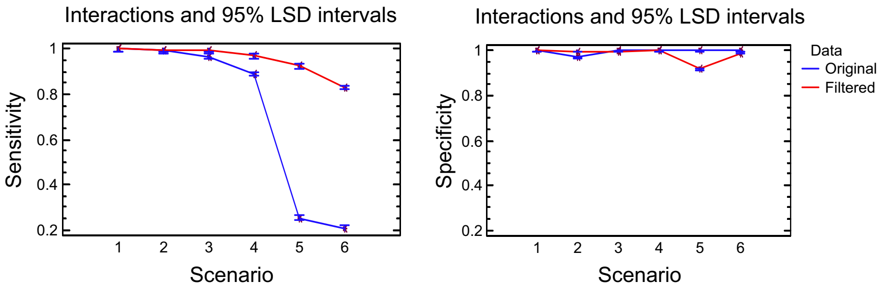


Figure 6.2: Interaction Scenario×Data for Sensitivity (left) and Specificity (right). The graphs represent the 95% Least Significance Difference (LSD) intervals obtained with the different levels of *R*-squared.

Timecourse

The application of the timecourse package to the simulated datasets showed negative MB-values in all cases. As this package does not provide a criterion to obtain a statistically significant list of genes and the criterion of taking genes with positive values was not applicable, we chose the first 410 genes of the ranking offered by the package. However, we know that the method aims at selecting genes with differences in trends between experimental groups and those genes with similar and significant trends for all the groups are not considered of interest. Therefore 100 genes simulated as Pattern A would not be target by the method. Therefore, we have computed as solution the first 310 genes simulated in Patterns B, C, D and E. By doing this, false positives and false negatives are always based on the same number. Sensitivity and specificity are also computed considering 310 genes as the right solution (Table 6.4). We can observe that in scenarios without structural noise the results with original and filtered data are equal. In scenarios 4 and 5 there is a slight improvement in sensitivity by applying the ASCA-filter, and in scenarios 3 and 6 this improvement becomes more evident.

SCENARIO	DATA	SELECTION*	FP	FN	SENSIT*	SPECIF*
1	Original	310	0	0	1.000	1.000
	Filtered	**310**	**1**	**1**	**0.997**	**1.000**
2	Original	310	0	0	1.000	1.000
	Filtered	**310**	**0**	**0**	**1.000**	**1.000**
3	Original	310	60	60	0.806	0.994
	Filtered	**310**	**0**	**0**	**1.000**	**1.000**
4	Original	310	6	6	0.981	0.999
	Filtered	**310**	**2**	**2**	**0.994**	**1.000**
5	Original	310	68	68	0.781	0.993
	Filtered	**310**	**58**	**58**	**0.813**	**0.994**
6	Original	310	90	90	0.710	0.991
	Filtered	**310**	**57**	**57**	**0.816**	**0.994**

Table 6.4: Timecourse results with original and filtered data for the 6 simulated scenarios with independent data. (*) Gene selection is obtained by choosing the first 310 genes of the ranking provided by the Timecourse package. FP, FN, sensitivity and specificity is computed by considering 310 genes as the right solution.

EDGE

EDGE does provide a *p*-value and multiple testing correction in its output. We used a *Q*-value cut-off of 0.05 to declare genes significant. Results are presented in Table 6.5. We can observe that the number of false negatives is 100 genes in many cases. To a great extent these are genes simulated with pattern A, cases with the same gene-expression evolution, which are hard to detect by this method. In general, we observe that sensitivity is lower than in the other two methods. Pre-processing of data by ASCA-filtering improves detection capacity in some scenarios, although sensitivity values are still reduced. Similar to other methodologies, the specificity of the method is good regardless of the pre-treatment of the data.

SCENARIO	DATA	SELECTION	FP	FN	SENSIT	SPECIF
1	Original	983	667	94	0.771	0.930
	Filtered	**366**	**58**	**102**	**0.751**	**0.994**
2	Original	1058	744	96	0.766	0.922
	Filtered	**428**	**117**	**99**	**0.759**	**0.988**
3	Original	310	0	100	0.756	1.000
	Filtered	**566**	**255**	**99**	**0.759**	**0.973**
4	Original	332	21	99	0.759	0.998
	Filtered	**363**	**53**	**100**	**0.756**	**0.994**
5	Original	279	0	131	0.680	1.000
	Filtered	**444**	**211**	**177**	**0.568**	**0.978**
6	Original	254	0	156	0.620	1.000
	Filtered	**210**	**9**	**209**	**0.490**	**0.999**

Table 6.5: EDGE results with original and filtered data for the 6 simulated scenarios with independent data taking genes with *Q*-values≤0.05.

6.3.2. Simulation study 2: longitudinal data.

One dataset per scenario was simulated to study the effects of the ASCA-filter on longitudinal data. We first applied the ASCA methodology to these datasets to explore the variation created in each submodel. Table 6.6 describes the percentage of variation simulated in each submodel and the explained percentage of variation in each case with the number of components indicated. The structural noise added to the data in scenario 2 (in which we included a deviation by genes) is not introducing any structural noise in the residuals (submodel $\boldsymbol{X}_{abg}$). As we simulated this scenario by adding a normally distributed variable to half of the genes, the ANOVA model considered for each gene captures the deviation of gene expression average and the residuals of the model are increased randomly.

	%Variation			Number of	% Explained in each model		
SCENARIO	Xa	Xbab	Xres	Components	Xa	Xbab	Xres
1	10.6	26.1	63.3	3,8,0	75.7	82.9	0
2	11	26	63	3,8,0	75.6	82.6	0
3	8.8	21.9	69.3	3,8,1	76	82.7	27
4	4.7	12.9	82.4	3,8,1	76	84	66
5	5.8	14.2	80	3,8,1	75.6	82.4	57.2

Table 6.6: Percentage of variation simulated in each submodel with longitudinal data and % explained variation by applying ASCA with the number of components indicated in each submodel.

Table 6.7, Table 6.8 and Table 6.9 show the results offered by maSigPro, timecourse and EDGE respectively applied to the longitudinal simulated datasets. Timecourse selection has been obtained by choosing the first 500 genes of the obtained ranking. On the other hand maSigPro has been applied by using FDR=0.05 and *R*-squared=0.6, and EDGE with *Q*-value cut-off=0.05.

By comparing false positives and negatives of the original and filtered data results we can observe, also in the case of longitudinal data, that ASCA-filter improves sensitivity. This improvement is more well known in scenarios with structural noise meanwhile in scenarios without noise results do not change substantially. We can also observe that maSigPro with filtered data is the strategy that offer the best results.

SCENARIO	DATA	SELECTION	FP	FN	SENSIT	SPECIF
No noise	Original	512	14	2	0.996	0.999
	Filtered	**498**	**3**	**5**	**0.990**	**1.000**
Rows	Original	509	12	3	0.994	0.999
	Filtered	**499**	**5**	**6**	**0.988**	**0.999**
Columns	Original	460	2	42	0.916	1.000
	Filtered	**499**	**4**	**5**	**0.990**	**1.000**
%chips	Original	249	1	252	0.496	1.000
	Filtered	**492**	**6**	**14**	**0.972**	**0.999**
Dye-Swap	Original	87	0	413	0.174	1.000
	Filtered	**489**	**3**	**14**	**0.972**	**1.000**

Table 6.7: maSigPro results with original and filtered data for the 5 simulated scenarios with longitudinal data by using FDR=0.05 and *R*-squared=0.6.

SCENARIO	DATA	SELECTION	FP	FN	SENSIT	SPECIF
No noise	Original	500	49	49	0.902	0.995
	Filtered	**500**	**51**	**51**	**0.898**	**0.995**
Rows	Original	500	64	64	0.872	0.993
	Filtered	**500**	**74**	**74**	**0.852**	**0.992**
Columns	Original	500	83	83	0.834	0.991
	Filtered	**500**	**56**	**56**	**0.888**	**0.994**
%chips	Original	500	171	171	0.658	0.982
	Filtered	**500**	**126**	**126**	**0.748**	**0.987**
Dye-Swap	Original	500	80	80	0.840	0.992
	Filtered	**500**	**68**	**68**	**0.864**	**0.993**

Table 6.8: Timecourse results with original and filtered data for the 5 simulated scenarios with longitudinal data. (*) Gene selection is obtained by choosing the first 500 genes of the ranking provided by the timecourse package.

SCENARIO	DATA	SELECTION	FP	FN	SENSIT	SPECIF
No noise	Original	317	9	192	0.616	0.999
	Filtered	**330**	**2**	**172**	**0.656**	**1.000**
Rows	Original	323	16	193	0.614	0.998
	Filtered	**340**	**11**	**171**	**0.658**	**0.999**
Columns	Original	339	22	183	0.634	0.998
	Filtered	**312**	**5**	**193**	**0.614**	**0.999**
%chips	Original	300	12	212	0.576	0.999
	Filtered	**332**	**11**	**179**	**0.642**	**0.999**
Dye-Swap	Original	283	10	227	0.546	0.999
	Filtered	**334**	**8**	**174**	**0.652**	**0.999**

Table 6.9: EDGE results with original and filtered data for the 5 simulated scenarios with longitudinal data selecting cases with *Q*-values≤0.05.

6.3.3 Toxicogenomics dataset.

In the exploratory analysis of the original toxicogenomics dataset, showed in Chapter 2, we detected a considerable dye effect. This structural noise was removed by centering each gene with its corresponding dye average. This centered data was used in inferential analysis described in Chapter 2 and also in maSigPro analysis. The application of the ASCA model to this centered data revealed some additional structural noise which had not been removed. In this study, we used the original data, without centering, to identify and remove structural noise by ASCA filtering, since this implies a single treatment of all structural noise. The proposed filter will be removing the structural noise due to the dye and other possible structures in an effective way. Consequently we are comparing the sophisticated ASCA-filter with the simple transformation done by centering the data with the dye averages. The component selection procedure described in the previous section gave as a result one, five and two components for submodel *a*, *b+ab* and *abg* respectively. This selection explains 75% of time variation, 78% of experimental groups plus interaction variation and 48% of the residuals variation.

	DATA	maSigPro	timecourse	EDGE
GENE SELECTION	Centered	155	237	298
	Filtered	180	189	206
	Coincidences	136	158	181
	Only Original data	19	79	117
	Only Filtered data	44	31	25

Table 6.10: Gene selection and comparisons obtained with the different approaches on original and filtered data of toxicogenomics dataset. maSigPro has been applied with FDR=0.01 and R^2=0.6 and EDGE with *Q*-values≤0.01.

Table 6.10 shows gene selection obtained with the different methods in original and filtered data and comparisons between both types of data. Timecourse selection has been obtained by choosing genes with a positive MB-statistic. On the other hand maSigPro has been applied by using FDR=0.01 and *R*-squared=0.6, and EDGE with a *Q*-value cut-off=0.01.

The enrichment functional analysis identified a varying number of functional categories for each of the methods (detailed in the Supplementary material Chapter 6 at http://www.ua.es/personal/mj.nueda). Several of these functional categories have parent-child relationships or are closely related. In Table A-9, included in Annex 6, we provide a subset of categories that try to be semantically orthogonal, i.e., reflect different cellular activities or processes. The first conclusion of the enrichment analysis is that a number of biological proceses were detected by any of the three methods with or without ASCA-filter. These are general terms such as *ribosome*, *translation* and *cytosol.* Other classes, describing more specific processes, were only identified by one or two methods as *oxidoreductase activity*, detected by maSigPro and EDGE, or *retinol binding* and *nitric oxide mediated signal transduction,* detected only by maSigPro. However, the most interesting result of this analysis is the nature of the additional processes that showed up when the ASCA-filter was used and that represent functions of known involvement in the toxicological response. These additional GO categories do not coincide in all three methods but do represent relevant processes to the biological problem under study and somehow provide a more complete functional picture of the process than when the filter is not applied. These processes include *heme binding, lipid metabolic* and *glutathione transferase activity,* with maSigPro, *oxidoreductase activity* and *nitric oxide mediated signal transduction,* with timecourse, and *glutathione transferase activity,* with EDGE.

6.3.4 NSF potato stress dataset

Again, filtered data is obtained by applying the component selection procedure described in the previous section. This choice gave as a result two, three and two components for submodel *a*, *b+ab* and *abg* respectively. This selection explains 100% of time variation, 80% of experimental groups plus interaction variation and 28% of the residuals variation.

	DATA	maSigPro	timecourse	EDGE
GENE SELECTION	Original	409	890	526
	Filtered	828	721	25
	Coincidences	368	602	19
	Only Original data	41	288	507
	Only Filtered data	460	119	6

Table 6.11: Gene selection and comparisons obtained with the different approaches on original and filtered data of NSF potato stress dataset. maSigPro has been applied with FDR=0.01 and R^2=0.8 and EDGE with Q-values≤0.01.

Gene selection obtained with the different methods in original and filtered data and comparisons between both results are shown in Table 6.11. Timecourse selection has been obtained by choosing genes with MB-statistic positive. On the other hand maSigPro has been applied by using FDR=0.01 and *R*-squared=0.8, and EDGE with a *Q*-vaule cut-off=0.01.

The enrichment functional analysis (detailed in the Supplementary material Chapter 6 at http://www.ua.es/personal/mj.nueda) showed firstly that the selection obtained with maSigPro in filtered data has over-represented a higher number of functional categories than the cases obtained with the original data. Timecourse in filtered data also has more categories than the original data in spite of the lower number of genes detected. However, the poor gene selection obtained with EDGE and filtered data do not show any over-represented functional category. Table A-10, included in Annex 6, lists the selected functional classes with each inference methodology with and without ASCA-filter pre-processing. We can observe a limited level of functional coincidences between the 3 methods that correspond with basic stress response functions. However, when data is cleaned by ASCA, maSigPro and timecourse analysis are able to reveal a much more specific view on the actual regulated processes, such as hormone signalling cascades, diverse enzymatic activities, specific metabolic functions and/or binding activities.

6.4 Discussion

In this chapter we show how a model-based multivariate projection technique ASCA can be used to pre-process microarray data. The rational of the methodology is the identification and removal of structured signals that cannot be associated with the experimental factors designed in the transcriptomics study. This structural noise is habitually referred to as the “batch” effect and corresponds to dye, lab, experimentalist, etc. The ASCA-filtering strategy uses ASCA to identify per gene signals associated with experimental factors and PCA to separate structured and random variation in these signals. By removing from the original dataset the non-structured part of the experimental factor signals and the structured variation of the factor-free ANOVA errors, we create a filtered dataset that is enriched in the information of interest and only keeps the random noise needed for inferential analysis.

The function of this filter was analyzed in two simulation studies where independent and longitudinal data, respectively, were mimicked. As we were interested in the way ASCA-filter removes noise from expression signals, different types “systematic and random” and the amount of noise “high or low” were introduced in the synthetic data. Additionally, we asked whether the filter would be generally valid, irrespective of the inference methodology used to identify differentially expressed genes. Therefore, we tested the filter with three published methods for the analysis of TCM data that follow very different statistical strategies. maSigPro applies polynomial regression, timecourse is based on empirical Bayes and EDGE uses B-splines to model the time gene expression patterns. The results showed that by filtering the data with ASCA there are great improvements in gene selection when a high quantity of structural noise is simulated, and there is no effect when no structural noise is present in the data. Although this pattern was observed for each of the three statistical methodologies, maSigPro is clearly the method where the ASCA-filter had a greater impact and is also the analysis strategy with the best end result, i.e. sensitivity increase with timecourse and EDGE were not as large as with maSigPro and the amount of differential expression detected when either methodology was applied in combination with the ASCA filter never reached the sensitivity levels obtained by the ASCA-maSigPro. This difference could be due to the fact that maSigPro applies independent analysis gene by gene without exploiting the correlation structure of the data and considering the noise as random. As the ASCA filter precisely exploits this correlation aspect, the benefit of combining the two methodologies is maximal. However, both timecourse as EDGE employ empirical methods to determine the statistical significance of the correspondending statistics used, which implicitly brings the consideration of possible structural noise in all data. On the other hand, as timecourse applies shrinking covariance estimates, it is considered the correlation structure of the data. In some way these methods take into account those aspects that are used by the proposed filter and therefore its impact is not as well known as in maSigPro. Still the ASCA filter improves sensitivity of timecourse and EDGE. We argue that this is related to the more refined treatment ASCA gives to variation by imposing an ANOVA model prior to component analysis. This decomposition allows for a more efficient, experimental factor focussed, analysis of covariance, in comparison to a design-blind consideration of correlation structures that would operate in timecourse and EDGE methods. Finally, we should mention that EDGE uses B-splines that, as already discussed in chapter 3, work well with series with more than 10 time-points and our simulations in this study restricted to short series, of 3 to 5 time-points.

When applied to experimental datasets, pre-processing by ASCA filter improved the identification of the affected biological processes by the maSigPro and timecourse methodologies, although this was not so clear when inference analysis was done by EDGE. The improvement in the two first cases was not associated with an increase in the number of genes declared as significant, as this only occurred with maSigPro but not with timecourse, but presumably because the better filter revealed the coordinative behaviour of genes belonging to the same functional class or because the filter eliminated noisy or poorly correlated genes that polluted functional enrichment analysis. Both in the case of toxicogenomics and abiotic stress datasets application of the filter not only increased the number of enriched functional class, but more importantly, the semantic diversity of the class selection, offers a more complete and detailed picture of the functional processes affected in the experiments. The reason why this did not happened with the EDGE methodology is unclear.

Chapter 7

Functional assessment of time course microarray data

The work presented in this Chapter is included in:

Nueda, M.J.; Sebastián, P.; Tarazona, S.; García-García, F.; Dopazo, J.; Ferrer, A. and Conesa, A. (2009) Functional Assessment of Time Course Microarray Data. BMC-Bioinformatics, in Press.

7.1 Introduction

In the previous chapters we have addressed the analysis of microarray time series data upon different considerations and statistical models (regression/univariate, latent-factor/multivariate and the combined). In all cases, statistical analysis has focussed on the modelling of gene expression patterns and on the identification of differentially expressed genes. This orientation, though valid and useful, solves only one (frequently the first) requirement to understand transcriptomics changes from any kind of microarray experiment. In most cases, the analysis proceeds through the identification of cellular processes and functions which are represented by the gene selection, i.e. genes are identified by their functional role and the question is then which functional modifications can be derived from the observed gene regulation. The incorporation of functional information into data analysis is normally obtained by the use of functional annotation databases that define and assign function labels to known genes. The most widely used functional annotation scheme is the Gene Ontology (GO) (Ashburner *et al*., 2000), which characterizes genes for their molecular functions (MF), cellular locations (CC) and involved biological processes (BP), but others such as the KEGG metabolic pathways (Kanehisa *et al*., 2004), transcription factor targets (Wingender *et al*., 2000) or Interpro functional motifs (Mulder *et al*., 2003) can be also employed to query for specific biological questions. This functional assessment aspect is traditionally handled in microarray data analysis via the so-called enrichment analysis: the list of significant genes is interrogated for the over (and/or under) abundance, as compared to the entire genome represented in the array, of the considered functional categories. In time course microarray data, this strategy could be similarly followed for the set of time-dependent differentially expressed genes (for example, as provided in the time course module of the GEPAS suite, Tárraga *et al*., 2008), or for the distinct clusters this gene selection can be divided into (available in STEM package, Ernst and Bar-Joseph, 2006). As a matter of fact, gene enrichment analysis is used many times to validate the results of a gene selection or a clustering strategy (Azuaje *et al.*, 2006 and Dopazo, 2006).

This strategy for functional evaluation of differential gene expression has a number of limitations (Dopazo, 2008). First, the functional enrichment analysis is greatly dependent on the definition of an arbitrary threshold for significance and gene selection, and eventually on the clustering strategy of choice. The threshold aspect has been overcome in two class experiments through the Gene Set Analysis approaches (GSA), which evaluate functional enrichment over a rank rather than a selection of genes (Al-Shahrour *et al*., 2005; Mootha *et al*., 2003; Subramanian *et al*., 2005 and Al-Shahrour *et al.,* 2007). To our knowledge, no equivalent approach is yet available for time series data. Secondly, functional assessment is done after gene selection and therefore does not allow for a direct evaluation of expression changes as gene functions which might obscure relationships between functional categories and ignore significant sub-patterns of variations within the functional class.

In this chapter we have set out to address the problem of the functional assessment of gene expression in time series data in an alternative manner. We have developed and tested three distinct strategies which respond differently to the various concerns mentioned above. The proposed methods derive from previous methodologies developed in this thesis for the analysis of short, multiple series data, which follow a gene-centric orientation: the maSigPro, a two-step regression approach; and the ASCA-genes, a multivariate method that combines ANOVA decomposition with Singular Component Analysis. In this study, we first assess the fully correlated nature of the functional category in a modification of the maSigPro methodology to directly model the combined expression of genes belonging to the same functional class (maSigFun). In a second approach, we consider the possibility of different patterns of coordinative time-dependent gene expression variation within the functional class and the selection of those with a significant change (PCA-maSigFun). Finally, we develop an adaptation of the GSA strategy to time series through the identification of main patterns of variation in the dataset and the ranking of genes according to their correlation to such patterns (ASCA-functional).

As in previous chapters, we have used both synthetic and experimental datasets to assess the different methods. Simulated data provides a means of understanding the working of the methodologies, while experimental data offered insights on the biological relevance of the strategies. Algorithms have been implemented in the R language and are available at http://bioinfo.cipf.es/downloads and also at http://www.ua.es/personal/mj.nueda. Supplementary material of this Chapter is also available at http://www.ua.es/personal/mj.nueda.

7.2 Methods

7.2.1 maSigFun

This methodology derives from maSigPro, a regression-based approach for the analysis of multiple series time-course microarray data (Chapter 4). The maSigPro method follows a two-stage regression strategy to model gene expression and select differential expressed genes: the first step uses a generic polynomial model to spot responsive genes, while the second applies step-wise regression to disclosure the patterns of significant differential time profiles.

The adaptation of maSigPro to consider functional information -maSigFun- is quite straightforward: the regression model is fit not gene-wise as in maSigPro, but to the data matrix composed by the expression values of all genes belonging to the functional class, thereby being one regression model fitted for each functional category. In this approach individual genes are considered as different observations of the expression profile of the class. As genes belonging to the same class may show different basal expression levels and this may influence negatively model parameters estimation, expression data is standardized gene-wise to better capture the correlation structure within the functional group (Figure 7.1.a). After this transformation, statistical analysis proceeds as in regular maSigPro. The expected result is that significant functional classes are those whose genes change their expression along time in the same manner, i.e. a high level of co-expression is present within the functional class.

7.2.2 PCA-maSigFun

In this strategy we consider that a functional block might display not only one but several patterns of coordinative gene expression. These distinct patterns are extracted following a strategy similar to that proposed by Conesa *et al.* (2008) to directly link gene function to the phenotype. Basically, the strategy applies Principal Component Analysis (PCA) to the gene expression matrices composed by all genes belonging to the same functional class. PCA modeling will dissect orthogonal, time-dependent, transcriptional patterns contained in the class, and a number of those will be selected. The selection criterion implemented in the PCA-maSigFun method follows the rationale of retaining patterns that represent non-random variation. Considering the general assumption that holds in transcriptomics analysis of global invariability in gene expression, a good estimate of noise level variation would be the mean gene variance across the complete dataset. Therefore, for each functional-class associated PCA, selected components are those having a normalized explained variance above this mean gene variance value. The scores vector of each component depicts an expression pattern that corresponds to a correlated gene subset of the functional class and can be taken as transformed expression values for that subset. All so-obtained scores vectors are collected into a matrix of function-labeled "synthetic genes" which is then subjected to regular maSigPro for regression-based statistical analysis. Selected features will therefore correspond to defined function patterns that show a significant association with the time (Figure 7.1.b). Once significant functional features are obtained the question is how individual genes relate to these significant patterns. This information can be obtained from the analysis of the gene loadings in each PCA model. Genes with a high absolute loading value in a given selected component will have an important contribution to the associated profile and therefore can be considered as members of the gene subset that defined that correlated pattern of the class. Genes with a low absolute loading value will not correspond to this subset. In the current PCA-maSigFun implementation, a value for loading cutoff is derived by bootstrapping over the whole dataset to create a null loadings distribution across all functional classes and defining an arbitrary threshold (typically the 95% percentile) to declare a gene significantly contributing.

7.2.3 ASCA-Functional

As we have shown in Chapter 5 ASCA analysis provides a PCA submodel for each experimental factor -time, treatment and the interaction- that encompass all genes in the dataset and collects most of the variability associated with each experimental factor. In ASCA-functional these models are used to create ranks of genes that can be subjected to GSA analysis. In this sense, the third proposed approach can be considered as an adaptation of GSA methods to situations when not only two, but more experimental conditions are involved, as it is the case of (multiple series) time course data. In two class data, genes are ranked according to a measure of differential expression such as fold change, a *t*-statistics or a similar statistics. Enrichment analysis is performed along this rank by assessing the differential distribution of each functional block along the ranked gene list. In the case of ASCA-functional, ASCA-genes is first applied to create PCA submodels associated with each experimental factor. Similarly to the previous method, the genes loadings at each component of each submodel are a measure of the similarity of each particular gene expression profile to the pattern depicted by the component of the submodel. Genes with high positive loadings will greatly follow the pattern indicated by the component, genes with high negative loadings follow an opposite pattern, while genes with loadings close to zero do not resemble the behavior represented by the principal component. Those bad modelled genes are identified in ASCA-genes by their high SPE (Chapter 5) and are assigned a loading value of 0. The gene loadings offer therefore the way to rank genes according to specific patterns of variation which corresponds to biological phenomena. This ranking can be then subjected to GSA analysis. In our particular implementation, ranked lists are analyzed by the partitioning method FatiScan (Al-Shahrour *et al.,* 2005 and Al-Shahrour *et al.,* 2007) to identify functional categories associated with specific time patterns (Figure 7.1.c).

7.2.4 Datasets

Synthetic and experimental datasets were used to assess the proposed methods. Synthetic data was designed to depict different scenarios of co-expression while the experimental sets reflect two microarray studies involving different probe sizes and biological systems.

Simulated datasets

Two simulation studies were designed to evaluate the effect of class size and percentage of co-expressed genes in the identification of time-course changing functional categories. Both studies use the same primary data structure. The hypothetical experiment contained two series (Control and Treatment) and three time-points (0, 1 and 2). Synthetic datasets consisted of a total of 10000 genes in study A and several sizes in study B, distributed in 250 classes from which 225 classes contain only flat genes and 25 classes include at least one differentially expressed gene. Modelled responsive genes follow one of four possible patterns of expression: 1) Flat profile for control and continuous induction for treatment, 2) Flat profile for control and continuous repression for treatment, 3) Flat profile for control and transitory induction for treatment and 4) Flat profile for control and transitory repression for treatment. In all of the 25 classes with some non-flat genes only one of the four patterns is present, meaning that all changing genes in the class follow the same profile, have a positive correlation and could be regarded as “co-expressed”. At each individual simulation, noise was introduced in the datasets by adding to the defined profiles random values taken from a normal distribution N(0, 0.01).

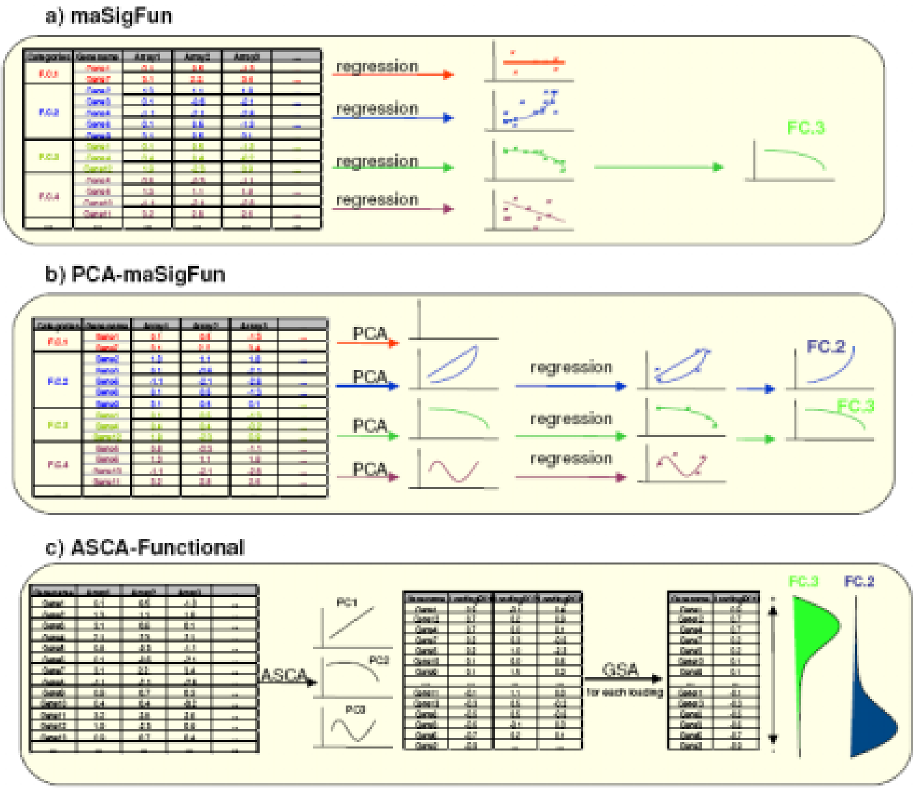


Figure 7.1: Schematic representation of the proposed methods a) **maSigFun** fits a regression model for each gene expression submatrix defined by the genes annotated to a given functional class (FC.1 to 4 in scheme). Significant classes are obtained by the maSigPro method (FC.3). b) **PCA-maSigFun** obtains a PCA model for the gene expression submatrix defined as in maSigFun and extracts a number of components that collect non-random variation. Generally 0 (FC.1) to 2 (FC.2) components are extracted for each functional class. A regression model is then fitted to the scores vector of extracted components to select function-defined patterns with a significant association to time (FC.2 and FC.3). c) **ASCA-functional** applies ASCA-genes to identify principal patterns of variation associated with time and time x treatment experimental factors (PC1 to 3 in scheme). Genes are ranked by loading value in each PC and GSA analysis is applied to each loading value ordered gene list to identify a functionally related block of genes associated with the principal patterns of variation (FC.2 and FC.3).

The first simulated study (A) analyzes how the percentage of co-expressed genes within the functional class affects the identification of the category. In this study, functional classes were varying in size (number of genes), taking values from 5, 10, 30, 55 and 100. Seven different datasets were created in this study, each of them with a different percentage of co-expressed genes (20, 30, 40, 50, 60, 70 and 80%) for all of the 25 non-all-flat classes present in the dataset. For example, dataset A-40 has 10000 genes distributed in 250 classes of different size from which 25 classes have all a 40% of genes which follow the same changing profile and 60% of the genes that are flat. In the remaining 225 classes of dataset A-40, 100% of the genes are invariant. Fifty simulations were run per each of the seven proportion levels.

In the second simulated study (B) we evaluated the effect of the class size. Here, 4 x 3 datasets were created, each of them having a fixed value for the class size (5, 10, 50 and 100) and a fixed value for the percentage of genes with change (30, 50, and 70%). For example, dataset B-50-70 contains 250 classes of size 50 (12500 genes), from which 25 classes have 35 genes with a defined changing profile and 15 flat genes, while the remaining 225 classes of dataset B-50-70 have all 50 genes flat. Again, 50 simulations are run for each size and proportion levels.

Experimental datasets

Two experimental datasets were selected for the evaluation of the methodologies on real data: the toxicogenomics dataset used in previous chapters, as this has been repeatedly characterized and the involved biological pathways are to a great extent known, and the potato dataset introduced in Chapter 6 which has a larger number of genes and a different pattern of variation across series. Briefly, the toxicogenomics dataset evaluates transcriptome response in rat liver to increasing doses of the drug bromobenzene (BB). In this study 2-6 rats were sacrificed after 6, 24 or 48 hours of drug exposure to extract liver mRNA which was then labeled and hybridized to a custom cDNA using a dye-swap design with a common reference. The dataset consists of 3 time points, 5 series (HIgh, LOw and MEdium BB levels, UnTreated and Corn Oil vehicle controls) and 2665 genes (Heijne *et al*., 2003). The second experimental dataset corresponds to a stress study in plants that investigates the transcriptional response to three different abiotic stressors (Salt, Cold and Heat) in potato using the NSF 10k potato array (Rensink *et al*., 2005). Also a common reference design is used in this case. The dataset has 4 series (3 treatments plus one Control), 3 time points and three replicates per experimental condition.

7.3 Results

7.3.1 Simulation studies

For either simulation study, fifty datasets of each type were generated and analyzed by the three proposed methods. For each observation, the identified categories were recorded and considering the 25 non-all-flat classes as “true positives”, values of false positives (FP), false negatives (FN), sensitivity (proportion of actual positives which were correctly detected) and specificity (proportion of negatives which were correctly identified) were computed. In all methods significance threshold were set to 0.05 false discovery rate (FDR).

maSigFun

In the case of maSigFun analysis recall statistics were calculated at different values of the R^2 parameter since this was expected to have a great influence on the results. The R^2 or goodness of fit indicates how well the model fits the data and therefore reflects the coherence within the observations. Previous studies with maSigPro indicated that a cut-off value of 0.6 would be appropriate for the selection of d.e.g.’s in time course microarray data (Conesa *et al.*, 2006). In this study, four levels of R^2, 0, 0.4, 0.6 and 0.8 were evaluated. Results are presented in Annex 7.

The *in silico* analysis revealed that the maSigFun methodology is sensitive for identifying functional classes with a high proportion of changing genes (70%) when a moderate R^2 cut-off (0.4) is imposed. At higher R^2 sensitivity drops while releasing the R^2 filter (R^2 = 0) has as a consequence that functional classes with a low proportion of regulated genes (20%-30%) could also be selected (Figure 7.2). In all cases, the rate of false positives is under control and specificity remains high (see Annex 7).

Regarding to class size, simulation study B showed that this factor is of little relevance when a sufficient level of co-expression and R^2 cut-off value are used, being the sensitivity of the method more dependent on the amount of regulated genes in the class (Figure 7.3, panels b, c and d). However, when functional classes have a lower level of co-expression and a permissive R^2 is used, maSigFun revealed a dependency on the size of the class, being the method more sensitive for classes with a large number of members (Figure 7.3.a). Again, specificity was high in all cases (see Annex 7).

Taken together, the simulation analysis showed that maSigFun is effective in identifying those functional classes for which a relative high level of gene expression coherence is present regardless the number of genes annotated to the class.

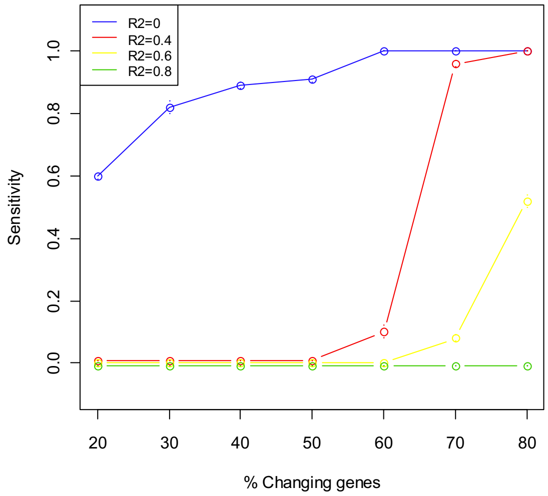


Figure 7.2: Results of simulation study A with the maSigFun method. Changes in sensitivity with the percentage of co-expressed genes in the class at four values of the goodness of fit R^2 of the regression models. Data points correspond to the mean value of 50 simulations. Confidence intervals were omitted due to their negligible size.

PCA-maSigFun

The simulation analysis for the PCA-maSigFun resulted in sensitivity and specificity values near to one in all scenarios and dataset types (see Annex 7), indicating that the method basically identifies any functional class with at least 20% of changing genes, regardless of it size, and also that the methodology is robust for the occurrence of false discoveries. This result is not surprising, since the specific property of the method is the ability of extracting gene expression sub-patterns within each class and the positive selection of the functional class happens by the identification of the correlated profile.

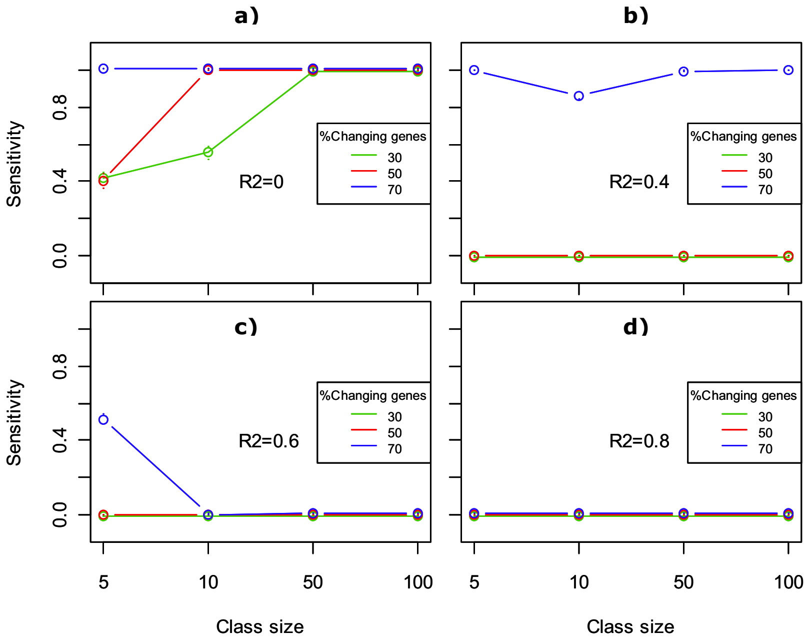


Figure 7.3: Results of simulation study B with the maSigFun method. Changes in sensitivity with the size of the class at three levels of percentage of changing genes (co-expression) in the class. One plot is provided for each level of the goodness of fit R^2 of the regression models. Data points correspond to the mean value of 50 simulations. Confidence intervals were omitted due to their negligible size.

ASCA-functional

As only one pattern of variation was modelled in each synthetic functional class, ASCA was applied with only one component in the submodel capturing time and treatment effects , denoted as "submodel b+ab" in the ASCA-genes paper (Nueda *et al.*, 2007). Genes were ranked according to the loading value of this single component, leading to one FatiScan analysis per synthetic dataset. The *in silico* study for ASCA-functional showed also interesting results. Simulation study A revealed a turning point for sensitive detection at a percentage of changing genes of 60% (Figure 7.4). This result is in agreement with the nature of the GSA strategy since the asymmetric distribution along the ordered gene list of the genes annotated to a given class is expected to occur when the percentage of genes associated with the biological phenomenon captured by the ASCA component is above half of the class size. On the other hand, simulation study B indicated that the size of the class does not affect sensitivity of detection which is merely dependent on the level of inner co-expression of the class. Full specificity was obtained for all dataset types in both studies (Annex 7).

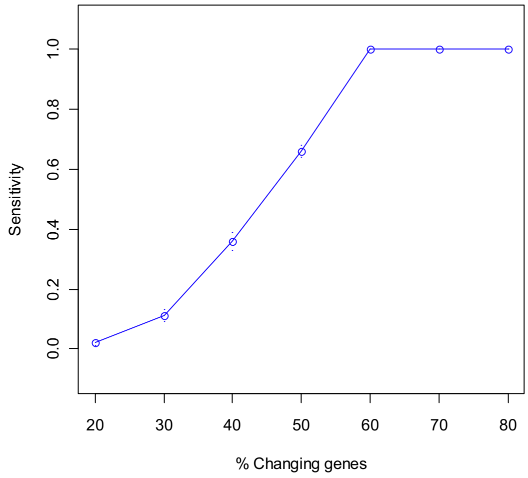


Figure 7.4: Results of simulation study A with the ASCA-functional method. Changes in sensitivity with the percentage of changing genes (co-expression) in the functional class. Data points correspond to the mean value of 50 simulations. Confidence intervals were omitted due to their negligible size.

7.3.2 Experimental datasets

The different functional assessment methods were applied to the analysis of two different experimental datasets. Since in real datasets the true differentially expressed genes are not known, recall statistics cannot be calculated. Therefore results were evaluated in terms of number of functional classes detected and biological coherence of the selection. The Gene Ontology was used as a functional classification scheme. The set of GO terms characterizing each dataset was obtained by fetching GO information from public databases, completing annotation with the Blast2GO software (Conesa *et al*., 2005), constructing the Direct Acyclic Graphs (DAGs) of each GO branch -BP, MF and CC- and obtaining all nodes in graph. This set of terms was then refined by removing annotation redundant terms. A GO term was considered annotation redundant if it has the same set of annotated genes than any of its child terms.

Toxicogenomics dataset

In this study, three increasing doses of the bromobenzene drug were tested for their toxic effects on rat liver. The original analysis of the data showed that most marked effects on the transcriptome were provoked at high BB doses and 24 hours post-administration. Also important but more moderate were the effects of medium dose and later time points. The 2665 probes contained in the rat chip were annotated to a total of 967 BP, 534 MF, and 243 CC non redundant GO terms (Table 7.1). The three analysis approaches provided semantically related results but with very different levels of specification (Supplementary material Chapter7).

			SELECTIONS		
GO category	**Original Annotation**	**Non redundant Annotations**	**maSigFun**	**PCA-maSigFun**	**ASCA-Functional**
BP	1828	967	7	33	15
MF	992	534	8	15	20
CC	398	243	0	10	8

Table 7.1: Functional analysis results for Toxicogenomics study. Number of functional terms in each of the three GO branches present in the original dataset, after removal of redundant annotations and selected after analysis with each of the proposed methods.

maSigFun analysis identified 7 BP, 8 MF and 0 CC categories (Table 7.1 and Supplementary material Chapter7) as significant at a FDR level of 0.05 and R^2 of 0.3. More restrictive values for the R^2 parameter failed to give any significant result. Functional categories included *heme oxydation*, *cell-aging*, *caspase activation via cytochrome c oxygenase*, *ferric ion binding*, *rRNA binding* and *plasminogen activator activity*, induced by BB administration, and *bile acid transporter activity*, *oxidoreductase activity*, *retinol binding* and *long-chain-fatty acid-CoA ligase activity*, repressed by high BB (Figure 7.5). Interestingly, selected categories had between 4 and 6 annotated genes and a mean inner correlation value (computed as the mean value of all pair-wise Pearson correlations of the expression profiles of the genes annotated to the class) of 0.6±0.1. This measure of class coherence is close to the critic value of 70% percentage of co-regulated genes obtained in the simulation studies for efficient selection by maSigFun.

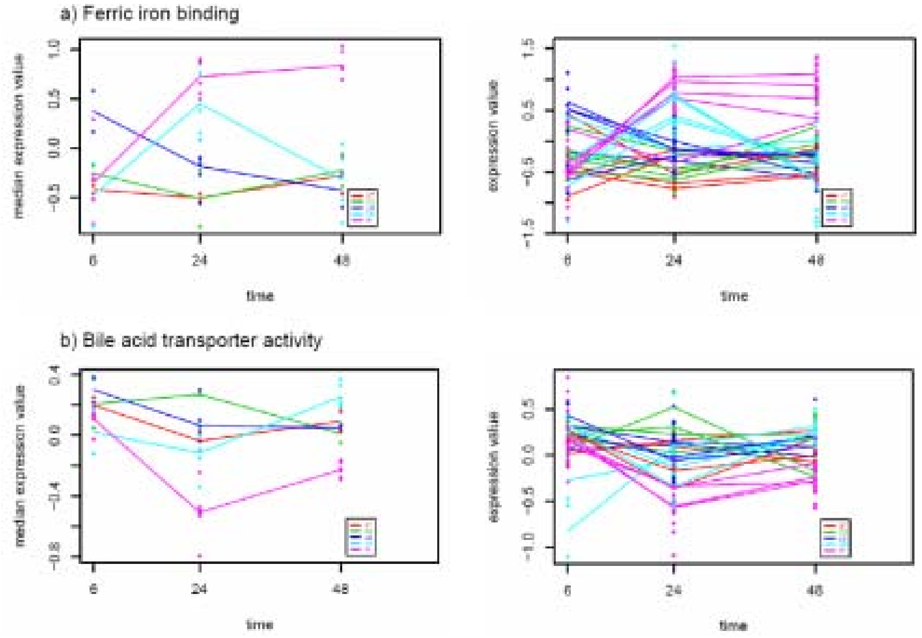


Figure 7.5: Gene expression profiles for two significant representative GO categories obtained by maSigFun analysis of the Toxicogenomics dataset. a) *Ferric iron binding* category induced by high Bromobenzene and b) *Bile acid transporter activity* category repressed by high Bromobenzene. On the left panel, the median value of the functional class is plotted while the right panel shows the expression profiles of all genes annotated in the class. Treatments are labeled by color: pink HI, light blue ME, dark blue LO, green CO, red UT.

Analysis by PCA-maSigFun provided a much richer repertoire of functional classes. GO-based PCA transformation of gene expression data compressed transcriptional information into function-associated transcriptome patterns ("synthetic genes", referred here as "GO-components"). In most cases one or two GO-components were obtained per GO term and only in very generic classes, such as *translation* or *ribosome*, up to 3 patterns of correlated behaviors were extracted. maSigPro analysis on the matrix of these new functional variables resulted in the identification of 33 BP 15 MF and 10 CC significant features (Table 7.1 and Supplementary material Chapter7). Interpretation of these results is facilitated by plotting the PCA scores of each maSigPro significant GO-component along with the PCA loading of the annotated genes. In this way we can identify the gene expression patterns captured by the significant GO-component (Figure 7.6 a) and locate the most contributing genes (Figure 7.6 b), i.e. genes that most closely follow the pattern indicated by the GO-component either with a positive (+, gene loading greater than 0), or negative correlation (-, gene loading smaller than 0).

Horizontal lines indicate the threshold for significant contribution of the gene to the GO-component pattern. The PCA-maSigFun approach identified 3 different patterns of expression: i) classes that show a peak of expression on high BB and 24 hours, ii) classes that also respond at 24 hours at medium BB and iii) classes that show a early (6 hrs) regulation for both high and medium BB (Figure 7.6). The first pattern was found for different GO terms pointing to processes as f*atty acid metabolism and oxidation* (-), *cell adhesion* (-), *amino acid metabolism* (-), *translation* (+,-) *microtubule organization* (+), *endopeptidase inhibitor activity* (-) and *vesicular fraction* (+). Functions associated with the second pattern include *translation* (+), n*egative regulation of cell proliferation* (+), *acute inflammatory response* (+,-), *xenobiotic metabolic process* (+,-), *signal transduction* (+,-), *biopolymer methylation* (-), *maintenance of localization* (+), *response to toxic compound* (+), *iron ion binding* (+,-), *exopeptidase activity* (+), *kinase activity* (+), *epoxide hydrolase activity* (+), *ribosome* (+,-). Finally, in the third pattern we found *cation homeostasis* (+), *nitric oxide mediated signal transduction* (+), *copper ion binding* (+) and *lysosome* (+). It is important to mention that, in most cases, only a subset of each GO term annotated genes showed significant contributions to the GO-component, indicating the predominant role of these genes in the determination of the pattern. In a few cases, corresponding to very general categories such as *translation* or *ribosome*, none of the annotated genes reach the threshold of significant contribution, but a continuum signal was observed, which would indicate a small but coordinated gene activity within the class. Finally, in some cases, such as *xenobiotic compound* and *acute-phase*, genes were observed that display either a positive or negative significant contribution to the component, which implies that coordination is present but with positively and negatively acting elements. For example, in the case of *acute-phase*, the *alpha-1-glycoprotein*, a positive acute phase protein, was found to have a significant contribution to the acute-phase GO-component pattern that represented gene expression activation with high BB at 24 h. Other three proteins, *alpha-1- inhibitor*, *albumin* and *tripsin*, known as negative acute–phase proteins (Al-Shahrour *et al.,* 2007), had significant but negative contributions to the GO pattern, which indicates an opposite pattern of expression (Figure 7.7). Therefore, this GO-component collects the induction of positive acute-phase proteins and the repression of negative acute-phase genes, suggesting a general activation of this cellular process.

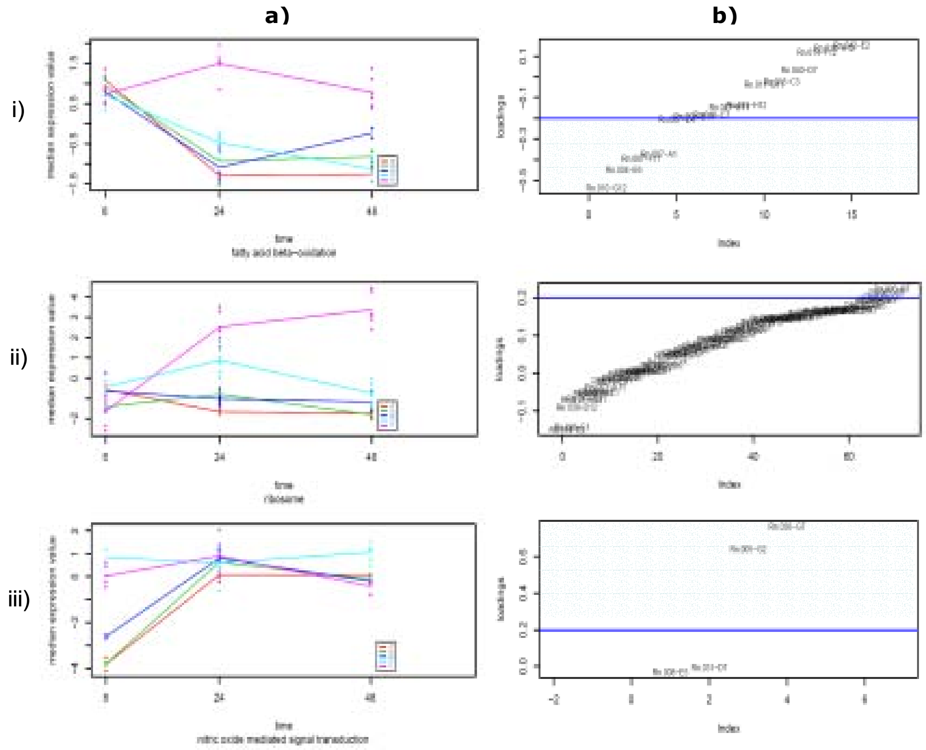


Figure 7.6: Score vs Loading analysis of PCA-maSigFun results on the Toxicogenomics dataset. a) Score profiles for three representative GO-components. b) Loading plot (gene contributions) for the same GO-components, genes labeled by their array ID. Blue lines indicate the threshold for significant contribution obtained by re-sampling (see methods).

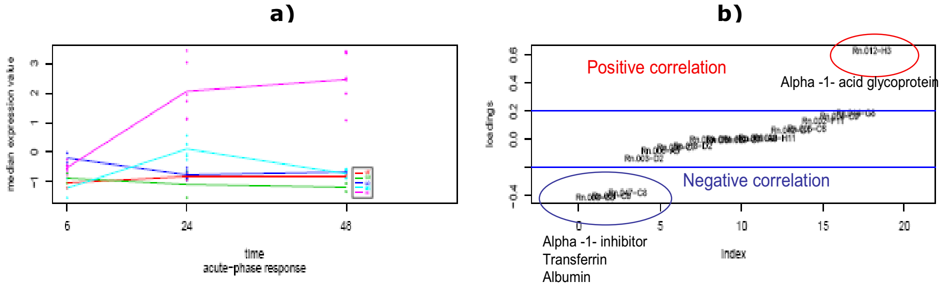


Figure 7.7: Principal variation pattern of *acute-phase response* GO category in Toxicogenomics dataset analyzed by PCA-maSigFun. a) Scores plot reveals the profile of the GO-component. b) Loadings plot show gene contributions. Threshold for significant contribution are indicated by blue line. Names of positively correlated and negatively correlated significant contributing genes are indicated.

Finally the ASCA-functional method gave an intermediate result between the two previous approaches. Analysis by ASCA indicated three main independent patterns of variation within the transcriptomics signal. As in the other approaches, the first component, which collects 46% of the gene expression variability, represents the pattern of change (induction or repression) by high BB at 24 hours (Figure 5.10). The second component, with 10% associated explained variance, represents the change of medium BB at 24 hours. The third component (9% explained variance) captures the early responses at medium and high BB. As the first principal component represents mostly the toxicological response, this was the one subjected to FatiScan that resulted in the identification of 15 BP 20 MF and 8 CC significant features (Table 7.1 and Supplementary material Chapter7). Significant processes included *ribosome*, *ferric ion binding*, *rRNA binding*, *energy* and *electron transport,* at the upper end of the gene rank, indicating that these functions are positively correlated with the pattern provided by the first ASCA-genes component of submodel b+ab, i.e, induction by high BB at 24 h. GO terms such as *retinoic metabolic process*, *fatty acid beta oxidation*, *glutamine family amino-acid metabolism*, *oxidorreductase activity* were found significantly enriched at the bottom end of the gene rank, indicating their opposite correlated pattern of change.

NSF potato stress dataset.

The Potato Stress dataset consists of three abiotic stress series (cold, heat and salt treatments) plus one control series measured along 3 time points on the NSF potato 10k chip. In general, the three different approaches behaved in a similar fashion as in the toxicogenomics dataset although a much richer functional response was observed in this study. The major gene expression pattern within this dataset corresponds to the differential behavior of the cold and salt stresses with respect to the control and heat conditions. A differential regulation is observed between the two pairs of series already at 3 hours, peaking at 9 hours and maintained till the end of the experiment (Figure 7.8).

The number of functional classes obtained with each of the methods is shown in Table 7.2. and a complete list of all significant GO terms is provided in Supplementary material Chapter7. maSigFun analysis gave the smallest amount of significant GO terms, which had on average 6.4 annotated terms and a mean inner correlation value of 0.63±0.1. Significant functions corresponded to profiles of induction (+) or repression (-) of the class as a whole for the cold and salt stressors with respect to the control and heat conditions. Down-regulated processes included photosynthesis-related terms, *fructose metabolism*, *cell-wall modification*, *lateral root morphogenesis* and *reductive pentose-phosphate cycle*. Up-regulated processed referred to protein turnover, r*esponse to hypoxia* and *glucose stimulus*, *multi-drug transport*, *salicylic acid signaling pathway* and diverse enzymatic activities. PCA-maSigFun gave again a much richer view on cellular processed (447 selected GO terms) and highlighted additional functions such as *response to stress, chitinase activity, oxidoreductase activity*, *transmembrane transport*, *secretory pathway*, j*asmonic acid signaling* and *abscisic acid pathways*, among many others.

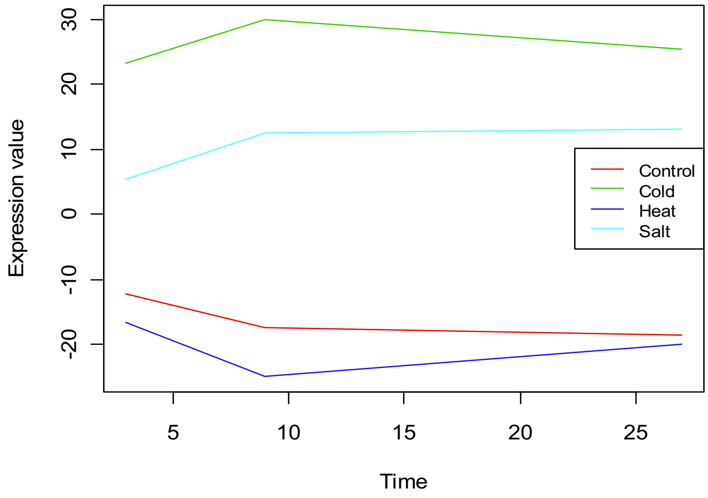


Figure 7.8: Principal variation pattern in the Potato Stress dataset. The pattern is captured by the first component of submodel b+ab (treatment + timextreatment) of ASCA-functional analysis. The plot shows the score values of this first component.

Finally ASCA-functional analysis indicated the major pattern of variability as the difference between the cold and salt stresses on one hand and heat and control conditions on the other, comprising this pattern 57% of the variability contained in the dataset (Figure 7.8). FatiScan analysis on the gene loadings rank provided by this first component indicated as significant most of the processes revealed by the other methods, i.e., response to several stimuli, *protein synthesis and degradation*, diverse hormone signaling pathways, *lignin biosynthesis* associated with genes in upper rank positions; *photosynthesis*, *microtubule-based movement*, *RNA binding* and *lypoxigenase activity* as processed over-represented in bottom rank genes. Taken together, the results of the three different approaches reveal, at different levels of detail, the cellular response triggered by the treatments. While the heat stress does not seem to provoke, at least in this experiment, a large response, cold and salt treatment produced similar patterns of transcriptome regulation. Hormone signaling cascades, response to stress markers, lignin biosynthesis, oxidoreductase activity and protein metabolism were induced processes while the whole photosynthetic machinery seems to be halted by these abiotic stress agents.

			SELECTIONS		
GO category	**Original Annotation**	**Non redundant Annotations**	**maSigFun**	**PCA-maSigFun**	**ASCA-Functional**
BP	2444	780	23	258	116
MF	943	431	21	141	29
CC	369	203	14	48	46

Table 7.2: Functional analysis results for Potato Stress study. Number of functional terms in each of the three GO branches present in the original dataset, after removal of redundant annotations and selected after analysis with each of the proposed methods.

7.4 Discussion

The understanding of the cellular and functional implications of global gene expression changes measured through microarrays is in many cases the ultimate and most important goal of the biological experiments analyzed by this technology. When the experiment includes a time component data has dynamic nature that needs to be incorporated in the functional analysis. The statistical approaches presented and evaluated in this study try to exploit this dynamic property from different perspectives and offer methods that explicitly focus on coordinative behaviors within the cellular functionality along the time span. This is in contrast to more traditional approaches that require a gene selection method and a partitioning algorithm before reaching the stage of functional assessment. maSigFun is, from the three algorithms proposed, the method that more strongly concentrates in co-expression. By fitting one regression model on the expression data gathered by each functional class, it follows that class members need to be highly correlated. Conceptually, maSigFun could be related to the *globaltest* developed by Goeman *et al*. (2004) where one statistical model is fit for a gene set, although the two methodologies have very different realizations. While the *globaltest* treats genes in the set as the dependent variables of the model, maSigFun regresses on experimental factors (time and treatment) and considers individual genes as observations of the values that time and treatment take for the functional class. The simulation studies indicated that only classes with a high proportion of coordinately changing genes (~ 70%) were readily detected by this method. The experimental datasets confirmed this tendency and also showed a bias in class selection for those with a reduced number of annotated genes and a relatively high (~60%) inner correlation. This is not surprising since large -and frequently more general- functional classes are more likely to include different regulation patterns and also to capture more noise. The consequence is that this method is able to reveal specific cellular functionalities which are affected by the experimental conditions but may escape to other interesting phenomena which are not so well defined by a one-block behavior of the functional class. This, which might be sufficient in some cases, may imply a partial result in some others where a broader view of the transcriptional changes is sought. In the case of the toxicogenomics dataset maSigFun analysis provided a clearly limited result. Although some detected functions such as *heme oxygenase activity* and *bile acid transporter activity* are key makers of the toxicological response (Heijne *et al*., 2003), many other important processes such as the *xenobiotic metabolic process, acute-phase response* and *epoxide hydrolase activity* did not show up in this analysis. In the case of the abiotic stress study, however, maSigFun analysis did provide already a quite extensive functional view of the regulated processes, possibly due to the involvement of numerous specific enzymatic activities and cellular locations with low number of annotated genes, and the more extensive transcriptional profiling (~ 10k probes) of the potato dataset.

The above mentioned aspect of the broader evaluation of the transcriptional response from a functional point of view is probably best addressed by the PCA-maSigFun method. In this strategy sub-patterns of time-associated changes within each functional class are identified by PCA analysis followed by regression modeling on the principal components. PCA-maSigFun provided the largest GO term selection in both experimental datasets and the simulated study indicated that the method is able to identify any functional group in which some correlation structure is present. The method should not be considered as an enrichment analysis strategy, but more a methodology to dissect and investigate how genes, functions and co-expression relate. This exercise can be very interesting in some cases such as in the acute-phase example shown in the toxicogenomics section. Here, PCA-maSigFun clearly showed the correlation and anti-correlation relationships between acute-phase positive and negative genes, which would presumably result in an activation of the process. Any method that would concentrate only in shared profiles would fail to identify this class in which co-regulation is clearly present. Possibly recently introduced term relationships in the Gene Ontology (r*egulates_positively* and *regulates_negatively*) would help to more formally consider these situations (see http://www.geneontology.org/GO.process.guidelines.shtml#reg) but, to our knowledge, there are not yet functional assessment methods that incorporate these relationship descriptors. It is also important to indicate, that although PCA-maSigFun is not an enrichment method, it does not return just any functional class. First, PCA assures that selected categories must contain a structure of correlation above the level of noisy variance of each particular dataset and secondly, the maSigPro analysis on the selected components imposes that these patterns can be fitted to a time-dependent model. In fact, in most of the selected functional terms the significant profile corresponded to the first component of the PCA analysis of the class (data not shown). This implies that the major function-dependent patterns of variation also corresponded to time-related events and consequently are consistent with the biological scenario investigated by the time-course experiment. A possible draw-back of this method is the large size of the resulting selections. This means that browsing of the analysis results could be time consuming and that some too general-low informative classes may "artificially" enlarge the output. We partially solved this problem by including only non-annotation redundant GO terms in the analysis (a GO term is considered annotation redundant if it has the same set of annotated genes as any of its child terms). Other options would be to filter results according to the GO structure (by level, by branch most specific term, etc) or to group significant functional patterns by some clustering method. The last option has been implemented in the PCA-maSigFun method and is included in the standard output.

An intermediate result between the restricted view of maSigFun and the profusion of classes given by PCA-maSigFun is obtained by ASCA-functional. In contrast to the two previous methods, this strategy does not imply a transformation from a gene profile to a class profile, but simply ranks genes according to a pattern of variation and assessing a functional enrichment along this rank. This pattern of variation is provided by the ASCA-genes model and, although in this work this is related to time series analysis, the method is generally applicable when more than two conditions are present in the study. In this sense ASCA-functional can be considered as an extension of GSA to multi-class and time series data. Other adaptations of the GSA methodology propose the employment of diverse statistics such as linear modeling and/or posterior probability to measure the association of the gene expression with the phenotype (Jiang and Gentleman, 2007), but to our knowledge no statistics have yet been suggested to consider dynamic data. The simulation study indicated that our strategy can identify classes from an inner co-expression level of 50% - 60%, which is indeed in between the other two methodologies presented. ASCA-functional does not provide a detailed analysis of co-expression as in PCA-maSigPro, but it does very naturally show the relationship between functional classes: as the rank provided by the gene loadings in the principal components of the ASCA submodels is a measure of how well each gene follows the pattern identified as major time-dependent expression trends, functional classes overrepresented in the upper part of the rank will follow this pattern while enriched terms at bottom positions will have the opposite profile. Another particularity of this method is that it only reaches major expression trends, since the PCA models simplify data by their predominant structures. We argue that this, which could be suggested as a limitation for a gene-centric analysis, is of little relevance when considering functional blocks with coordinated behaviors. Recently, Chen *et al.* (2008) proposed a methodology for gene set enrichment analysis based on PCA. However, their approach is very different to ours since the authors use PCA to select gene sets whose one-component projection best associates to the phenotype, rather than to quantify the relationship of individual gene profiles to a defined generic pattern.

We can conclude that the methodologies presented in this paper are valuable and offer different approaches to study microarray time series data from a functional perspective. The methods should not be considered as competitive but providing different insights on the molecular and functional events taken place within the biological system under study.

Chapter 8

General conclusion

This thesis is dedicated to the development and application of new statistical methods for the analysis of multiple series of "Time Course Microarray" (TCM) by considering the specific problematic of this type of data. The work was started by providing a general overview of microarray technology and a review of statistical general methods used and also the specific ones for TCM. Furthermore, the main limitations of the application of general tools to time series microarray data were identified using a practical study on experimental data. Next, the methodologies developed in this thesis, maSigPro, ASCA-genes, ASCA as pre-processing technique and a group of tools to deal with functional categories: maSigFun, PCA-maSigFun and ASCA-Functional are presented. These new techniques were sistematically applied on simulated and real datasets to analyze their performance and comparison with other existing methodologies.

Here the main conclusions of the thesis are summarized, organized according to the objectives presented at the beginning of the document:

a) **Study of the state-of-the-art transcriptomic analysis methodologies applied to TCM.**

- The analysis of TCM data by clustering approaches poses limitations to the interpretation of partioning results and time differences across multiple series are difficult to extract from clustered profiles.

- The aplication of classical inferential tools to TCM are not adequate as they do not analyze the dynamics of the data.

- The majority of the specific methodologies for TCM, both in clustering and in gene selection approaches, were developed for long series, such as is the case of cell cycle and long developmental studies.

- There was a need to develop methodologies to visualise and analyse short MSTC due to the increasing use of microarrays to explore the gene expression response to different stimuli by using short series.

b) **Development of new statistical methods to deal with TCM focusing on short, independent and multiple series time course (MSTC)**

- **maSigPro** is a powerful statistical procedure to identify genes that have different expression profiles among experimental groups in TCM experiments.

- **maSigPro** detects significant profiles differences without carrying out tedious multiple pair-wise comparisons, allowing for unbalanced designs and heterogeneous sampling times.

- The variable definition of the **maSigPro** models allows us not only finding genes with temporal expression changes between experimental groups, but also to analyze the magnitude of these differences.

- The application of ASCA to TCM data, **ASCA-genes**, helps to understand the shared behaviours of gene expression under the studied factors (time, experimental group and interaction) and allows the identification of the genes that follow the discovered patterns.

- **ASCA** can be used as **pre-processing** technique to remove systematic noise from microarray datasets. The ANOVA decomposition and analysis of covariance present in ASCA provide the means to remove the noise present in the gene expression signal and the signal present in the arrays noise.

- TCM lacks of specific methodologies that integrate biological knowledge in statistical analysis and exploit the dynamics of functional –rather than gene expression- changes.

- The developed methodologies for the functional assessment of time course, **maSigFun**, **PCA-maSigFun** and **ASCA-functional**, are able to capture different aspects of the relationship between genes, functions and co-expression that are biologically meaningful.

- The availability of the developed methodologies in this thesis as R packages makes these analysis approaches easily accessible to the research community.

c) Study of the effectiveness of the developed methods by comparing

their performance with other available techniques.

- Comparisons between different analysis methodologies are difficult to carry out and they are not always fair. Different treatment of the data or different criteria to establish cut-offs for gene selection can make this task complicated and prone to arbitrary considerations.

- Simulations are very useful to investigate how to use a statistical methodology and to understand its working. However, performance results obtained with simulation studies must be considered as orientative due to synthetic data is a necessary simplification of true microarray data.

- **maSigPro** and the combination **ASCA-filter** and **maSigPro** has proven to provide good gene-selection results over a wide variety of data analysis scenarios and to outperform other methodologies for the analysis of TCM. Similarly, our methodologies developed for functional assessment of TCM provide quality and exhaustive functional results, difficult to obtain by other available methods.

- The rapid evolution of genomics research poses continuous challenges to the bioinformatics and statistics disciplines which need to be dynamic in delivering new analysis methodologies that are able of processing datasets of increasing complexity and size.

●●●●●●●●

To summarize, this thesis offers four useful techniques for the analysis of Time Course Microarray data and a review of the state-of-the-art of the developed methodologies for transcriptomic data. Our methods, as any other, do not intent to model gene expression perfectly, but provide us with useful tools for studying and understanding the biology. Pharaphrasing the statistician George Box: "All models are wrong, but some are useful".

Apart from all this technical conclusions, the developed work has offer to me the opportunity of involving in the research world developing (I think) important personal skills. Critical view of the proposed problems and of the works developed for researchers is one of them. However, on my point of view, the most important thing the bioinformatics field has shown me is the need of collaborating with the scientific community to the advance of the science.

Future lines

This thesis opens interesting research lines. In the following we mention several options that arise throughout the different chapters of this work. In each line of research we indicate the specific chapter which it is related to.

- The incorporation of statistical tools to maSigPro to explicitly consider long, heteroscedastic and longitudinal data in an accurate way (Chapter 4). maSigPro applies classical least-squares technique to estimate the coefficients of the regression model and these estimates are optimum if the assumptions of homoscedasticity and independence of the observations are satisfied. Since microarray data might not always meet these requirements, it is possible that the obtained coefficients are not the best ones. The study of incorporating in our model generalized least squares, weighted least squares and splines regression to our model could be an option.

- Similarly, ASCA-genes has been also developed for independent and homoscedastic data (Chapter5). The study of the application of ANOVA models for longitudinal and heteroscedastic data within the ASCA model could be another option.

- The search of an independent criterion of the signal-to-noise ratio for gene selection in ASCA-genes methodology (Chapter 5). The gene selection method implemented in ASCA-genes was developed by testing the strategy in experiments with high level of signal-to-noise ratio. Posterior studies revealed that this criterion is not robust with experiments with a large number of genes and low signal-to-noise ratio. The application of other thresholds statistics and strategies for feature selection is a line of research we are already studying.

- Multivariate statistics, while powerful in extracting knowledge in large datasets, can also render difficult to interpret when applied to omics data of low signal-to-noise ratios. Recent tendencies try to use simple models that describe the effect of small groups of variables, in our case genes. It would be a good extension of ASCA-genes to apply this approach to improve interpretation.

- Selection of components to apply the ASCA-filter strategy (Chapter 6). In spite of the good results obtained and the ingenious theoretical strategy proposed with ASCA-filter to remove the noise of the signal and the signal of the noise, we admit that the method is not easy to apply. The main difficulty comes from the selection of the number of principal components. It requires a good comprehension of the variation of the data to analyse and the results are very sensitive to the number of components chosen. Furthermore, as the variation removed depends on these components, which is a discrete parameter, we can not choose a quantity of variation in a continuous way being the different solutions associated with the different selections of components also not continuous selections of genes.

- The development of new tools for functional categories (Chapter 7). The application of several combinations of tools to groups of genes has shown to be an efficient way to asses the functional aspects of time course transcriptomics data. However, more possibilities can be studied as ASCA-maSigFun (instead of PCA-maSigFun), that could apply maSigPro to the patterns discovered with ASCA (instead of the patterns discovered with PCA). This alternative would help to avoid the structural noise present in the group of genes and would focus on the time patterns of variation.

- In simulation studies we have always generated different patterns of variation. We think that it would be interesting to simulate combinations of them to analyse how our methods, especially ASCA-genes, are able to capture these behaviours.

- The application of Multiway techniques to omics data. This thesis work concentrates on transcriptomics, i.e. one data matrix is subjected to analysis. However to obtain biological knowledge in System Biology, multiple omics technologies should be combined resulting in multiple data structures. In this line Conesa *et al*. (2008) developed a strategy that uses Partial Least Squares (PLS) to identify correlated gene function features with physiological variables. Multi-way methods are able to take into account the different levels of data organization and analyze the underlying components of variability that affect different types of biological variables. They seem to us as a very attractive tool for explorative and variable selection analysis of Systems Biology data.

Annexes

Annex 1: maSigPro package scheme

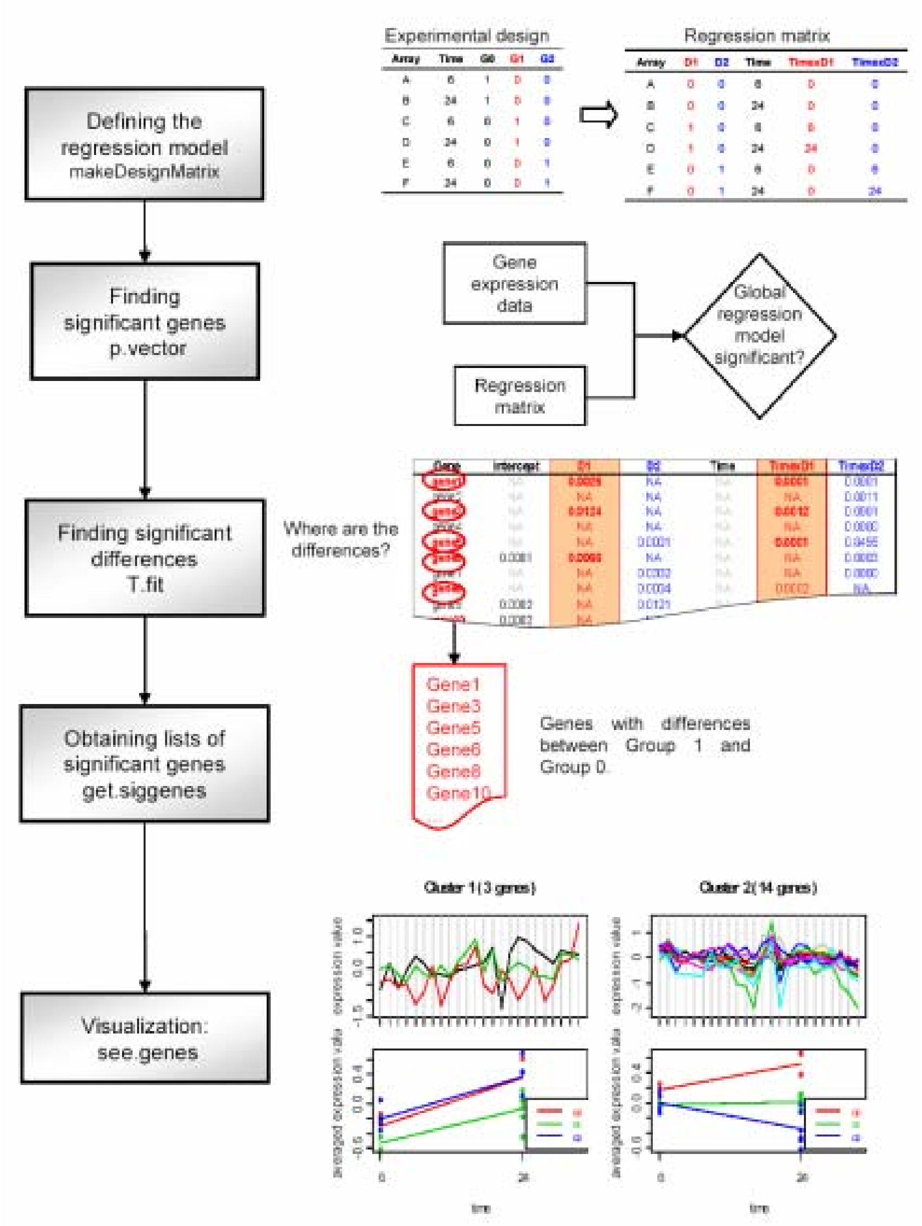

Defining the regression model makeDesignMatrix
Experimental design
Array Time G0 G1 G2
Regression matrix
Array D1 D2 Time TimexD1 TimexD2
Finding significant genes p.vector
Gene expression data
Regression matrix
Global regression model significant?
Finding significant differences T.fit
Where are the differences?
Gene intercept D1 D2 Time TimexD1 TimexD2
Obtaining lists of significant genes get.siggenes
Gene1
Gene3
Gene5
Gene6
Gene8
Gene10
Genes with differences between Group 1 and Group 0.
Visualization: see.genes
Cluster 1 (3 genes)
Cluster 2 (14 genes)
expression value
averaged expression value
time

Figure A-1: **maSigPro scheme.** Grey boxes include the name of main functions of the maSigPro package in order of intervention: *makeDesignMatrix, p.vector, T.fit, get.siggenes* and *see.genes*.

The scheme showed in Figure A-1 summarizes the performance of maSigPro package. The scheme is illustrated with some pictures of an example with only two time-points and two groups to clarify the results.

The maSigPro package can be obtained from the Bioconductor repository or downloaded from the personal webs of the authors: www.ua.es/persona/mj.nueda and http://bioinfo.cipf.es/aconesa. Load maSigPro by typing at the R prompt:

```
>library(maSigPro)
```

The maSigPro vignette will be added to the Vignettes menu of R and it can be downloaded from the personal webs of the authors.

The analysis approach implemented in maSigPro is executed in 5 major steps which are run by the package core functions make.design.matrix(), p.vector(), T.fit(), get.siggenes() and see.genes(). Additionaly, the package provides the wrapping function maSigPro which executes the entire analysis in one go.

The maSigPro vignette explains the usage of each of these functions using as example a dataset from a multiple series time course experiment. At the end of the document there is also explained how to apply maSigPro to other experimental designs.

Annex 2: Supplementary figures of Chapter 5

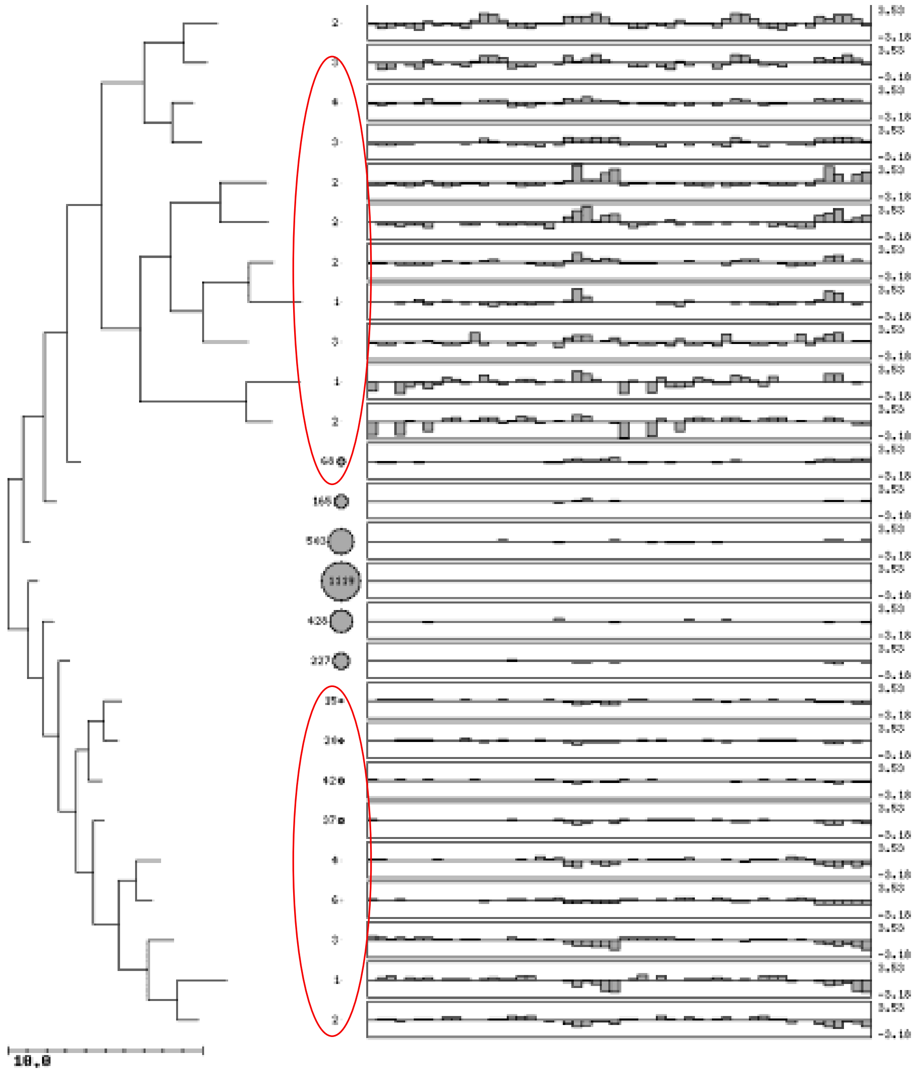
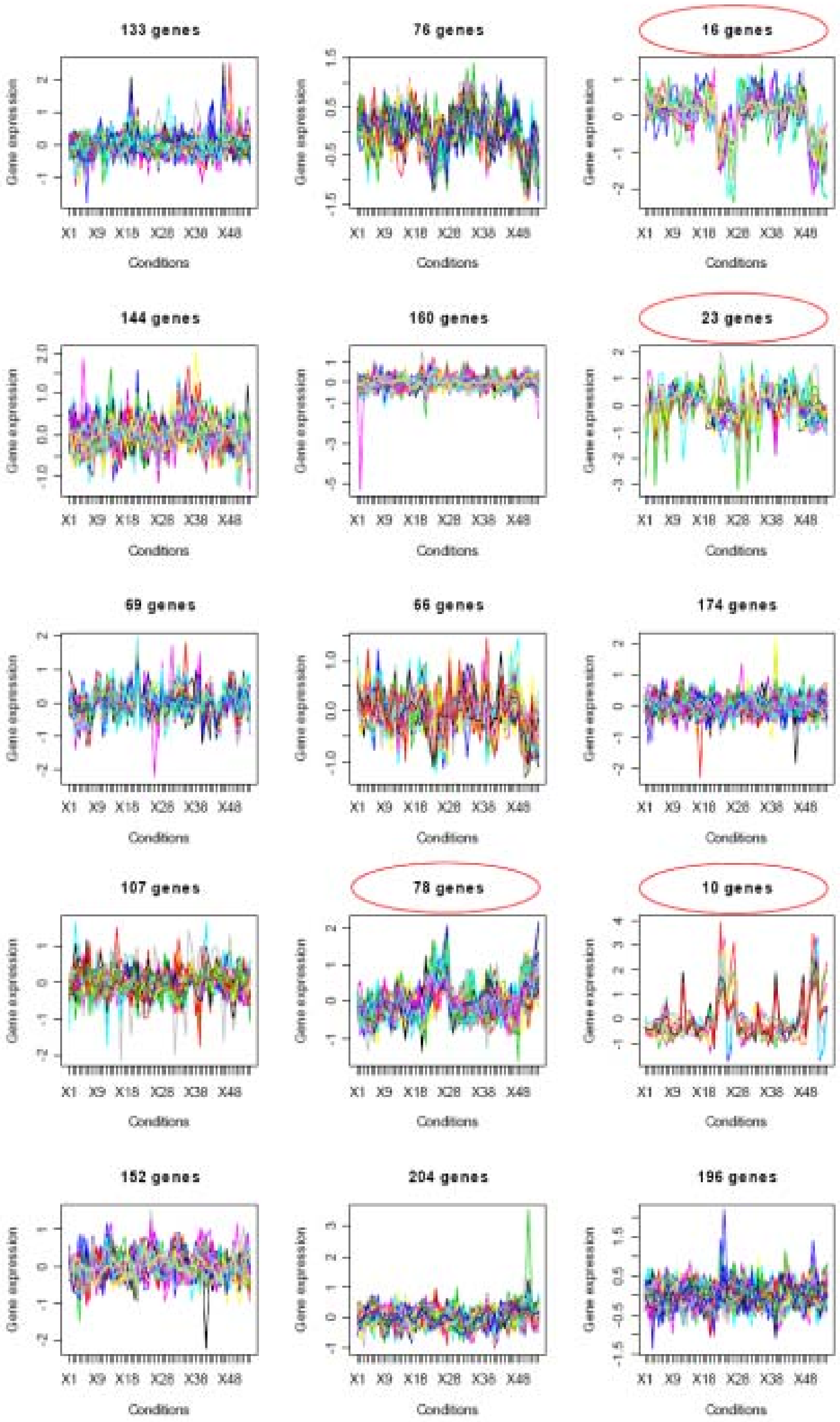


Figure A-2: SOTA cluster analysis. Clusters where changes in gene expression profiles were evident (red circled) were selected for further analysis.


133 genes
76 genes
16 genes
144 genes
160 genes
23 genes
69 genes
66 genes
174 genes
107 genes
78 genes
10 genes
152 genes
204 genes
196 genes
Gene expression
X1 X9 X18 X28 X38 X48
Conditions

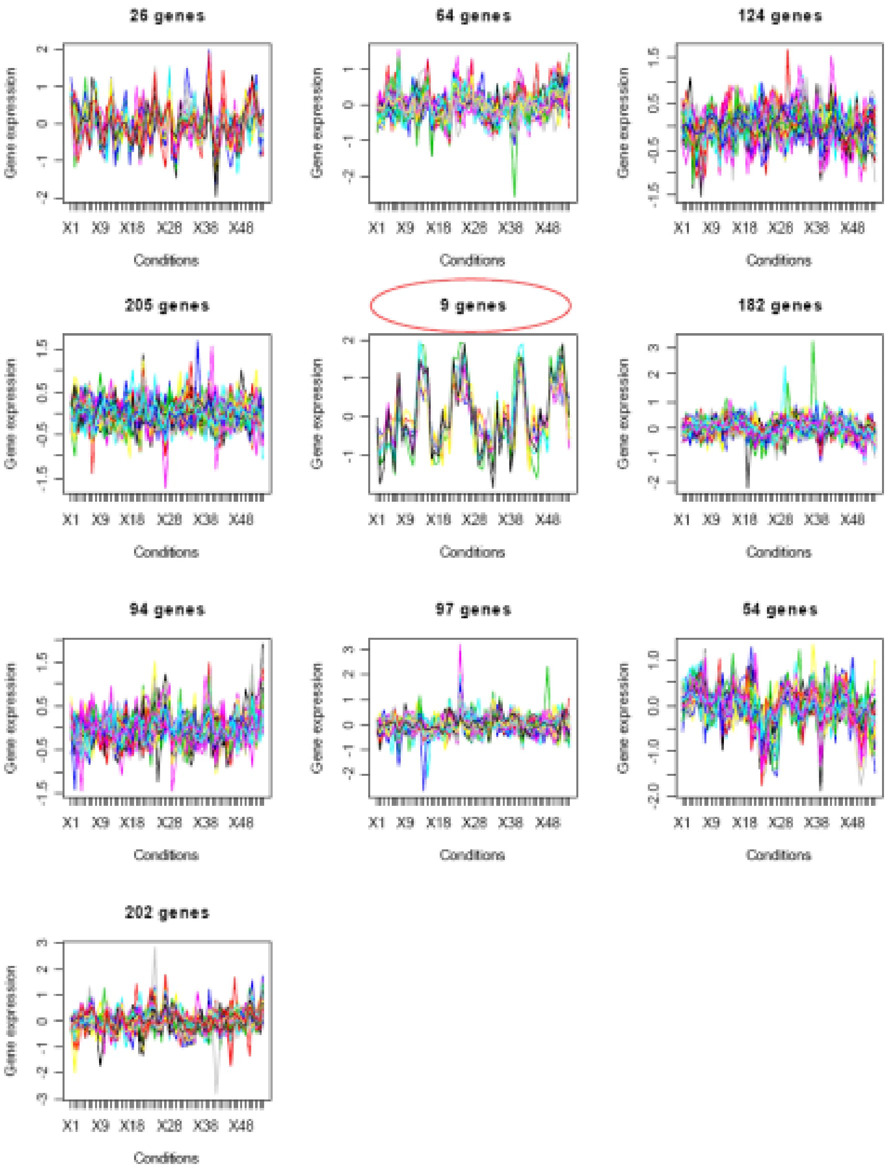


Figure A-3: K-means cluster analysis. Clusters where changes in gene expression profiles were evident (red circled) were selected for further analysis.

Annex 3: ASCA-genes on simulated datasets with high number of genes

In this annex we show gene-selection obtained with ASCA-genes when the experiment has a high number of genes. The study has been carried out with the simulation study 1 of Chapter 6, i.e. an independent time-course experiment with 3 time-points, 3 experimental groups and 410 changing genes of a total of 10000 genes. 6 different scenarios were simulated with different type and amount of noise (see Chapter 6 for details).

Table A-1 shows that in scenarios without noise or with low noise the obtained selection is adequate. However when high structural noise is added to the data, the solution is not as adequate. In Chapter 5 we obtained a good solution by applying ASCA-genes to a scenario with high structural noise. However in such case we simulated a dataset with 2600 genes to resemble the structure of the bromobenzene dataset. By considering a more real experiment with 10000 genes, the leverage limit is affected by the high proportion of flat profiles providing a considerable number of false positives (Figure A-5). On the other hand, leverage limits obtained in scenarios without noise or with small number of genes provide a good solution (see Figure A-4 and Figure 5.6 of Chapter 5).

SCENARIO	SELECTION	FP	FN	SENSIT	SPECIF
1	410	0	0	1.000	1.000
2	413	3	0	1.000	1.000
3	434	24	0	1.000	0.997
4	450	40	0	1.000	0.996
5	649	239	0	1.000	0.975
6	614	204	0	1.000	0.979

Table A-1: ASCA-genes results.

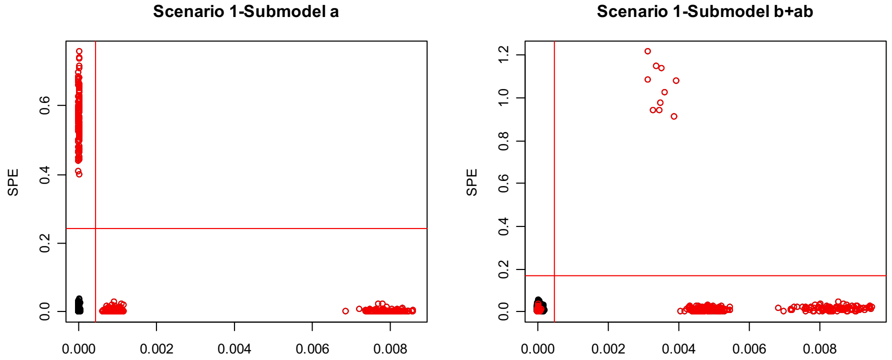


Figure A-4: Leverage and SPE of "submodels a" and "b+ab" for scenario 1. Black colours represent flat genes and red colours non flat genes.

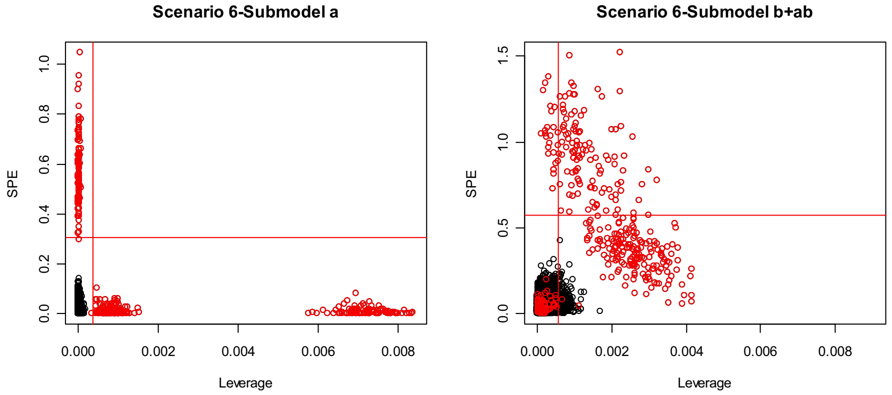


Figure A-5: Leverage and SPE of "submodels a" and "b+ab" for scenario 6. Black colours represent flat genes and red colours non flat genes.

We have tried other different criteria to select genes with ASCA approach by using scaled leverage and SPE to compute the distances of each gene to the origin. The use of scaled leverage and SPE allows that both measures have the same importance. By ordering these distances we obtain a ranking of genes in each submodel. Graphically we can determine a level to decide the important genes (see case of scenario 6 in Figure A-6, the remaining scenarios not shown). This level has been choosen looking for a hollow that indicates change of the distances to the origin. This option offers the correct solution in all the scenarios (410 genes). We have also designed a non graphical criterion assigning to each gene the largest distance of each submodel to get a unique measure for each gene. By choosing the first 410 genes of this distance we obtain also the correct solution. However, although these selections are correct the criteria do not offer a statistic significance level. Currently, we are studying new ways to obtain an adequate statistic gene selection criterion. This is a future line of research to be developed.

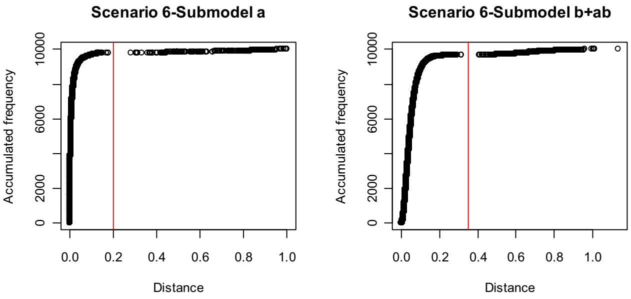


Figure A-6: Distances from leverage-SPE scaled points to the origin in Scenario 6.

Annex 4: Supplementary tables of maSigPro applied to the simulation study 1 of Chapter 6

SCENARIO 1						
DATA	RSQ	SELECTION	FP	FN	SENSIT	SPECIF
Original	0	430	20	0	1.000	0.998
	0.6	414	4	0	1.000	1.000
	0.7	410	0	0	1.000	1.000
	0.8	410	0	0	1.000	1.000
Filtered 1,2,0	**0**	**411**	**1**	**0**	**1.000**	**1.000**
	0.6	**411**	**1**	**0**	**1.000**	**1.000**
	0.7	**410**	**0**	**0**	**1.000**	**1.000**
	0.8	**410**	**0**	**0**	**1.000**	**1.000**
Filtered 1,2,1	0	411	1	0	1.000	1.000
	0.6	411	1	0	1.000	1.000
	0.7	410	0	0	1.000	1.000
	0.8	410	0	0	1.000	1.000
Filtered 1,2,3	0	412	2	0	1.000	1.000
	0.6	412	2	0	1.000	1.000
	0.7	410	0	0	1.000	1.000
	0.8	410	0	0	1.000	1.000
Filtered 1,2,5	0	417	7	0	1.000	0.999
	0.6	412	2	0	1.000	1.000
	0.7	410	0	0	1.000	1.000
	0.8	410	0	0	1.000	1.000
Filtered 1,3,0	0	411	1	0	1.000	1.000
	0.6	410	0	0	1.000	1.000
	0.7	410	0	0	1.000	1.000
	0.8	410	0	0	1.000	1.000
Filtered 1,3,1	0	411	1	0	1.000	1.000
	0.6	411	1	0	1.000	1.000
	0.7	410	0	0	1.000	1.000
	0.8	410	0	0	1.000	1.000
Filtered 1,3,3	0	413	3	0	1.000	1.000
	0.6	411	1	0	1.000	1.000
	0.7	410	0	0	1.000	1.000
	0.8	410	0	0	1.000	1.000
Filtered 1,3,5	0	419	9	0	1.000	0.999
	0.6	411	1	0	1.000	1.000
	0.7	410	0	0	1.000	1.000
	0.8	410	0	0	1.000	1.000

Table A-2: maSigPro results for scenario 1 with original and filtered data with different strategies.

SCENARIO 2						
DATA	**RSQ**	**SELECTION**	**FP**	**FN**	**SENSIT**	**SPECIF**
Original	0	427	17	0	1.000	0.998
	0.6	412	2	0	1.000	1.000
	0.7	410	0	0	1.000	1.000
	0.8	410	0	0	1.000	1.000
Filtered 1,2,0	0	410	0	0	1.000	1.000
	0.6	410	0	0	1.000	1.000
	0.7	409	0	1	0.998	1.000
	0.8	391	0	19	0.954	1.000
Filtered 1,2,1	0	410	0	0	1.000	1.000
	0.6	410	0	0	1.000	1.000
	0.7	409	0	1	0.998	1.000
	0.8	392	0	18	0.956	1.000
Filtered 1,2,3	0	413	3	0	1.000	1.000
	0.6	410	0	0	1.000	1.000
	0.7	410	0	0	1.000	1.000
	0.8	395	0	15	0.963	1.000
Filtered 1,2,5	0	414	4	0	1.000	1.000
	0.6	411	1	0	1.000	1.000
	0.7	410	0	0	1.000	1.000
	0.8	405	0	5	0.988	1.000
Filtered 1,3,0	**0**	**410**	**0**	**0**	**1.000**	**1.000**
	0.6	**410**	**0**	**0**	**1.000**	**1.000**
	0.7	**410**	**0**	**0**	**1.000**	**1.000**
	0.8	**399**	**0**	**11**	**0.973**	**1.000**
Filtered 1,3,1	0	411	1	0	1.000	1.000
	0.6	410	0	0	1.000	1.000
	0.7	410	0	0	1.000	1.000
	0.8	400	0	10	0.976	1.000
Filtered 1,3,3	0	415	5	0	1.000	0.999
	0.6	411	1	0	1.000	1.000
	0.7	410	0	0	1.000	1.000
	0.8	404	0	6	0.985	1.000
Filtered 1,3,5	0	419	9	0	1.000	0.999
	0.6	413	3	0	1.000	1.000
	0.7	410	0	0	1.000	1.000
	0.8	408	0	2	0.995	1.000

Table A-3: maSigPro results for scenario 2 with original and filtered data with different strategies.

SCENARIO 3						
DATA	**RSQ**	**SELECTION**	**FP**	**FN**	**SENSIT**	**SPECIF**
Original	0	413	3	0	1.000	1.000
	0.6	411	1	0	1.000	1.000
	0.7	410	0	0	1.000	1.000
	0.8	369	0	41	0.900	1.000
Filtered 1,2,0	0	410	2	2	0.995	1.000
	0.6	392	1	19	0.954	1.000
	0.7	370	0	40	0.902	1.000
	0.8	315	0	95	0.768	1.000
Filtered 1,2,1	**0**	**823**	**413**	**0**	**1.000**	**0.957**
	0.6	**489**	**79**	**0**	**1.000**	**0.992**
	0.7	**430**	**21**	**1**	**0.998**	**0.998**
	0.8	**409**	**1**	**2**	**0.995**	**1.000**
Filtered 1,2,3	0	1733	1323	0	1.000	0.861
	0.6	662	252	0	1.000	0.974
	0.7	489	79	0	1.000	0.992
	0.8	415	5	0	1.000	0.999
Filtered 1,2,5	0	2126	1716	0	1.000	0.820
	0.6	754	344	0	1.000	0.964
	0.7	525	115	0	1.000	0.988
	0.8	422	12	0	1.000	0.999
Filtered 1,3,0	0	411	3	2	0.995	1.000
	0.6	404	1	7	0.983	1.000
	0.7	382	0	28	0.932	1.000
	0.8	322	0	88	0.785	1.000
Filtered 1,3,1	0	1183	773	0	1.000	0.919
	0.6	599	189	0	1.000	0.980
	0.7	454	45	1	0.998	0.995
	0.8	411	2	1	0.998	1.000
Filtered 1,3,3	0	2643	2233	0	1.000	0.765
	0.6	911	501	0	1.000	0.947
	0.7	563	153	0	1.000	0.984
	0.8	426	16	0	1.000	0.998
Filtered 1,3,5	0	3047	2637	0	1.000	0.723
	0.6	1091	681	0	1.000	0.928
	0.7	626	216	0	1.000	0.977
	0.8	439	29	0	1.000	0.997

Table A-4: maSigPro results for scenario 3 with original and filtered data with different strategies.

SCENARIO 4						
DATA	**RSQ**	**SELECTION**	**FP**	**FN**	**SENSIT**	**SPECIF**
Original	0	411	1	0	1.000	1.000
	0.6	410	0	0	1.000	1.000
	0.7	387	0	23	0.944	1.000
	0.8	281	0	129	0.685	1.000
Filtered 1,2,0	0	395	0	15	0.963	1.000
	0.6	348	0	62	0.849	1.000
	0.7	325	0	85	0.793	1.000
	0.8	264	0	146	0.644	1.000
Filtered 1,2,1	0	429	19	0	1.000	0.998
	0.6	412	2	0	1.000	1.000
	0.7	400	0	10	0.976	1.000
	0.8	363	0	47	0.885	1.000
Filtered 1,2,3	0	457	47	0	1.000	0.995
	0.6	419	9	0	1.000	0.999
	0.7	410	1	1	0.998	1.000
	0.8	377	0	33	0.920	1.000
Filtered 1,2,5	0	495	85	0	1.000	0.991
	0.6	425	15	0	1.000	0.998
	0.7	411	1	0	1.000	1.000
	0.8	389	1	22	0.946	1.000
Filtered 1,3,0	0	404	0	6	0.985	1.000
	0.6	389	0	21	0.949	1.000
	0.7	339	0	71	0.827	1.000
	0.8	268	0	142	0.654	1.000
Filtered 1,3,1	**0**	**450**	**40**	**0**	**1.000**	**0.996**
	0.6	**414**	**5**	**1**	**0.998**	**0.999**
	0.7	**405**	**1**	**6**	**0.985**	**1.000**
	0.8	**379**	**0**	**31**	**0.924**	**1.000**
Filtered 1,3,3	0	505	95	0	1.000	0.990
	0.6	431	21	0	1.000	0.998
	0.7	413	3	0	1.000	1.000
	0.8	393	0	17	0.959	1.000
Filtered 1,3,5	0	590	180	0	1.000	0.981
	0.6	447	37	0	1.000	0.996
	0.7	415	5	0	1.000	0.999
	0.8	396	1	15	0.963	1.000

Table A-5: maSigPro results for scenario 4 with original and filtered data with different strategies.

SCENARIO 5						
DATA	**RSQ**	**SELECTION**	**FP**	**FN**	**SENSIT**	**SPECIF**
Original	0	255	0	155	0.622	1.000
	0.6	215	0	195	0.524	1.000
	0.7	84	0	326	0.205	1.000
	0.8	8	0	402	0.020	1.000
Filtered 1,2,0	0	211	0	199	0.515	1.000
	0.6	154	0	256	0.376	1.000
	0.7	72	0	338	0.176	1.000
	0.8	6	0	404	0.015	1.000
Filtered 1,2,1	**0**	**3702**	**3294**	**2**	**0.995**	**0.654**
	0.6	**1829**	**1429**	**10**	**0.976**	**0.850**
	0.7	**1035**	**653**	**28**	**0.932**	**0.931**
	0.8	**558**	**206**	**58**	**0.859**	**0.978**
Filtered 1,2,3	0	7125	6715	0	1.000	0.294
	0.6	4534	4125	1	0.998	0.566
	0.7	3089	2681	2	0.995	0.718
	0.8	1527	1125	8	0.980	0.882
Filtered 1,2,5	0	7389	6979	0	1.000	0.266
	0.6	4843	4434	1	0.998	0.534
	0.7	3393	2984	1	0.998	0.686
	0.8	1771	1366	5	0.988	0.856
Filtered 1,3,0	0	229	0	181	0.559	1.000
	0.6	206	0	204	0.502	1.000
	0.7	83	0	327	0.202	1.000
	0.8	8	0	402	0.020	1.000
Filtered 1,3,1	0	3992	3583	1	0.998	0.623
	0.6	2043	1641	8	0.980	0.827
	0.7	1144	759	25	0.939	0.920
	0.8	581	227	56	0.863	0.976
Filtered 1,3,3	0	7442	7032	0	1.000	0.261
	0.6	4770	4360	0	1.000	0.542
	0.7	3270	2861	1	0.998	0.699
	0.8	1624	1218	4	0.990	0.872
Filtered 1,3,5	0	7686	7276	0	1.000	0.235
	0.6	5095	4685	0	1.000	0.507
	0.7	3584	3175	1	0.998	0.666
	0.8	1868	1461	3	0.993	0.846

Table A-6: maSigPro results for scenario 5 with original and filtered data with different strategies.

SCENARIO 6						
DATA	RSQ	SELECTION	FP	FN	SENSIT	SPECIF
Original	0	255	0	155	0.622	1.000
	0.6	199	0	211	0.485	1.000
	0.7	64	0	346	0.156	1.000
	0.8	4	0	406	0.010	1.000
Filtered 1,2,0	0	203	0	207	0.495	1.000
	0.6	151	0	259	0.368	1.000
	0.7	64	0	346	0.156	1.000
	0.8	2	0	408	0.005	1.000
Filtered 1,2,1	**0**	**1198**	**805**	**17**	**0.959**	**0.915**
	0.6	**610**	**235**	**35**	**0.915**	**0.975**
	0.7	**391**	**49**	**68**	**0.834**	**0.995**
	0.8	**311**	**3**	**102**	**0.751**	**1.000**
Filtered 1,2,3	0	2570	2165	5	0.988	0.772
	0.6	1041	648	17	0.959	0.932
	0.7	560	184	34	0.917	0.981
	0.8	358	25	77	0.812	0.997
Filtered 1,2,5	0	3012	2605	3	0.993	0.726
	0.6	1214	818	14	0.966	0.914
	0.7	646	266	30	0.927	0.972
	0.8	381	39	68	0.834	0.996
Filtered 1,3,0	0	228	0	182	0.556	1.000
	0.6	194	0	216	0.473	1.000
	0.7	66	0	344	0.161	1.000
	0.8	5	0	405	0.012	1.000
Filtered 1,3,1	0	1460	1055	5	0.988	0.889
	0.6	656	277	31	0.924	0.971
	0.7	419	73	64	0.844	0.992
	0.8	315	3	98	0.761	1.000
Filtered 1,3,3	0	3016	2609	3	0.993	0.726
	0.6	1144	746	12	0.971	0.922
	0.7	592	216	34	0.917	0.977
	0.8	371	30	69	0.832	0.997
Filtered 1,3,5	0	3484	3077	3	0.993	0.676
	0.6	1328	927	9	0.978	0.903
	0.7	692	311	29	0.929	0.967
	0.8	391	45	64	0.844	0.995

Table A-7: maSigPro results for scenario 6 with original and filtered data with different strategies.

Annex 5: Details of the evolution of specificity and sensitivity with maSigPro on simulated study 1 of Chapter 6.

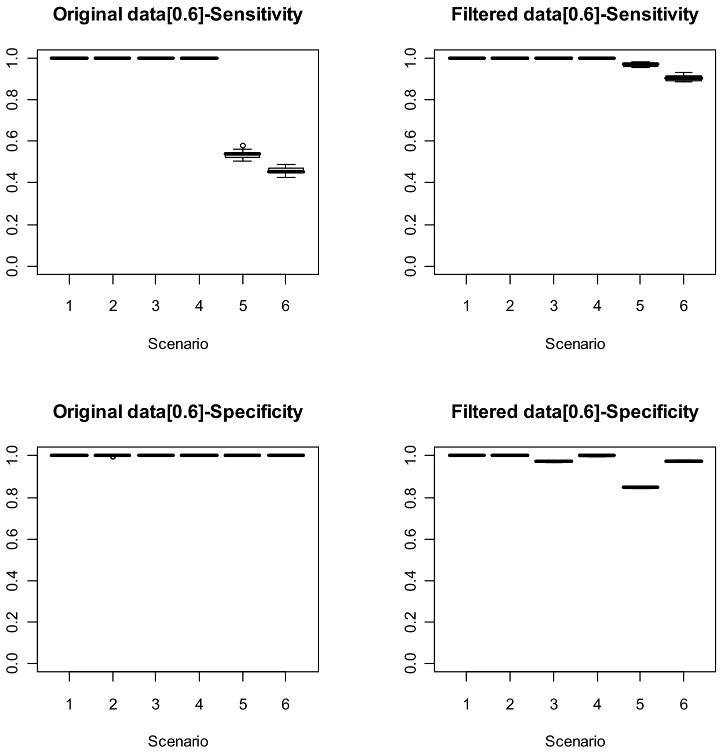


Figure A-7: Sensitivity and specificity maSigPro results with *R*-squared=0.6 of original and ASCA-filtered data simulated in six different scenarios.

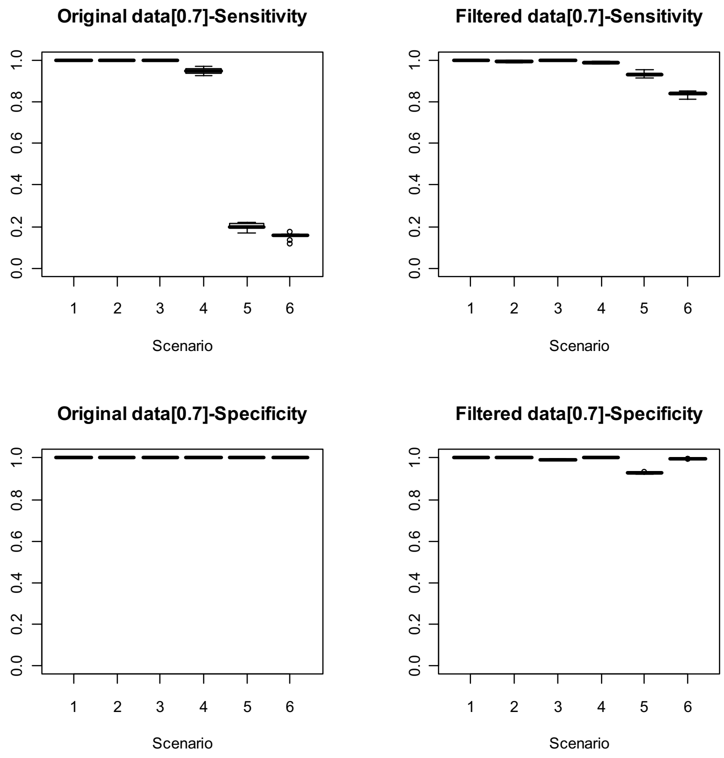


Figure A-8: Sensitivity and specificity maSigPro results with *R*-squared=0.7 of original and ASCA-filtered data simulated in six different scenarios.

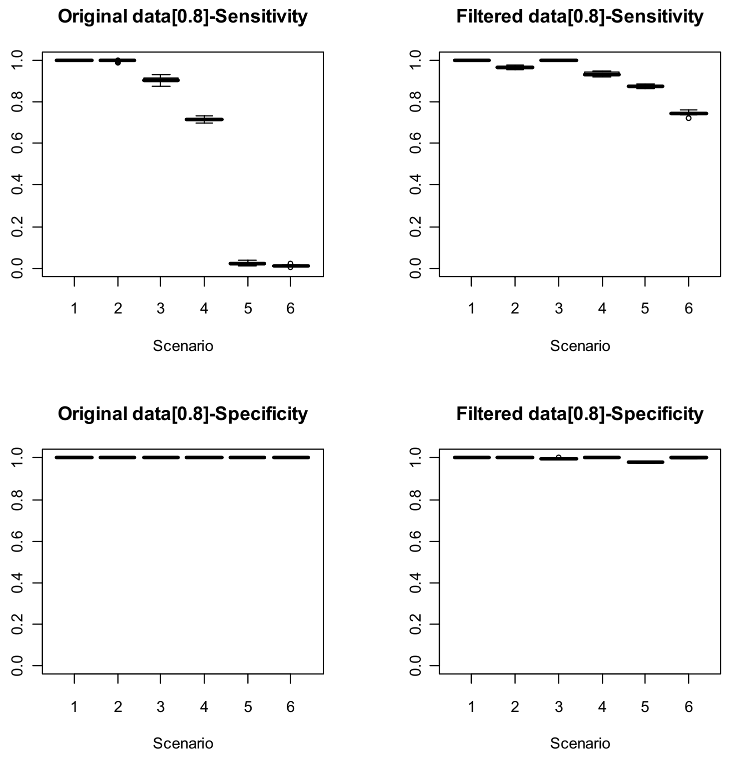


Figure A-9: Sensitivity and specificity maSigPro results with *R*-squared=0.8 of original and ASCA-filtered data simulated in six different scenarios.

To carry out the ANOVA analysis we applied the $\arcsin\sqrt{x}$ transformation to the sensitivity and specificity. This transformation is necessary to avoid the dependence between the variance and the average, as these parameters are proportions. Doing this we could see in the residuals graphs of the respective ANOVA models that the residuals are independent (graphs not shown). Table A-8 includes the ANOVA tables for these two transformed measures. We can observe that all the effects are statistically significant. Interaction ScenarioxData graphs of the transformed sensitivity and specificity have the same evolution than those shown in Chapter 6.

ANALYSIS OF VARIANCE arc.sin(sqrt(sensitivity))					
Source of Variation	**D.F.**	**Sum Squares**	**Mean Square**	**F value**	**Pr(>F)**
Data	1	25547	25547	21428.30	< 2.2E-16
Scenario	5	119135	23827	19985.36	< 2.2E-16
R-squared	2	12363	6181	5184.73	< 2.2E-16
Data x Scenario	5	36245	7249	6080.28	< 2.2E-16
Data x R-squared	2	2619	1310	1098.50	< 2.2E-16
Scenario x R-squared	10	5923	592	496.80	< 2.2E-16
Data x Scenario x R-squared	10	2240	224	187.84	< 2.2E-16
Residuals	324	386	1		

ANALYSIS OF VARIANCE arc.sin(sqrt(specificity))					
Source	**D.F.**	**Sum Squares**	**Mean Square**	**F value**	**Pr(>F)**
Data	1	805.0	805.0	421.87	< 2.2E-16
Scenario	5	2534.0	506.8	265.58	< 2.2E-16
R-squared	2	995.7	497.9	260.90	< 2.2E-16
Data x Scenario	5	3658.5	731.7	383.45	< 2.2E-16
Data x R-squared	2	96.8	48.4	25.37	5.79E-11
Scenario x R-squared	10	888.3	88.8	46.55	< 2.2E-16
Data x Scenario x R-squared	10	1279.5	128.0	67.05	< 2.2E-16
Residuals	324	618.3	1.9		

Table A-8: ANOVA tables for transformed sensitivity and specificity.

Annex 6: Enrichment functional analysis of experimental data used in Chapter 6

	maSigPro		timecourse		EDGE	
	Original	**Filtered**	**Original**	**Filtered**	**Original**	**Filtered**
ribosome	X	X	X	X	X	X
translation	X	X	X	X	X	X
cytosol	X	X	X	X	X	X
oxidoreductase activity	X	X		X	X	X
retinol binding	X	X				
nitric oxide mediated signal transduction	X	X		X		
heme binding		X	X		X	X
lipid metabolic process		X				
glutathione transferase activity		X				X
long-chain fatty acid metabolic process	X					
aldo-keto reductase activity				X		
peroxisome					X	
coenzyme binding					X	
fatty acid metabolic process					X	
iron ion binding					X	

Table A-9: Summary of the biological processes detected with maSigPro, timecourse and EDGE applied to the original and filtered data of Toxicogenomics experiment. Red crosses indicate that the process is over-represented in filtered data and not in the original, and blue crosses is the opposite case.

	maSigPro		timecourse		EDGE	
	Original	**Filtered**	**Original**	**Filtered**	**Original**	**Filtered**
Hormonal Response						
Abscisic acid mediated signaling	X	X		X		
Jasmonic acid				X		
response to salicylic acid stimulus				X		
auxin mediated signaling pathway				X		
Response to stimuli						
response to stress	X	X		X	X	
Response to light				X	X	
Response to abiotic stimuli	X	X				
Response to biotic stimuli				X		
Response to water peroxide			X			
response to osmotic stress			X	X		
response to toxin			X			
Enzimatic activities						
pectinesterase inhibitor activity		X	X	X		
racemase and epimerase activity		X		X		
isomerase activity		X				
phosphoric monoester hydrolase activity			X			
oxidoreductase activity			X	X		
phosphoric ester hydrolase activity			X			
lyase activity			X	X		
transferase activity, transferring hexosyl groups			X			
N-acetyltransferase activity			X	X		
protein tyrosine/serine/threonine phosphatase activity			X	X		
phosphoprotein phosphatase activity			X	X		
glutathione transferase activity				X		
dodecenoyl-CoA delta-isomerase activity				X		
monodehydroascorbate reductase (NADH) activity				X		
3-hydroxyacyl-CoA dehydratase activity				X		
4-coumarate-CoA ligase activity				X		
3-hydroxybutyryl-CoA epimerase activity				X		
glutamate metabolic process				X		
hydrolase activity, acting on glycosyl bonds				X		
dephosphorylation				X		
Binding						
FK506 binding		X				
rRNA binding		X				
protein folding		X				
drug binding		X				
Protein metabolism						
protein folding		X			X	
cysteine-type endopeptidase activity		X	X	X		
cellular carbohydrate metabolic process			X			
sucrose synthase activity			X			
phenylpropanoid metabolic process			X	X		
glutamine family amino acid metabolic process			X	X		
inmune response				X		
spermidine metabolic process				X		
Others						
Chloroplast		X				
cell wall			X			
glyoxysome			X	X		
leaf senescence				X		

Table A-10: Summary of the biological processes detected with maSigPro, timecourse and EDGE applied to the original and filtered data of NSF potato stress experiment. Red crosses indicate that the process is over-represented in filtered data and not in the original, and blue crosses is the opposite case.

Annex 7: Details of simulated datasets results of Chapter 7

In this annex we present detailed information about the results of maSigFun, PCA-maSigFun and ASCA-functional to the simulation studies A and B of analysed in Chapter 7. In all tables results are provided as average and limits of the confidence interval at 95% of the selection, computed from 50 independent simulation runs. False positives (FP), false negatives (FN), sensitivity (SENSIT) and specificity (SPECIF) are reported.

maSigFun

Simulation study A

%Changing genes	R2	SELECTION	FP	FN	SENSIT	SPECIF
20	0	16.02 ± 0.49	1.1 ± 0.28	10.08 ± 0.36	0.6 ± 0.01	1 ± 0
	0.4	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.6	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.8	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
30	0	22.32 ± 0.69	1.7 ± 0.4	4.38 ± 0.48	0.82 ± 0.02	0.99 ± 0
	0.4	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.6	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.8	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
40	0	23.66 ± 0.49	1.38 ± 0.31	2.72 ± 0.34	0.89 ± 0.01	0.99 ± 0
	0.4	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.6	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.8	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
50	0	24.26 ± 0.52	1.58 ± 0.35	2.32 ± 0.29	0.91 ± 0.01	0.99 ± 0
	0.4	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.6	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.8	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
60	0	26.96 ± 0.38	1.98 ± 0.38	0.02 ± 0.04	1 ± 0	0.99 ± 0
	0.4	2.52 ± 0.43	0 ± 0	22.48 ± 0.434	0.1 ± 0.02	1 ± 0
	0.6	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.8	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
70	0	26.62 ± 0.34	1.62 ± 0.34	0 ± 0	1 ± 0	0.99 ± 0
	0.4	24.1 ± 0.24	0 ± 0	0.9 ± 0.24	0.96 ± 0.01	1 ± 0
	0.6	2.12 ± 0.37	0 ± 0	22.88 ± 0.37	0.08 ± 0.01	1 ± 0
	0.8	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
80	0	27.06 ± 0.47	2.06 ± 0.47	0 ± 0	1 ± 0	0.99 ± 0
	0.4	25 ± 0	0 ± 0	0 ± 0	1 ± 0	1 ± 0
	0.6	12.96 ± 0.61	0 ± 0	12.04 ± 0.61	0.52 ± 0.02	1 ± 0
	0.8	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0

Figure A-10: Results of simulated datasets for different proportions of co-expression in functional categories and a mixed class size.

Simulation study B

#Genes in each category	R2	SELECTION	FP	FN	SENSIT	SPECIF
5	0	26.66 ± 0.37	1.66 ± 0.37	0 ± 0	1 ± 0	0.99 ± 0
	0.4	25 ± 0	0 ± 0	0 ± 0	1 ± 0	1 ± 0
	0.6	12.86 ± 0.67	0 ± 0	12.14 ± 0.67	0.51 ± 0.03	1 ± 0
	0.8	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
10	0	26.82 ± 0.44	1.82 ± 0.44	0 ± 0	1 ± 0	0.99 ± 0
	0.4	21.58 ± 0.52	0 ± 0	3.42 ± 0.52	0.86 ± 0.02	1 ± 0
	0.6	0.08 ± 0.11	0 ± 0	24.92 ± 0.11	0 ± 0	1 ± 0
	0.8	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
50	0	27.08 ± 0.42	2.08 ± 0.42	0 ± 0	1 ± 0	0.99 ± 0
	0.4	24.84 ± 0.10	0 ± 0	0.16 ± 0.10	0.99 ± 0	1 ± 0
	0.6	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.8	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
100	0	26.94 ± 0.44	1.94 ± 0.44	0 ± 0	1 ± 0	0.99 ± 0
	0.4	25 ± 0	0 ± 0	0 ± 0	1 ± 0	1 ± 0
	0.6	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.8	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0

Figure A-11: Results of simulated datasets with 70% of coexpression for different sizes of functional category.

#Genes in each category	R2	SELECTION	FP	FN	SENSIT	SPECIF
5	0	10.96 ± 1.04	0.84 ± 0.33	14.88 ± 0.91	0.4 ± 0.04	1 ± 0
	0.4	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.6	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.8	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
10	0	26.66 ± 0.44	1.66 ± 0.44	0 ± 0	1 ± 0	0.99 ± 0
	0.4	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.6	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.8	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
50	0	27.22 ± 0.43	2.22 ± 0.43	0 ± 0	1 ± 0	0.99 ± 0
	0.4	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.6	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.8	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
100	0	26.9 ± 0.43	1.9 ± 0.43	0 ± 0	1 ± 0	0.99 ± 0
	0.4	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.6	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.8	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0

Figure A-12: Results of simulated datasets with 50% of coexpression for different sizes of functional category.

#Genes in each category	R2	SELECTION	FP	FN	SENSIT	SPECIF
5	0	11.46 ± 1.03	1.08 ± 0.38	14.62 ± 0.87	0.42 ± 0.04	1 ± 0
	0.4	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.6	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.8	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
10	0	15.08 ± 0.99	1.2 ± 0.34	11.12 ± 0.89	0.56 ± 0.04	0.99 ± 0
	0.4	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.6	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.8	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
50	0	26.62 ± 0.35	1.62 ± 0.35	0 ± 0	1 ± 0	0.99 ± 0
	0.4	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.6	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.8	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
100	0	26.98 ± 0.41	1.98 ± 0.41	0 ± 0	1 ± 0	0.99 ± 0
	0.4	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.6	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0
	0.8	0 ± 0	0 ± 0	25 ± 0	0 ± 0	1 ± 0

Figure A-13: Results of simulated datasets with 30% of coexpression for different sizes of functional category.

PCA-maSigFun

Simulation study A

%Changing genes	SELECTION	FP	FN	SENSIT	SPECIF
20	25.76 ± 0.23	0.76 ± 0.23	0 ± 0	1 ± 0	1 ± 0
30	25.62 ± 0.24	0.62 ± 0.24	0 ± 0	1 ± 0	1 ± 0
40	25.52 ± 0.22	0.52 ± 0.22	0 ± 0	1 ± 0	1 ± 0
50	25.4 ± 0.21	0.4 ± 0.21	0 ± 0	1 ± 0	1 ± 0
60	25.52 ± 0.20	0.52 ± 0.20	0 ± 0	1 ± 0	1 ± 0
70	25.42 ± 0.21	0.42 ± 0.21	0 ± 0	1 ± 0	1 ± 0
80	25.44 ± 0.21	0.44 ± 0.21	0 ± 0	1 ± 0	1 ± 0

Figure A-14: Results of simulated datasets for different proportions of co-expression in functional categories and a mixed class size.

Simulation study B

#Genes in each category	SELECTION	FP	FN	SENSIT	SPECIF
5	25.28 ± 0.14	0.28 ± 0.14	0 ± 0	1 ± 0	1 ± 0
10	25.5 ± 0.22	0.5 ± 0.22	0 ± 0	1 ± 0	1 ± 0
50	25.56 ± 0.23	0.56 ± 0.23	0 ± 0	1 ± 0	1 ± 0
100	25.84 ± 0.28	0.84 ± 0.28	0 ± 0	1 ± 0	1 ± 0

Figure A-15: Results of simulated datasets with 70% of coexpression for different sizes of functional category.

#Genes in each category	SELECTION	FP	FN	SENSIT	SPECIF
5	25.22 ± 0.17	0.22 ± 0.17	0 ± 0	1 ± 0	1 ± 0
10	25.42 ± 0.17	0.42 ± 0.17	0 ± 0	1 ± 0	1 ± 0
50	25.7 ± 0.25	0.7 ± 0.25	0 ± 0	1 ± 0	1 ± 0
100	25.58 ± 0.24	0.58 ± 0.24	0 ± 0	1 ± 0	1 ± 0

Figure A-16: Results of simulated datasets with 50% of coexpression for different sizes of functional category.

#Genes in each category	SELECTION	FP	FN	SENSIT	SPECIF
5	25.38 ± 0.19	0.38 ± 0.19	0 ± 0	1 ± 0	1 ± 0
10	25.54 ± 0.22	0.54 ± 0.22	0 ± 0	1 ± 0	1 ± 0
50	25.74 ± 0.23	0.74 ± 0.23	0 ± 0	1 ± 0	1 ± 0
100	25.6 ± 0.24	0.6 ± 0.24	0 ± 0	1 ± 0	1 ± 0

Figure A-17: Results of simulated datasets with 30% of coexpression for different sizes of functional category.

ASCA-Functional

Simulation study A

%Changing genes	SELECTION	FP	FN	SENSIT	SPECIF
20	0.48 ± 0.33	0.06 ± 0.07	24.58 ± 0.32	0.02 ± 0.01	1 ± 0
30	2.86 ± 0.57	0.06 ± 0.07	22.2 ± 0.56	0.11 ± 0.02	1 ± 0
40	9 ± 0.75	0.12 ± 0.11	16.12 ± 0.75	0.36 ± 0.03	1 ± 0
50	16.64 ± 0.47	0.18 ± 0.12	8.54 ± 0.45	0.66 ± 0.02	1 ± 0
60	25.26 ± 0.14	0.26 ± 0.14	0 ± 0	1 ± 0	1 ± 0
70	25.5 ± 0.18	0.5 ± 0.18	0 ± 0	1 ± 0	1 ± 0
80	25.6 ± 0.26	0.6 ± 0.26	0 ± 0	1 ± 0	1 ± 0

Figure A- 18: Results of simulated datasets for different proportions of co-expression in functional categories and a mixed class size.

Simulation study B

#Genes in each category	SELECTION	FP	FN	SENSIT	SPECIF
5	25.16 ± 0.13	0.16 ± 0.13	0 ± 0	1 ± 0	1 ± 0
10	25.24 ± 0.12	0.24 ± 0.12	0 ± 0	1 ± 0	1 ± 0
50	25.68 ± 0.29	0.68 ± 0.29	0 ± 0	1 ± 0	1 ± 0
100	25.58 ± 0.22	0.58 ± 0.22	0 ± 0	1 ± 0	1 ± 0

Figure A-19: Results of simulated datasets with 70% of coexpression for different sizes of functional category.

#Genes in each category	SELECTION	FP	FN	SENSIT	SPECIF
5	0.3 ± 0.13	0 ± 0	24.7 ± 0.13	0.01 ± 0.01	1 ± 0
10	0.7 ± 0.25	0 ± 0	24.3 ± 0.25	0.03 ± 0.01	1 ± 0
50	1.52 ± 0.48	0 ± 0	23.48 ± 0.48	0.06 ± 0.02	1 ± 0
100	1.66 ± 0.43	0 ± 0	23.34 ± 0.43	0.07 ± 0.02	1 ± 0

Figure A-20: Results of simulated datasets with 50% of coexpression for different sizes of functional category.

#Genes in each category	SELECTION	FP	FN	SENSIT	SPECIF
5	0.5 ± 0.27	0 ± 0	24.5 ± 0.27	0.02 ± 0.01	1 ± 0
10	0.64 ± 0.25	0 ± 0	24.36 ± 0.25	0.03 ± 0.01	1 ± 0
50	1.46 ± 0.36	0 ± 0	23.54 ± 0.36	0.06 ± 0.01	1 ± 0
100	1.38 ± 0.38	0 ± 0	23.62 ± 0.38	0.06 ± 0.02	1 ± 0

Figure A-21: Results of simulated datasets with 30% of coexpression for different sizes of functional category.

Annex 8: Gene Ontology (GO)

Gene Ontology (GO) Consortium was developed over the past few years in order to unify different formats and vocabularies used in the existing databases of functional gene annotation (Ashburner *et al*., 2000 and Ashburner *et al*., 2001). The advantage of such ontology is the ability to explore functional annotations of genomes of different organisms in an automatic way. This ontology provides a set of structured and controlled vocabulary for specific biological domains that can be used to describe gene products in any organism.

Several organisms database groups joined to carry out this project. They annotate genes and gene products using GO vocabulary terms and incorporate these annotations into their respective model organism databases. Each database contributes its annotation files to a shared GO data resource accessible to the public at http://www.geneontology.org/.

GO includes three extensive ontologies to describe molecular function, biological process, and cellular component. These are attributes of a gene, a gene product or a gene product group. The molecular function is the biochemical activity of a gene product. The biological process is a biological objective to which the gene or gene product contributes. The cellular component refers to the place in the cell where the gene product is active. A gene product can have one or more molecular functions, be used in one or more biological processes and may be associated with one or more cellular components that are related between them.

Each term in GO is a node of a Directed Acyclic Graph (DAG), which is a tree where it is possible for a node to have more than one parent. The relationship between a child and a parent can be: "it is part of", "it is a" and "it regulates". Each GO term is identified with "*GO:nnnnnnn*", where *nnnnnnn* is an integer of seven digits.

GO is dynamic, the ontologies must be updated continuously as more information becomes available. To keep up with the new information, each GO term is cross-referenced with external databases (GenBank, EMBL, SWISS-PROT,…) by links.

Due to the availability of GO and their cross-referencing databases, GO terms are usually used to obtain biological meaning to the results of a microarray experiment. There are many tools to provide efficient search mechanisms that return quickly the annotation information associated with specific lists of genes (for instance Blast2GO tool used in this thesis, Conesa *et al*., 2005). These tools incorporate also statistical methods to decide if a functional category is more represented in the obtained sample than in the whole microarray to offer a list of statistical significant functional categories. This is the named "functional enrichment analyses". Fisher's exact test, the Kolmogorov-Smirnov test, or the chi-squared are common statistics to identify statistical significant functional classes (Rivals *et al*., 2007). These methods consider only the functional categories related to the subset of genes selected and most of the tests used to this selection assume independence in the behaviours of the genes (Dopazo, 2008). Therefore, there are also appearing approaches that consider the functional role of genes while trying to capture the cooperative acting of the whole set of genes, GSA (Subramanian *et al.*, 2005) and FatiScan (Al-Sharour *et al*. 2007).

References

References

A

Affimetrix (1999) Affimetrix Microarray Suite User Guide. Affymetrix, Santa Clara, CA.

Ahmed, F.E. (2002) Molecular techniques for studying gene expression in carcinogenesis. Journal of Environmetal Science and Health, 20 (2): 77-116.

Al-Shahrour, F.; Diaz-Uriarte, R. and Dopazo, J. (2005) Discovering molecular functions significantly related to phenotypes by combining gene expression data and biological information. Bioinformatics, 21: 2988-2993.

Al-Shahrour, F.; Arbiza, L.; Dopazo, H.; Huerta-Cepas, J.; Mínguez, P.; Montaner, D. and Dopazo, J. (2007) From genes to functional classes in the study of biological systems. BMC Bioinformatics, 8: 114.

Arbeitman, M.; Furlong, E.; Imam F.; Johnson, E.; Null, B.; Baker, B.; Krasnow, M.; Scott, M.; Davis, R. and White, K. (2002) Gene expression during the life cycle of drosophila melanogaster. Science, 298, 2270-75.

Ashburner, M.; Ball, C.A.; Blake, J.A.; Botstein, D.; Butler, H.; Cherry, J.M.; Davis, A.P.; Dolonski, K.; Kwight, S.S.; Eppig, J.T.; Harris, M.A.; Hill, D.P.; Issel-Tarver, L.; Kasarskis, A.; Lewis, S.; Matese, J.C.; Richardson, J.E.; Ringwald, M.; Rubin, G.M. and Sherlock, G. (2000) Gene Ontology: tool for the unification of biology. The Gene Ontology Consortium. Nat Genet, 25 (I): 25-29.

Ashburner, M.; Ball, C.A.; Blake, J.A.; Butler, H.; Cherry, J.M.; Corradi, J.; Dolonski, K.; Eppig, J.T.; Harris, M.A.; Hill, D.P.; Lewis, S.; Marshall, B.; Mungall, C.; Reiser, L.; Rhee, S.; Richarson, J.E.; Rchter, J.; Ringwald, M.; Rubin, G.M.; Sherlock, G. and Yoon, J. (2001) Creating the gene ontology resource: Design and implementation. Genome Research, 11 (8): 1425-1422.

Azuaje, F.; Al-Shahrour, F. and Dopazo, J. (2006) Ontology-driven approaches to analyzing data in functional genomics. Methods Mol. Biol., 316: 67-86.

B

Bansal, M.; Gatta, G.D. and di Bernardo, D. (2006) Inference of gene regulatory networks and compound mode of action from time course gene expression profiles. Bioinformatics, 22 (7):815-22.

Bansal, M.; Belcastro, V.; Ambesi-Impiombato, A. and diBernardo, D. (2007) How to infer gene networks from expression profiles. Molecular Systems Biology, 3:78.

Bar-Joseph, Z.; Gerber, G.; Jaakkola, T.; Gifford, D. and Simon, I. (2003) Comparing the continuous representation of time series expression profiles to identify differ-entially expressed genes. Proc. Natl. Acad. Sci. USA (PNAS), 100 (18), 10146-10151.

Bar-Joseph, Z. (2004) Analyzing time series gene expression data. Bioinformatics, 20, 2493-2503.

Beal, M.J.; Falciani, F.; Ghahramani, Z.; Rangel, C. and Wild, D.L. (2005) A Bayesian approach to reconstructing genetic regulatory networks with hidden factors. Bioinformatics, 21 (3), 349-356.

Benjamini, Y. and Hochberg, Y. (1995) Controlling the false discovery rate: a practical and powerful approach to multiple testing. J. Roy. Stat. Soc. B., 57, 289-300.

Bolstad, B.M.; Irizarry, R.A.; Astrand, M. and Speed, T.P. (2003) A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. Bioinformatics, 19 (2), 185-193.

Boulesteix, A.L. (2004) PLS dimension reduction for classification with microarray data. Statistical Applications in Genetics and molecular Biology, 3, Issue 1, Article 33.

Boulesteix, A.L.; Strobl, C.; Augustin, T. and Daumer, M. (2008) Evaluating microarray-based classifiers: an overview. Cancer informatics, 4:77-97.

Box, G.E.P. (1954) Some theorems on quadratic forms applied in the study of analysis of variance problems: effect of inequality of variance in one-way classification. The annals of mathematical statistics, 25, 290-302.

Brun, M.; Kim, S.; Choi, W. and Dougherty, E.R. (2007) Comparison of Gene Regulatory Networks via Steady-state Trajectories. EURASIP J. on Bioinformatics and Systems Biology, article: 82702.

C

Cercós, M.; Soler, G.; Iglesias, D.J.; Gadea, J.; Forment, J. and Talón, M. (2006) Global analysis of gene expression during development and ripening of citrus fruit flesh. A proposed mechanism for citric Acid utilization. Plant Mol Biol., 62(4-5): 513-27.

Chen X.; Wang. L; Smith, J.D. and Zhang, B. (2008) Supervised principal component analysis for gene set enrichment of microarray data with continuous or survival outcomes. Bioinformatics, 24 (21): 2474-81.

Ching, W.; Zhang, S.; Ng, M.K. and Akutsu, T. (2007) An approximation method for solving the steady-state probability distribution of probabilistic Boolean networks. Bioinformatics, 12: 1511-1518.

Cho, R.; Campbell, M; Winzeler, E.; Steinmetz, E.; Conway, A.; Wodicka, L.; Wolsfsberg, T.; Gabrielian, A.; Landsman, D.; Lockhart, D. and Davis, R. (1998) A genome-wide transcriptional analysis of the mitotic cell cycle. Mol Cell., 2(1):65-73.

Chu, S.; DeRisi, J.; Eisen, M.; Mulholland, J.; Botstein, D.; Brown, P.O. and Herskowitz, I. (1998) The transcriptional program of sporulation in budding yeast. Science, 282, 699-705.

Cleveland, W.S. and Loader, C. (1996) Smoothing by Local Regression: Principles and Methods. W.Haerdle & M.G. Schimek editores, "Statistical Theory and Computational aspects of Smoothing". Springer, New York, 10-49.

Conesa, A.; Gotz, S.; García-Gómez, J.M.; Terol, J.; Talón, M. and Robles, M. (2005) Blast2GO: a universal tool for annotation, visualization and analysis in functional genomics research. Bioinformatics, 21 (18), 3674-3676.

Conesa, A.; Nueda, M.J.; Ferrer, A. and Talón, M. (2006) maSigPro: a Method to Identify Significantly Differential Expression Profiles in Time-Course Microarray Experiments. Bioinformatics, 22 (9), 1096-1102.

Conesa, A.; Bro, R.; García-García, F.; Prats, J.M.; Götz, S.; Kjeldahl, K.; Montaner, D. and Dopazo, J. (2008) Direct functional assessment of the composite phenotype through multivariate projection strategies. Genomics, doi: 10.1016/j.ygeno.2008.05.015.

Cui, X. and Churchill, G.A. (2003) Statistical tests for differential expression in cDNA microarray experiments. Genome Biol. 4: 210-220.

D

Dai, J.J.; Lieu, L. and Rocke, D. (2006) Dimension reduction for classification with gene expression microarray data. Statistical applications in genetics and molecular biology, 5, article 6.

DeCook, R. ; Nettleton, D. ; Foster, C. and Wurtele, E.S. (2006) Identifying differentially expressed genes in unreplicated multiple-treatment microarray timecourse experiments. Computational statistics and data analysis, 50: 518-532.

de Hoon, M.J.L.; Imoto, S. and Miyano, S. (2002) Statistical analysis of a small set of time-ordered gene expression data using linear splines. Bioinformatics, 18 (11) 1477-1485.

de Hoon, M.J.L.; Imoto, S. and Kobayashi, K. (2003). Inferring gene regulatory networks from time-ordered gene expression data of Bacillus subtilis using differential equations. Pacific Symposium on Biocomputing, World Scientific, Singapore, Vol.8, pp 17-28.

de Jong, H.D. (2002) Modelling and simulation of genetic regulatory systems: a literature review. J. Comp. Biol., 9 (1): 67-103.

DeRisi, J.L.; Iyer, VR. and Brown PO. (1997) Exploring the metabolic and genetic control of gene expression on a genomic scale. Science 278:680-686.

Dopazo, J. (2006) Functional Interpretation of Microarray Experiments. OMICS:A Journal of Integrative Biology, 10: 398-410.

Dopazo, J. (2008) Formulating and testing hypotheses in functional genomics. Artif. Intell. Med., doi: 10.1016/j.artmed.2008.08.003.

Draghici, S. (2003) Data Analysis Tools for DNA Microarrays. Chapman & Hall/CRC, London.

Draper, N. and Smith, H. (1998) Applied Regression Analysis. 3rd edition. Wiley, New York.

Dudoit, S.; Shaffer, J.P. and Boldrick, J.C. (2002a) Multiple hypothesis testing in microarray experiments. U.C. Berkeley Division of Biostatistics Working Paper Series. 110.

Dudoit, S.; Yang, Y.H.; Calow, M.J. and Speed, T.P. (2002b) Statistical methods for identifying genes with diferential expression in replicated cDNA microarray experiments. Statistica Sinica, 12, 111-139.

Durbin, B.P. and Rocke, D.M. (2004) Variance-stabilizing transformations for two-color microarrays. Bioinformatics, 20, 5: 660-667.

E

Efron, B.; Tibshirani, R.; Storey, J.D. and Tusher, V. (2001) Empirical Bayes Analysis of a Microarray Experiment. J. Am. Stat. Assoc. 96, 1151-1160.

Eisen, M.B.; Spellman, P.T.; Brown, P.O. and Botstein, D. (1998) Cluster analysis and display of genome-wide expression patterns. PNAS, 95(25): 14863-14868.

Ernst, J.; Nau, G.J. and Bar-Joseph, Z. (2005) Clustering short time series gene expression data. Bioinformatics, 21, Suppl.1, 159-168.

Ernst, J. and Bar-Joseph, Z. (2006) STEM: a tool for the analysis of short time series gene expression data. BMC Bioinformatics, 7: 191.

F

Fischer, E.A.; Friedman, M.A. and Markey, M.K. (2007) Empirical comparison of tests for differential expression on time-series microarray experiments. Genomics, 89: 460-470.

Friedman, N.; Linial, M ; Nachman, I. and Pe'er, D. (2000) Using BN to analyze expression data. J. Comp. Biol., 7: 601-620.

G

Geier, F.; Timmer, J. and Fleck C. (2007) Reconstructing gene-regulatory networks from time series, knock-out data and prior knowledge. BMC Systems Biology, I:II.

Goeman, J.J.; van de Geer, S.A.; de Kort, F. and van Houwelingen, H.C. (2004) A global test for groups of genes: testing association with a clinical outcome. Bioinformatics, 20, 1: 93-99.

Gresham, D.; Dunham, M.J. and Botstein, D. (2008) Comparing whole genomes using DNA microarrays. Nature reviews Genetics, 9, 291-302.

Guo, X.; Qi, H.; Verfaillie, C.M. and Pan, W. (2003) Statistical significance analysis of longitudinal gene expression data. Bioinformatics, 19, 13: 1628-1635.

H

Harrell, F. (2002) Regression modeling strategies: with applications to linear models, logistic regression and survival analysis. Springer, New York.

Hartigan JA, Wong MA. (1979) A *k*-means clustering algorithm. Applied Statistics, 28:100-108.

Heijne, W.H.M.; Stierum, R.; Slijper, M.; van Bladeren, P.J. and van Ommen, B. (2003) Toxicogenomics of bromobenzene hepatotoxicity: a combined transcriptomics and proteomics approach. Biochemical Pharmacology, 65, 857-875.

Herrero, J.; Valencia, A. and Dopazo, J. (2001) A hierarchical unsupervised growing neural network for clustering gene expression patterns. Bioinformatics, 17, 126-136.

Herrero, J. and Dopazo, J. (2002) Combining hierarchical clustering and self-organizing maps for exploratory analysis of gene expression patterns. Journal of proteome research, 1, 467-470.

Hilsenbeck, S.G.; Friedrichs, W.E.; Schiff, R.; O'Connell, P.; Hansen, R.K.; Osborne, C.K. and Fuqua, S.W. (1999) Statistical analysis of array expresion data as applied to the problem of tamoxifen resistance. Journal of the National Cancer Institute, 91 (5), 453-459.

Himanen, K.; Vuylsteke, M.; Vanneste, S.; Vercruysse, S.; Boucheron, E.; Alard, P. Chriqui, D.; Van Montagu, M.; Inze, D. and Beeckman, T. (2004) Transcript profiling of early lateral root initiation. PNAS, 101(14):5146-5151.

Huber, W.; Von Heydebreck, A.; Sültmann, H.; Poustka, A.; Vingron, M. (2002) Variance stabilization applied to microarray data calibration and to the quantification of differential expression. Bioinformatics, 18, 96-104.

J

Jansen, J.J.; Hoefsloot, H.C.J.; Timmerman, M.E.; Van der Greef, J.; Westerhuis, J. and Smilde, A.K. (2005) ASCA: analysis of multivariate data obtained from an experimental design. Journal of Chemometrics, 19, 469-481.

Jiang, Z. and Gentleman, R. (2007) Extensions to gene set enrichment. Bioinformatics, 23 (3): 306-313.

K

Kanehisa, M.; Goto, S.; Kawashima, S.; Okuno, Y. and Hattori, M. (2004) The KEGG resource for deciphering the genome. Nucleic Acids Res, 32 (Database issue): D277-D280.

Kapushesky, M.; Kemmeren, P.; Culhane, A.C.; Durinck, S.; Ihmels, J.; Körner, C.; Kull, M.; Torrente, A.; Sarkans, U.; Vilo, J. and Brazma, A. (2004) Expression Profiler: next generation – an online platform for analysis of microarray data, Nucleic Acids Res, 32,, Web Server issue: W465-W470.

Kerr, M.K.; Martin, M. and Churchill, G.A. (2000) Analysis of variance for gene expression microarray data. J. Comput. Biol., 7: 819-837.

Kerr, M.K. and Churchill, G.A. (2001) Experimental design for gene expression microarrays. Biostatistics, 2: 183-201.

Kim, S.Y.; Imoto, S. and Miyano, S. (2003) Inferring gene networks from time series microarray data using dynamic Bayesian networks. Briefings in Bioinformatics, 4, 3: 228-235.

Kim, B.R.; Littell, R.C. and Wu, R.L. (2006) Clustering periodic patterns of gene expression based on Fourier approximations. Current Genomics, 7 (3): 197-203.

Kim, J. and Kim, J.H. (2007) Difference-based clustering of short time-course microarray data with replicates. BMC-Bioinformatics, 8:253.

Kohonen, T. (1997) Self-organizing Maps. Springer, Berlin.

L

Lähdesmäki, H. ; Hautaniemi, S. ; Shmulevich, I. and Yli-Harja, O. (2006)

Relationships between probabilistic Boolean networks and dynamic Bayesian networks as models of gene regulatory networks. Signal Processing. 2006 Apr; 86(4):814-834.

Landgrebe, J.; Wolfgang, W. and Welzl, G. (2002) Permutation-validated principal components analysis of microarray data. Genome Biology, 3 (4), 19.1-19.11.

Larrañaga, P.; Calvo, B.; Santana, R.; Bielza, C.; Galdiano, J.; Inza, I.; Lozano, J.A.; Armañanzas, R.; Santafe, G.; Pérez, A. and Robles, V. (2006) Machine learning in bioinformatics. Briefings in bioinformatics, 7, I : 86 :112.

Leek, J.T.; Monsen, E.; Dabney A.R. and Storey, J.D. (2006) EDGE: extraction and analysis of differential gene expression. Bioinformatics, 22 (4): 507-508.

Liu, H.; Tarima, S.; Borders, A.S.; Getchell, T.V.; Getchell, M.L. and Stromberg, A.J. (2005) Quadratic regression analysis for gene discovery and pattern recognition for non-cyclic short time-course microarray experiments. BMC Bioinformatics, 6, 106.

Lönnstedt, I. and Speed, T. (2002) Replicated microarray data. Statistica Sinica, 12: 21.

Luan, Y. and Li, C. (2003) Clustering of time-course gene expression data using a mixed-effects models with B-splines. Bioinformatics, 19 (4), 474-482.

Lukashin, A.V. and Fuchs, R. (2001) Analysis of temporal gene expression profiles: clustering by simulated annealing and determining the optimal number of clusters. Bioinformatics. 17 (5), 405-414.

M

Ma, P; Castillo-Davis, C.I.; Zhong, W. and Liu, J.S. (2006) A data-driven clustering method for time course expression data. Nucleic Acids Research, 34 (4): 1261-69.

Marsh, L.C. and Cormier, D.R. (2001) Spline Regression Models. Sage University Paper. Series on Quantitative Applications in the Social Sciences.

Martens, H. and Næs, T. (1989) Multivariate calibration, John Wiley & Sons, Ltd. Chichester.

McLachlan, G.J. and Krishnan, T. (1997) The EM algorithm and extensions. Wiley, New York.

McLachlan, G.J.; Peel, D.; Basfor, K.E. and Adams, P. (1999) The EMMIX software for the fitting of mixtures of normal and t-components. J. Stat. Softw., 4.

McLachlan, G.J.; Bean, R.W. and Peel, D. (2002) A mixture model-based approach to the clustering of microarray expression data. Bioinformatics, 18, 3: 413-422.

Medina, I.; Montaner, D.; Tárraga, J. and Dopazo, J. (2007) Prophet, a web-based tool for class prediction using microarray data. Bioinformatics, 23, 3: 390-391.

Mootha, V.K.; Lindgren, C.M.; Eriksson, K.F.; Subramanian, A.; Sihag, S.; Lehar, J.; Puigserver, P.; Carlsson, E.; Ridderstrale, M.; Laurila, E.; Houstis, N.; Daly, M.J.; Patterson, N.; Mesirov, J.P.; Golub, T.R.; Tamayo, P.; Spiegelman, B.; Lander, E.S.; Hirschhorn, J.N.; Altshuler, D. and Groop, L.C. (2003) PGC-l alpha-responsive genes involver in oxidative phosphorylation are co-ordinately downregulated in human diabetes. Nat. Genet, 34 (3): 267-273.

Mulder, N.J.; Apweiler, R.; Attwood, T.K.; Bairoch, A.; Barrell, D.; Bateman, A.; Binns, D.; Biswas, M.; Bradley, P.; Bork, P.; Bucher, P.; Copley, R.R.; Courcelle, E.; Das, U.; Durbin, R.; Falquet, L.; Fleischmann, W.; Griffiths-Jones, S.; Haft, D.; Harte, N.; Hulo, N.; Kahn, D.; Kanapin, A.; Krestyaninova, M.; Lopez, R.; Letunic, I.; Lonsdale, D.; Silventoinen, V.; Orchard, S.E.; Pagni, M.; Peyruc, D.; Ponting, C.P.; Selengut, J.D.; Servant, F.; Sigrist, C.J.A.; Vaughan R. and Zdobnov, E.M. (2003) The InterPro Database, 2003 brings increased coverage and new features. Nucleic Acids Res., 31, 315-318.

Murphy, K. and Mian, S. (1999) Modelling gene expression data using Dynamic Bayesian Networks. Technical Report, University of California, Berkeley, CA.

N

Nueda, M.J.; Conesa, A.; Westerhuis, J.A.; Hoefsloot, H.C.J.; Smilde, A.K.; Talón, M. and Ferrer, A. (2007) Discovering gene expression patterns in Time Course Microarray Experiments by ANOVA-SCA. Bioinformatics, 23 (14), 1792-1800.

Nueda, M.J.; Sebastián, P.; Tarazona, S.; García-García, F.; Dopazo, J.; Ferrer, A. and Conesa, A. (2009) Functional Assessment of Time Course Microarray data. BMC Bioinformatics. In Press.

Nguyen, D. and Rocke, D. (2002) Tumor classification by partial least squares using microarray gene expression data. Bioinformatics, 18 (1), 39-50.

P

Pan, W. (2002) A comparative review of statistical methods for discovering differentially expressed genes in replicated microarray experiments. Bioinformatics, 18: 546-554.

Pan, W.; Lin, J. and Le, C. (2003) A mixture model approach to detecting differentially expressed genes with microarray data. Funct. Integr. Genomics, 3: 117-124.

Papana, A. and Ishwaran, H. (2006) CART variance stabilization and regularization for high-throughput genomic data. Bioinformatics, 22, 18: 2254-2261.

Park, T.; Yi, S.G.; Lee, S. Lee, S.Y.; Yoo, D.H.; Ahn, J.I. and Lee, Y.S. (2003) Statistical tests for identifying differentially expressed genes in time-course microarray experiments. Bioinformatics, 19, 694-703.

Q

Quackenbush, J. (2001) Computational Analysis of Microarray data. Nature Reviews Genetics, 2, 418-427.

R

Ramoni, M.F.; Sebastiani, P. and Kohane, I.S. (2002) Cluster analysis of gene expression dynamics. PNAS, 99(14): 9121-9126.

Rangel, C.; Angus, J.; Ghahramani, Z.; Lioumi, M.; Sotheran, E. ; Gaiba, A.; Wild, D.L. and Falciani, F. (2004) Modelling T-cell activation using gene expression profiling and state space models. Bioinformatics, 20, 1361-1372.

Raychaudhuri, S.; Stuart J.M. and Altman, R.B. (2000) Principal Components analysis to summarize microarray experiments: application to sporulation time series. Pacific Symposium on Biocomputing, 5, 452-463.

Reiner, A.; Yekutieli, D. and Benjamini, Y. (2003) Identifying differentially expressed genes using false discovery rate controlling procedures. Bioinformatics, 19 (3), 368-375.

Rensink, W.A.; Iobst, S.; Hart, A.; Stegalkina, S.; Liu, J. and Buell C.R. (2005) Gene expression profiling of potato responses to cold, heat, and salt stress. Funct. Integr. Genomics, 5 (4):201-207.

Rivals, I; Personnaz, L.; Taing, L. and Potier, M.C. (2007) Enrichment or depletion of a GO category within a class of genes: which test? Bioinformatics, 23, 401-407.

Roden, J.C.; King, B.W.; Trout, D.; Mortazavi, A.; Wold, B. and Hart, C.E. (2006) Mining gene expression data by interpreting principal components. BMC Bioinformatics, 7, 194.

S

Schliep, A.; Schönhuth, A. and Steinhoff, C. (2003) Using hidden Markov models to analyze gene expression time course data. Bioinformatics, 19, Suppl.1: i255-i263.

Schmulevich, I.; Dougherty E.R. and Zhang, W. (2002a) From Boolean to probabilistic Boolean networks as models of genetic regulatory networks. Proc. IEEE, 90, 1778-1792.

Schmulevich, I.; Dougherty E.R. and Zhang, W. (2002b) Gene perturbation and intervention in probabilistic Boolean networks. Bioinformatics, 18 (10): 1319-1331.

Shi, Y.; Mitchell, T. and Bar-Joseph, Z. (2007) Inferring pairwise regulatory relationships from multiple time series datasets. Bioinformatics, 23, 6, 755-63.

Smilde, A.K.; Jansen, J.J.; Hoefsloot, H.C.J.; Lamers, R.J.A.N.; Van der Greef, J. and Timmerman, M.E. (2005) ANOVA-Simultaneous component analysis (ASCA): a new tool for analyzing designed metabolomics data. Bioinformatics, 21 (13), 3043-3048.

Smyth, G.K. (2004) Linear Models and Empirical Bayes Methods for Assessing Differential Expression in Microarray Experiments. Statistical Applications in Genetics and Molecular Biology, 3, (1), article 3.

Spellman, P.T.; Sherlock, G.; Zhang, M.Q.; Iyer, V.R.; Anders, K.; Eisen, M.B.; Brown, P.O.; Botstein, D. and Futcher, B. (1998) Comprehensive identification of cell cycle-regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization. Mol. Biol. Cell., 9 (12), 3273-3297.

Storey, J.D. and Tibshirani, R. (2003) Statistical significance for genomewide studies. PNAS, 100 (16): 9440-9445.

Storey, J.D. (2005) The optimal discovery procedure: a new approach to simultaneous significance testing. UW Biostatistics Working Paper Series, 259.

Storey, J.D.; Xiao, W.; Leek, J.T.; Tompkins, R.G. and Davis, R.W. (2005) Significance analysis of time course microarray experiments. PNAS, 102 (36): 12837-12842.

Storey, J.D.; Dai, J.Y. and Leek J.T. (2007) The optimal discovery procedure for large-scale significance testing, with applications to comparative microarray experiments. Biostatistics, 8, 2: 414-432.

Subramanian, A.; Tamayo, P.; Mootha, V.K.; Mukherjee, S.; Ebert, B.L.; Gillette, M.A.; Paulovich, A.; Pomeroy, S.L.; Golub, T.R.; Lander, E.S. and Mesirov, J.P. (2005) Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. Proc Natl Acad Sci USA, 102 (43): 15545-15550.

T

Tai, Y.C. and Speed, T.P. (2005) Statistical analysis of microarray time course data. In U. Nuber, editor, DNA Microarrays. BIOS Scientific Publisher Limited, Taylor &Francis, 4 Park Square, Milton Park, Abingdon OX14 4RN, Chapter 20.

Tai, Y.C. and Speed, T.P. (2006) A multivariate empirical Bayes statistic for replicated microarray time course data. Annals of Statistics, 34 (5): 2387-2412.

Tamayo, P.; Slonim, D.; Mesirov, J.; Zhu, Q.; Kitareewan, S.; Dmitrovsky, E.; Lander, E. and Golub, T. (1999) Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation. PNAS, 96(6): 2907-2912.

Takashima, K.; Mizukawa, Y.; Morishita, K.; Okuyama, M.; Kasahara, T.; Toritsuka, N.; Miyagishima, T.; Nagao, T. and Urushidani, T. (2006) Effect of the difference in vehicles on gene expression in the rat liver--analysis of the control data in the Toxicogenomics Project Database. Life Sci., 78 (24): 2787-2796.

Tárraga, J.; Medina, I.; Carbonell, J.; Huerta-Cepas, J.; Minguez, P.; Alloza, E.; Al-Shahrour, F.; Vegas-Azcárate, S.; Goetz, S.; Escobar, P.; Garcia-Garcia, F.; Conesa, A.; Montaner, D. and Dopazo, J. (2008) GEPAS, a web-based tool for microarray data analysis and interpretation. Nucleic Acids Research, 36.

Tibshirani, R.; Hastie, T.; Narasimhan, B. and Chu, G. (2002) Diagnosis of multiple cancer types by shrunken centroids of gene expression. Proceedings of the National Academy of Sciences, 99: 6567-6572.

Timmerman, M.E. and Kiers H.A.L. (2003) Four simultaneous component models of multivariate time series from more than one subject to model intraindividual and interindividual differences. Psychometrika, 86, 105-122.

Tomancak, P.; Beaton, A.; Weiszmann, R.; Kwan, E.; Shu, S.; Lewis, S.; Richards, S.; Ashburner, M.; Hartenstein, V.; Celniker, S. and Rubin G. (2002) Systematic determination of patterns of gene expression during drosophila embryogenesis. Genome Biology, 3 (12): 0088.1-008814.

Tusher, V.; Tibshirani, R. and Chu, C. (2001) Significance analysis of microarrays applied to transcriptional responses to ionizing radiation. Proceedings of the National Academy of Sciences, 98, 5116-5121.

V

Vapnik, V.N. (1995) The nature of statistical learning theory. New York. Srpinger.

Vittinghoff, E.; Shiboski, S. C.; Glidden, D. V. and Mc Culloch, C. E. (2005) Regression methods in biostatistics. Linear, logistic, survival, and repeated measures models. Springer, New York.

W

Wang, Y.; Joshi, T.; Zhang, X.S.; Xu, D. and Chen, L. (2006) Inferring gene regulatory networks from multiple microarray datasets. Bioinformatics, 22(19):2413-20.

Wingender, E.; Chen, X.; Hehl, R.; Karas, H.; Liebich, I.; Matys, V.; Meinhardt, T.; Pruss, M.; Reuter, I. and Schacherer, F. (2000) TRANSFAC: an integrated system for gene expression regulation. Nucleic Acids Res, 28 (I): 316-319.

Wolfinger, R.D.; Gibson, G.; Wolfinger, E.; Bennett, L.; Hamadeh, H.; Bushel, P.; Afshari, C. and Paules, R.S. (2001) Assessing gene significance from cDNA microarray expression data via mixed models. Journal of Computational Biology, 8 (6), 625-637.

X

Xu, X.L.; Olson, J.M. and Zhao, L.P. (2002) A regression-based method to identify differentially expressed genes in microarray time course studies and its application in an inducible Huntington's disease transgenic model. Hum Mol Genet., 11 (17), 1977-1985.

Y

Yang, Y. H. Dudoit, S.; Luu, P. and Speed, T.P. (2001) Normalization for cDNA microarray Data. In Bittner, M.L.; Chen, Y.; Dorsel, A.N. and Dougherty, E. R. (eds), Microarrays: Optical Technologies and Informatics. SPIE, Society for Optical Engineering, San Jose, CA.

Yang, Y.H. and Speed, T. (2002) Design issues for cDNA microarray experiments. Nature reviews genetics, 3: 579-588.

Yeung, K.Y. and Ruzzo W.L. (2001) Principal component analysis for clustering gene expresión data. Bioinformatics, 17 (9), 763-774.

Yeung, K.Y.; Fraley, C.; Murua, A.; Raftery, A.E. and Ruzzo, W.L. (2001) Model-based clustering and data transformations for gene expression data. Bioinformatics, 17 (10): 977-987.

Z

Zhang, S. (2007) A comprehensive evaluation of SAM, the SAM R-package and a simple modification to improve its performance. BMC Bioinformatics, 8:230.

Zhu, J. (2004) Classification of gene expression microarrays by penalized logistic regression. Bioestatistics, 5: 427-443.